



UNIVERSIDAD POLITÉCNICA DE CARTAGENA
DEPARTAMENTO DE ESTRUCTURAS Y
CONSTRUCCIÓN

"RECONSTRUCCIÓN GEOMÉTRICA DE SÓLIDOS
UTILIZANDO TÉCNICAS DE OPTIMIZACIÓN "

Julián Conesa Pastor

2001

TESIS DOCTORAL

RECONSTRUCCIÓN GEOMÉTRICA DE SÓLIDOS
UTILIZANDO TÉCNICAS DE OPTIMIZACION

presentada en la
UNIVERSIDAD POLITÉCNICA DE CARTAGENA

por
JULIAN CONESA PASTOR

para la obtención del
GRADO DE DOCTOR INGENIERO INDUSTRIAL

Director de tesis
PEDRO P. COMPANYY CALLEJA

Codirector de tesis
JOSE MARIA GOMIS MARTÍ

Cartagena, Julio de 2001

AUTORIZACIÓN DEL DIRECTOR DE LA TESIS PARA SU PRESENTACIÓN

Dr. Pedro P. Company Calleja como Director de la Tesis Doctoral *Reconstrucción Geométrica de Sólidos utilizando técnicas de optimización* realizada en el Departamento de Estructuras y Construcción de la Universidad Politécnica de Cartagena por el doctorando D. Julián Conesa Pastor, autorizo la presentación de la citada Tesis Doctoral, dado que reúne las condiciones necesarias para su defensa.

En Cartagena, a 16 de Julio de 2001

EL DIRECTOR DE LA TESIS

A handwritten signature in black ink, consisting of several overlapping loops and a long horizontal stroke extending to the right.

Fdo.: Pedro P. Company Calleja

AUTORIZACIÓN DEL CODIRECTOR DE LA TESIS PARA SU PRESENTACIÓN

Dr. José María Gomis Martí como Codirector de la Tesis Doctoral *Reconstrucción Geométrica de Sólidos utilizando técnicas de optimización* realizada en el Departamento de Estructuras y Construcción de la Universidad Politécnica de Cartagena por el doctorando D. Julián Conesa Pastor, autorizo la presentación de la citada Tesis Doctoral, dado que reúne las condiciones necesarias para su defensa.

En Cartagena, a 16 de Julio de 2001

EL CODIRECTOR DE LA TESIS

A handwritten signature in black ink, appearing to be 'J.M. Gomis Martí', written in a cursive style.

Fdo.: José María Gomis Martí

AUTORIZACIÓN DEL TUTOR DE LA TESIS PARA SU PRESENTACIÓN

Dr. Gregorio Sánchez Olivares como Tutor de la Tesis Doctoral *Reconstrucción Geométrica de Sólidos utilizando técnicas de optimización* realizada en el Departamento de Estructuras y Construcción de la Universidad Politécnica de Cartagena por el doctorando D. Julián Conesa Pastor, autorizo la presentación de la citada Tesis Doctoral, dado que reúne las condiciones necesarias para su defensa.

En Cartagena, a 16 de Julio de 2001

EL TUTOR DE LA TESIS

A handwritten signature in black ink, consisting of several overlapping loops and a horizontal line extending to the left, positioned below the text 'EL TUTOR DE LA TESIS'.

Fdo.: Gregorio Sánchez Olivares



UNIVERSIDAD POLITÉCNICA

(Autorización del Departamento responsable)

José Antonio Hernández Cañadas, Director del Departamento de Estructuras y Construcción INFORMA:

Que la Tesis Doctoral titulada “Reconstrucción Geométrica de Sólidos utilizando técnicas de optimización” , ha sido realizada por Don Julián Conesa Pastor, bajo la dirección y supervisión de Don Pedro P. Company Calleja y José María Gomis Martí y que el Departamento ha dado su conformidad para que sea presentada ante la Comisión General de Doctorado.

En Cartagena, a 19 de Julio de 2001

EL DIRECTOR DEL DEPARTAMENTO

A handwritten signature in black ink, appearing to read "J.A.H.C.", with a horizontal line underneath.

Fdo.: José Antonio Hernández Cañadas

AGRADECIMIENTOS

Quiero dedicar en este documento unos breves párrafos en agradecimiento a cuantas numerosas personas me han ayudado a realizar la presente Tesis Doctoral.

En primer lugar expresar mi sincera gratitud a los profesores Dr. Pedro P. Company Calleja y Dr. José María Gomis Martí por su constancia y apoyo en la dirección de esta Tesis Doctoral. Sus ideas y consejos han sido un importante camino para llevar esta Tesis a buen final.

En segundo lugar a todos aquellos profesores del Departamento de Estructuras y Construcción, que sobre todo en las fases iniciales de esta Tesis Doctoral, me han prestado su ayuda. De igual forma, vaya mi agradecimiento a todos los profesores y amigos del Departamento de Expresión Gráfica que siempre me han apoyado incondicionalmente.

No sería justo el no expresar mi agradecimiento también a aquellos profesores de la Universidad Jaume I de Castellón y de la Universidad Politécnica de Valencia con los que en estos años he compartido ideas y trabajos.

En último lugar, pero no por ello menos importante, agradecer el apoyo moral a mi esposa Ana Belen que durante todos estos años me ha animado para finalizar esta Tesis Doctoral, y con la espero poder recuperar algún día los momentos que perdí.

A Ana Belén.

INDICE

INDICE GENERAL

INDICE GENERAL.....	VIII
INDICE DE TABLAS Y FIGURAS	XIV
GLOSARIO DE SÍMBOLOS	XXII
RESUMEN	1
1. INTRODUCCIÓN	3
1.1 VISION POR ORDENADOR Y RECONSTRUCCIÓN GEOMÉTRICA.....	4
1.2 AMBITO DE LA RECONSTRUCCIÓN GEOMÉTRICA.....	7
1.2.1 Vectorización.....	9
1.2.2 Reconstrucción geométrica 2D.....	10
1.2.3 Reconstrucción geométrica 3D.....	11
1.2.4 Refinado del modelo	11
1.3 CLASIFICACIÓN DE LOS MÉTODOS DE RECONSTRUCCIÓN....	12
1.3.1 Número de vistas	12
1.3.2 Modelos generados	14
1.3.3 Metodología y número de soluciones	16
1.4 OBJETIVOS DEL TRABAJO	18
1.5 EL PROGRAMA REFER.....	18
2. ESTADO DEL ARTE	21
2.1 ANÁLISIS DE LOS MÉTODOS DE VISTAS MULTIPLES	22
2.1.1 Métodos basados en la identificación de primitivas	22
2.1.2 Métodos de formulación analítica	25
2.1.3 Métodos basados en el Teorema de Pohlke	26

2.2	ANÁLISIS DE LOS MÉTODOS DE VISTA ÚNICA.....	28
2.2.1	Métodos basados en el etiquetado de vértices	29
2.2.2	Métodos basados en el espacio gradiente	32
2.2.3	Métodos basados en la programación lineal.....	33
2.2.4	Métodos perceptuales	35
3. PERCEPCION VISUAL Y RECONSTRUCCIÓN GEOMÉTRICA		41
3.1	LA RECONSTRUCCIÓN GEOMÉTRICA COMO PROCESO DE OPTIMIZACIÓN	41
3.2	DEFINICION DE LAS VARIABLES DE RECONSTRUCCIÓN	43
3.3	FORMULACIÓN DE LA FUNCIÓN OBJETIVO	49
3.4	REGULARIDADES DE LA IMAGEN	51
3.4.1	Tratamiento de las regularidades en la optimización.....	52
3.4.2	Regularidades de la imagen.....	55
	Regularidad de paralelismo de líneas	55
	Regularidad de planicidad de caras	58
	Regularidad de colinealidad de líneas	61
	Regularidad de ortogonalidad facial oblicua	64
	Regularidad de verticalidad	67
	Regularidad de isometría	69
	Regularidad de mínima desviación estándar de ángulos (MSDA)	71
	Regularidad de perpendicularidad de caras	72
	Regularidad de ortogonalidad de esquinas	73
	Regularidad de ortogonalidad de líneas (MSDP)	75
4. COMPORTAMIENTO DE LOS ALGORITMOS DE OPTIMIZACIÓN EN LA RECONSTRUCCIÓN GEOMÉTRICA.....		79
4.1	DIFERENTES ASPECTOS DE LOS ALGORITMOS DE OPTIMIZACIÓN	80
4.1.1	Naturaleza de las magnitudes de reconstrucción.....	80
4.1.2	Complejidad de la función objetivo	81
4.1.3	Método de búsqueda acorde al espacio de soluciones	81
4.1.4	Restricciones	83
4.2	PROBLEMATCA DE LA OPTIMIZACIÓN EN LA RECONSTRUCCIÓN GEOMÉTRICA.....	84

4.3 ANTECEDENTES DE LA RECONSTRUCCIÓN MEDIANTE OPTIMIZACION	88
4.4 ALGORITMOS IMPLEMENTADOS.....	90
4.4.1 Algoritmo Hill-Climbing.....	90
Estructura del algoritmo Hill-Climbing implementado.....	90
Ensayos y conclusiones	92
4.4.2 Algoritmo Simulated Annealing.....	97
Estructura del algoritmo S.A.	97
Mecanismo de generación de soluciones	99
Aceptación de una solución dada	100
Temperatura inicial.....	101
Ley de enfriamiento de la temperatura	102
Criterio de equilibrio	103
Criterio de congelación.....	104
Elección del punto inicial de reconstrucción.....	105
Ensayos y conclusiones	106
5. ESTRATÉGIAS DE INFLADO PREVIO	111
5.1 ESTRATÉGIAS DE INFLADO CONOCIDAS	112
5.2 ESTRATÉGIAS PROPIAS PARA GENERACIÓN DE MODELOS TENTATIVOS	114
5.2.1 Determinación de la tipología del modelo	114
5.2.2 Inflado axonométrico	117
Aplicaciones del método a otras tipologías.....	127
Ejemplo práctico de la estrategia de inflado axonométrico	135
5.2.3 Inflado por niveles.....	136
Clasificación por niveles de las tipologías.....	137
Detección de aristas predominantes.....	140
Determinaciones analíticas para las obtenciones de tipologías .	143
Correspondencia entre coordenadas Z y niveles	147
Ampliación del método.....	152
5.2.4 Inflado por plano oblicuo	154
6. DETECCIONES PREVIAS EN LA IMAGEN	159
6.1 CONCEPTOS BÁSICOS Y DEFINICIONES DE UN GRAFO GEOMÉTRICO.....	160
6.2 IDENTIFICACIÓN DE CARAS EN UN GRAFO.....	162

6.2.1	Antecedentes de la detección de caras de un poliedro	163
6.2.2	Identificación de circuitos fundamentales	165
6.2.3	Generación del caras potenciales del grafo	166
6.2.4	Selección de las caras del modelo	168
	Ecuaciones de rango máximo	169
	Función de conformidad	172
6.2.5	Selección de la solución final.....	172
6.2.6	Reducción del espacio de circuitos	173
6.3	DETERMINACION DE LAS ARISTAS OCULTAS DE UN GRAFO GEOMÉTRICO	174
6.3.1	Detección de aristas total o parcialmente ocultas.....	175
	Reducción de número de aristas que conforman las posibles configuraciones del modelo	175
	Tratamiento agrupado de subconjuntos de aristas	179
6.3.2	Segmentación de las aristas parcialmente oculta	182
6.3.3	Determinación de las aristas del contorno aparente.....	184
7.	ENSAYOS Y VALIDACIONES	191
7.1	ALGORITMOS DE OPTIMIZACIÓN.....	191
7.2	MÉTODO DE INFLADO PREVIO AXONOMETRICO	192
7.2.1	Modelos de tipología normalón.....	192
7.2.2	Modelos de tipología cuasi-normalon	195
7.2.3	Limitaciones del método.....	198
7.3	MÉTODO DE INFLADO PREVIO POR PLANO OBLICUO	203
7.4	MÉTODO DE INFLADO PREVIO POR NIVELES	206
7.5	REFINADO POR PLANICIDAD DE CARAS	212
7.6	SELECCIÓN DE REGULARIDADES	213
7.6.1	Modelos de tipología normalon y cuasi-normalon.....	213
7.6.2	Modelos de tipología piramidal.....	214
7.6.3	Modelos de tipología prismática.....	215
7.6.4	Modelos de tipología indefinida	216
7.7	PARTICULARIDADES DE LA REGULARIDAD PLANICIDAD DE CARAS	220

7.8	TIPOLOGÍAS DE LOS MODELOS	221
7.9	IDENTIFICACION DE CARAS EN UN GRAFO.....	222
7.9.1	Reducción de caras potenciales propuesta.....	223
7.9.2	Refinado del número de caras definidas en una imagen.....	225
7.10	DETECCION DE ARISTAS OCULTAS	231
7.11	SIMPLIFICACION DEL ESPACIO DE SOLUCIONES	233
8.	CONCLUSIONES Y DESARROLLOS FUTUROS	235
8.1	CONCLUSIONES FINALES	235
8.2	DESARROLLOS FUTUROS	237
8.2.1	Desarrollos abordables a corto plazo	238
8.2.2	Desarrollos no abordables a corto plazo	241
9.	REFERENCIAS	243
	ANEXO 1. BÚSQUEDAS BIBLIOGRAFICAS	249
	ANEXO 2. ALGORITMOS ENSAYADOS EN LA RECONSTRUCCION GEOMÉTRICA.....	263
	ANEXO 3. RELACION DE ESCALAS Y ANGULOS AXONOMETRICOS	273
	ANEXO 4. GLOSARIO DE TÉRMINOS.....	279
	BREVE CURRICULUM	289

INDICE DE TABLAS Y FIGURAS

Figura 1	Campos de acción de la visión por ordenador	5
Figura 2	Fases de la reconstrucción de dibujos de ingeniería	9
Figura 3	Reconstrucción mediante métodos de a) Múltiples vista b) Vista única	13
Figura 4	Modelo representado por CSG	15
Figura 5	Modelo representado por Brep alámbrico (sin caras)	16
Figura 6	Imagen que pueden corresponder a más de un modelo tridimensional	17
Figura 7	Evolución de los métodos basados en identificación de primitivas a partir de vista múltiples. Representación CSG	23
Figura 8	Evolución de los métodos basados en identificación de primitivas a partir de vista múltiples. Representación Brep	24
Figura 9	Evolución de los métodos basados en formulación analítica	25
Figura 10	Evolución de los métodos basados en el Teorema de Pohlke	26
Figura 11	Evolución de los métodos de reconstrucción de vistas múltiples	27
Figura 12	Evolución de los métodos basados en identificación de primitivas a partir de vista única. Representación CSG	28
Figura 13	Evolución de los métodos basados en identificación de primitivas a partir de vista única. Representación Brep	29
Figura 14	Etiquetado de vértices propuesto por Huffman-Clowes	30
Figura 15	Etiquetado propuesto por Malik	31
Figura 16	Evolución de los métodos de etiquetado	32
Figura 17	Evolución de los métodos basados en el espacio gradiente	33
Figura 18	Evolución de los métodos basados en programación lineal	34
Figura 19	Evolución de los métodos basados en reglas de la percepción	36
Figura 20	Evolución de los métodos de reconstrucción de vista única	38
Figura 21	Resumen cronológico de los antecedentes de la Reconstrucción Geométrica	39
Figura 22	Analogía entre la percepción visual humana y el proceso de reconstrucción mediante métodos de optimización de regularidades	43
Figura 23	Concepto de proyección	44
Figura 24	Clases de proyección. A) proyección cónica b) proyección cilíndrica	45
Figura 25	Sistema de referencia para inflado del modelo	46
Figura 26	Regla de las visiones genéricas	47
Figura 27	Hipótesis 1 en la figura de Kopffermann	48

Figura 28	Método de inflado	49
Figura 29	El conjunto de todos los modelos obtenidos por inflado, determinan la “Extensión ortográfica” de una imagen.....	50
Figura 30	Efectos del rigor matemático en la formulación de regularidades	53
Figura 31	Representación gráfica de la función de confianza	54
Figura 32	Regularidad de paralelismo	56
Figura 33	Ley de buena continuación. Paralelismo frente a simetría y simplicidad.....	57
Figura 34	Regularidad de colinealidad	62
Figura 35	Contraejemplo de la regularidad de colinealidad	63
Figura 36	Proyección ortogonal oblicua de tres caras ortogonales	64
Figura 37	Caras con a) ortogonalidad parcial b) sin ortogonalidad	66
Figura 38	Ley de simetría	67
Figura 39	Regularidad de verticalidad	67
Figura 40	Error en la regularidad de verticalidad	69
Figura 41	Consenso de proporcionalidad en las tres dimensiones a) Modelo cúbico b) Modelo paralelepípedo.....	70
Figura 42	Efectos de proporcionalidad en las dimensiones de una imagen.....	71
Figura 43	Vectores unitarios de superficie que determinan la perpendicularidad de caras	73
Figura 44	Posibles proyecciones de esquinas ortogonales.....	74
Figura 45	Modelo imposible a) Modelo definido b) Desarrollo de solución posible	77
Figura 46	Mínimo local en la imagen	85
Figura 47	Distintos mínimos locales en la Reconstrucción Geométrica a) Imagen de partida b) Mínimo global c) Mínimo local	87
Figura 48	Mínimo local obtenido como consecuencia de una mala elección de parámetros que dirigen el algoritmo de optimización a) Mínimo global b) Imagen de partida c) Mínimo local	88
Figura 49	Clasificación de los algoritmo de optimización utilizados por autores anteriores	89
Figura 50	Diagrama de flujo del algoritmo de optimización Hill Climbing.....	91
Figura 51	Diagrama de evolución del coste para escalones definidos por a) Marill b) Leclerc y Fischler	92
Figura 52	Número máximo de pasos por escalón 300, reducciones de paso: a) 75% del actual b) 50% del actual c) 25% del actual	96
Figura 53	Número máximo de pasos por escalón 200, reducciones de paso: a) 75% del actual b) 50% del actual c) 25% del actual	96
Figura 54	Número máximo de pasos por escalón 100, reducciones de paso: a) 75% del actual b) 50% del actual c) 25% del actual	96
Figura 55	Diagrama de flujo del algoritmo S.A.....	99
Figura 56	Criterio Metrópolis	100
Figura 57	Evolución del coste y la temperatura en el algoritmo Simulated Annealing.....	105

Figura 58	Efectos del número de pasos por escalón en el algoritmo SA.....	106
Figura 59	Efectos de la longitud de paso inicial en la temperatura inicial	106
Figura 60	Representación del coste y la temperatura para un paso inicial igual al 4% $\min(\text{Rango}, \text{RangoY})$	107
Figura 61	Efectos del coeficiente de aceptación en la determinación de la temperatura inicial	108
Figura 62	Representación del coste y la temperatura para un paso inicial igual al 4% $\min(\text{RangoX}, \text{RangoY})$ y ley de enfriamiento del 87%	109
Tabla 63	Criterios de clasificación de los modelos según las propiedades de la figura	116
Figura 64	Asignación de coordenadas según el algoritmo de Lamb y Bandopadhy	119
Figura 65	Plano oblicuo en un modelo simétrico.....	119
Figura 66	Plano oblicuo en un modelo asimétrico	120
Figura 67	Dependencia de los ángulos en una proyección	121
Figura 68	Relación entre ángulo y las coordenadas “Z”	122
Figura 69	Determinación de coordenadas “Z” de una imagen mediante inflado axonométrico	123
Figura 70	Efectos de la asignación de coordenada inicial arbitraria en el inflado axonométrico	123
Figura 71	Selección de la orientación del ángulo axonométrico	124
Figura 72	Inversión de Necker	125
Figura 73	Criterios para la selección en la orientación del ángulo axonométrico	126
Figura 74	Reconstrucción obtenida por inflado axonométrico a) Imagen original b) Objeto reconstruido.....	127
Figura 75	Primera transformación a normalón evaluable a) Imagen de tipología cuasi-normalón b) Imagen de tipología normalón.....	128
Figura 76	Recorrido en análisis de inflado axonométrico	128
Figura 77	Primera transformación a normalón no evaluable a) Imagen de tipología cuasi-normalón b) Imagen de tipología normalón.....	129
Figura 78	Segunda transformación a normalón	130
Figura 79	Indeterminación del inflado axonométrico ante aristas colineales ...	131
Figura 80	Inflado axonométrico en imagen con arista colineales	132
Figura 81	Inconsistencias en el paralelismo de aristas.....	133
Figura 82	Primera transformación a normalón de una imagen con inconsistencias de paralelismo.....	133
Figura 83	Errónea reconstrucción de un modelo con inconsistencias de paralelismo.....	134
Figura 84	Reconstrucción de un modelo con inconsistencias de paralelismo.....	134
Figura 85	Aplicación del método de inflado axonométrico. a) Imagen de tipología cuasi-normalón b) Imagen tras segunda transformación.	135

Tabla	86	Vértices analizados por inflado axonométrico en un modelo cuasi-normal on	136
Figura	87	Relación entre niveles y tipologías de vértices	138
Figura	88	Inflado de un modelo y su inversión de Necker	139
Tabla	89	Clasificación por niveles de tipologías de vértices	139
Figura	90	Aristas predominantes y giro de los vértices	140
Figura	91	Rotación de un objeto normalón para orientar la arista predominante. a) Imagen con orientación original b) Imagen girada para detectar niveles en vértices	141
Figura	92	Rotación de un objeto prismático para orientar la arista predominante. a) Imagen con orientación original b) Imagen girada para detectar niveles en vértices	142
Figura	93	Rotación de un objeto piramidal para orientar la arista predominante. a) Imagen con orientación original b) Imagen girada para detectar niveles en vértice	142
Figura	94	Rotación de un objeto indefinido para orientar la arista predominante. a) Imagen con orientación original b) Imagen girada para detectar niveles en vértice	143
Figura	95	Vértices iniciales y finales de aristas en la determinación de ángulos	144
Tabla	96	Reglas analíticas para la determinación de tipologías para arista predominantes con ángulo de 90 grados.....	145
Tabla	97	Reglas analíticas para la determinación de tipologías para arista predominantes con ángulo de 270 grados.....	146
Figura	98	Vértices sin aristas predominantes.....	147
Figura	99	Asunción de proporcionalidad	148
Figura	100	Poliedro con hendidura	149
Figura	101	Poliedro con hendidura. a) inflado por niveles fijos b) inflado por niveles incrementales	151
Figura	102	Inflado previo por niveles y optimización.....	152
Tabla	103	Clasificación por niveles de tipologías de vértices en imágenes con aristas ocultas	153
Figura	104	Selección de aristas que definen la tipología de un vértice	153
Figura	105	Planicidad del polígono directriz de una pirámide	155
Figura	106	Orientación del eje de la pirámide en el inflado previo por plano oblicuo	156
Figura	107	Grafos genéricos/geométricos	161
Figura	108	Definición del camino de un grafo	161
Figura	109	Determinación de las caras en un grafo	163
Figura	110	Contraejemplo del algoritmo de detección de caras propuesto por Leclerc	164
Figura	111	Circuitos fundamentales en un modelo prismático.....	165
Figura	112	Figura Base	167
Figura	113	Circuitos fundamentales	167
Figura	114	Combinación de circuitos fundamentales	167
Figura	115	Circuitos de aristas cerrados y no interceptados entre sí en la proyección de un cubo	168

Figura	116	Aplicación de las ecuaciones de rango máximo en un objeto de calderería	171
Figura	117	Eliminación de caras potenciales no plausibles	173
Figura	118	Definición de planta superior en una imagen	176
Figura	119	Estilos de las aristas de la cara de planta superior de la imagen.....	177
Figura	120	Estilos de las aristas del contorno aparente	178
Figura	121	Definición de caras visibles del modelo	178
Tabla	122	Permutaciones para modelo de “n” caras	179
Tabla	123	Permutaciones en un modelo de cuatro caras	180
Tabla	124	Configuraciones a evaluar del modelo de cuatro caras	181
Figura	125	Segmentación de aristas parcialmente ocultas.....	183
Figura	126	Representación en “explosión” de las caras identificadas en una imagen 2D de un poliedro	185
Figura	127	Barridos horizontales en una imagen 2D de un poliedro.....	186
Tabla	128	Obtención de aristas pertenecientes al contorno aparente	187
Figura	129	Adición de barridos horizontales	188
Tabla	130	Modelos de tipología normalón propuestos por Lamb y Bandopadhay	192
Tabla	131	Modelos de tipología normalón propuestos por Marill	193
Tabla	132	Modelos de tipología normalón propuestos por Lipson y Shpitalni	193
Tabla	133	Modelos de tipología normalón propuestos en esta tesis	194
Figura	134	Modelo de tipología cuasi- normalón propuesto por Lamb y Bandopadhay	195
Figura	135	Modelo de tipología cuasi-normalón propuestos por Leclerc y Fischler	195
Tabla	136	Modelos de tipología cuasi- normalón propuestos por Lipson y Shpitalni	195
Tabla	137a	Modelos de tipología cuasi-normalón propuestos en esta tesis	196
Tabla	137b	Modelos de tipología cuasi-normalón propuestos en esta tesis	197
Tabla	137c	Modelos de tipología cuasi-normalón propuestos en esta tesis	198
Figura	138	Error en la detección de direcciones principales	199
Figura	139	Efectos del criterio de perpendicularidad	200
Figura	140	Combinación de ángulos en los que el inflado previo axonométrico no puede ser aplicado.....	201
Figura	141	Modelo de tipología cuasi-normalón no reconstruible por inflado previo axonométrico	203
Tabla	142	Modelos piramidales propuestos en esta tesis	204
Figura	143	Reconstrucción de un modelo piramidal con inflado previo por plano oblicuo	205
Figura	144	Modelos piramidales detectados con error en la definición de su tipología.....	205
Figura	145	Reconstrucción de un modelo prismático con inflado previo por niveles.....	206
Tabla	146	Modelos prismáticos convexos propuestos en esta tesis	207
Figura	147	Modelo prismático con tipología en vértices simétricas y correlativas.....	207
Figura	148	Reconstrucción de un modelo prismático. a) sin refinado por planicidad de caras b) con refinado por planicidad de caras	208

Tabla	149	Modeos prismáticos cóncavos propuestos en esta tesis.....	209
Tabla	150	Modelos definidos erróneamente de tipología prismática	210
Figura	151	Modelo de naturaleza indefinida propuestos por Lamb y Bandopadhyay	211
Tabla	152	Modelos de tipología indefinida propuestos en esta tesis.....	211
Tabla	153	Modelos de tipología indefinida propuestos por Marill	212
Tabla	154	Modelos de tipología indefinida propuestos por Lipson y Shpitalni	212
Figura	155	Reconstrucción de modelos de tipología indefinida con predominio de aristas paralelas a tres direcciones principales a) Imagen de partida b)Modelo obtenido por primera optimización c) Modelo obtenido por segunda optimización.....	217
Figura	156	Reconstrucción de modelos de tipología indefinida sin predominio de aristas paralelas a tres direcciones principales a) Imagen de partida b)Modelo obtenido por primera optimización c) Modelo obtenido por segunda optimización.....	218
Figura	157	Modelos poliédricos regulares	219
Figura	158	Reconstrucción de modelos poliédricos regulares.....	220
Figura	159	Modificación para la eliminación de circuitos no plausibles.....	223
Figura	160	Dos circuitos de menor número de aristas	224
Figura	161	Error en la detección de caras en una imagen.....	224
Figura	162	Caras detectadas en el modelo 126	225
Figura	163	Detección de caras en un modelo sólido.....	225
Tabla	164	Análisis de la identificación de circuitos fundamentales.....	227
Tabla	165	Análisis del resto de circuitos	228
Tabla	166	Análisis de la eliminación de circuitos no plausibles	229
Tabla	167	Análisis de la selección de caras.....	230
Figura	168	Configuraciones válidas en la detección de aristas ocultas	231
Figura	169	Configuración no analizada en la detección de aristas parcialmente ocultas	232
Tabla	170	Modelos no eulerianos	238
Figura	171	Ley de proximidad	239
Tabla	172	Modelos alámbricos	240
Figura	173	Imagen que permite combinación de métodos de inflado previo	241
Figura	174	Optimización mediante gradiente conjugado	265
Figura	175	Interpolación parabólica	266
Figura	176	Estructura de un algoritmo genético	267
Figura	177	Ruleta de un algoritmo genético	268
Tabla	178	Análisis de calidades de un algoritmo genético.....	268
Figura	179	Recombinación de cadenas	270
Figura	180	Mutación de cadenas.....	270
Figura	181	Deducción de los coeficientes de reducción a partir de los ángulos que forman los ejes con sus proyecciones	273
Figura	182	Deducción analítica del ángulo formado por un arista y su correspondiente proyección sobre un plano	275

GLOSARIO DE SÍMBOLOS

SIMBOLOS GENERALES

Bajo este epígrafe hemos incluido aquellos símbolos de uso común en notación matemática que han sido utilizados en el documento de esta tesis intentando aclarar cuantas confusiones pudiesen producirse.

tg (x)	Tangente del ángulo x
cotg (x)	Cotangente del ángulo x
atan (x)	Arco tangente del ángulo x
arcsen (x)	Arco seno del ángulo x
arc cos (x)	Arco coseno del ángulo x
max	Valor máximo del entorno definido
D	Incremento
ln	Logaritmo neperiano
S	Sumatorio de funciones
det	Determinante
s	Desviación estándar
Ñ	Gradiente
H	Matriz hessiana

SIMBOLOS ESPECÍFICOS

Bajo este epígrafe se especifican aquellos símbolos que en la medida de lo posible hemos generalizado para su uso en este documento.

l	Parámetro de ponderación variable
a_j	Coficiente de ponderación j-ésimo
R_j	Regularidad j-ésima
$m_{a,b}(x)$	Función de confianza
a	Valor nominal de referencia de la función de confianza
b	Desviación límite permisible de la función de confianza
x	Desviación angular de dos segmentos de recta en la imagen
X_i	Coordenada "x" del vértice "i" del modelo
Y_i	Coordenada "y" del vértice "i" del modelo
Z_i	Coordenada Z del vértice "i" del modelo
w_i	Coficiente de ponderación de regularidades
u_i	Vector unitario de la arista "i" del modelo
u'_i	Vector unitario del segmento de recta "i" de la imagen
v_j	Vértice j-ésimo del modelo
n_i	Vector unitario normal a la cara "i" del modelo

RESUMEN

Este trabajo tiene por objetivo la reconstrucción automática de modelos geométricos, a partir de la información contenida en una única imagen vectorial y geoméricamente consistente de un objeto poliédrico. Los procesos de optimización son a nuestro entender el camino más prometedor para la reconstrucción, en tanto que pueden simular la manera en que percibe el ser humano. Sin embargo la Reconstrucción Geométrica planteada como proceso de optimización presenta como problema fundamental una función objetivo compleja: con muchos mínimos locales. Los mínimos locales son modelos no válidos, porque no son acordes con la percepción visual humana (no son *psicológicamente plausibles*). Además, el punto de partida del algoritmo (la imagen), constituye un mínimo local.

Nuestro trabajo se orientó inicialmente a implementar un algoritmo de optimización de los que se proclaman capaces de obtener mínimos globales. Sin embargo, llegamos a la conclusión de que ni siquiera dichos algoritmos garantizan el óptimo en el caso de la Reconstrucción Geométrica, porque su comportamiento depende mucho de sus propios parámetros de ajuste y de la naturaleza del modelo a reconstruir.

Es por ello que creemos necesario que los algoritmos de optimización vengan asistidos de estrategias de *inflado tentativo*, para generar modelos iniciales tan próximos como sea posible al óptimo global, es decir, que sean lo más parecidos posible al modelo psicológicamente plausible. En ese camino hemos desarrollado tres estrategias que permiten generar modelos iniciales. Hemos comprobado que cada una de estas estrategias funcionan bien cuando se aplican a modelos de ciertas tipologías, por lo que hemos desarrollado una clasificación específica de poliedros acorde con nuestros fines. Dado que la clasificación está orientada a seleccionar la estrategia de inflado tentativo más conveniente, también hemos desarrollado un algoritmo para detectar el tipo de poliedro automáticamente a partir de la imagen de entrada.

A pesar de los importantes avances en los sistemas CAD, muchos ingenieros y diseñadores todavía siguen prefiriendo el lápiz y el papel en las fases más conceptuales del diseño. Esta preferencia se justifica porque en estas fases el diseñador baraja una colección de ideas incompletas y abstractas sobre lo que el producto diseñado deberá ser, y un conjunto mal definido de requerimientos que el producto deberá cumplir. Sin embargo, los interfaces de los sistemas CAD actuales no permiten una rápida expresión de estas ideas.

En otras palabras, para manejar diseños ambiguos o inconsistentes se usan los croquis y los bocetos. Entendiendo que los croquis permiten expresar formas geométricas sin atarse a los criterios estrictos de la geometría (se trata de dibujos imperfectos o inconsistentes desde el punto de vista geométrico), y los bocetos permiten expresar ideas parciales o inacabados (dibujos *incompletos*). Por ello se suele decir que los croquis y los bocetos constituyen el lenguaje “natural” que los ingenieros y diseñadores usan para sintetizar nuevos diseños. Sin embargo, los sistemas CAD actuales son excesivamente rígidos. Es decir, sólo pueden manejar modelos consistentes y completos, y no incorporan ninguna forma de croquizado que permitan una rápida expresión de ideas ambiguas. Usando una terminología más propia de la informática, también se puede decir que las aplicaciones CAD son incapaces de ayudar a los diseñadores porque no integran “la metodología del croquis” en una “máquina de ideas” más compleja. Por tanto, son ciegos para ver el croquis; es decir, carecen del *interface* necesario para interpretar un croquis ya acabado, o para permitir croquizar directamente en el ordenador. Esta deficiencia de los sistemas CAD fuerza al proyectista a generar el diseño por medio de croquis y bocetos y, posteriormente, reconstruir enteramente el modelo en un sistema CAD una vez que el proceso de croquización ha finalizado. En consecuencia, es preciso que sea el proyectista quien lea el croquis final y guíe al sistema CAD para construir el correspondiente modelo. Es obvio que un sistema que agilice la elaboración de un diseño en estas fases iniciales con efectividad y calidad, debe ser capaz de crear e interpretar bocetos mediante un *interface* adecuado, y dirigir el proceso de diseño [Qin00].

Otro problema importante relacionado con el uso de sistemas CAD es que el papel ha sido, durante muchos años, el soporte principal para los dibujos de ingeniería. Por ello, todavía hoy existen numerosos dibujos de ingeniería en soporte papel. Rescatar estos dibujos es necesario para que el “saber como” (o “know-how”) de las empresas pueda aprovecharse y el tratamiento de toda la documentación pueda unificarse. La migración de estos dibujos en soporte papel a sistemas CAD es difícil, por lo que en algunos casos se renuncia a esa “herencia” y en otros casos el trabajo es subcontratado a empresas especializadas. Esto es debido a que este trabajo supone una serie de tareas aún no automatizadas. O bien se precisa de un delineante que guía al sistema CAD, o en el caso de existir algunas herramientas para ello, tal como es el caso de la vectorización de planos, la tarea requiere del manejo de software por personal especializado [Dori95].

Como consecuencia de todo ello, cada vez está siendo más demandado un sistema que sea capaz de generar modelos CAD a partir de todo tipo de planos de ingeniería. Es más, la demanda abarca tanto dibujos croquizados como delineados, y va desde bocetos con unos pocos rasgos generales del objeto hasta planos con gran variedad de detalles, incluyendo todo tipo de símbolos normalizados.

1.1 Visión por ordenador y reconstrucción geométrica.

La Reconstrucción Geométrica constituye una pequeña parte del complejo campo de la *Visión Artificial* o *Visión por Ordenador* que se puede definir de forma breve, aunque aproximada, como el conjunto de todas las herramientas y estrategias empleadas para hacer “ver” a los ordenadores. Por tanto, su objetivo es dar métodos para que los ordenadores puedan extraer información de las imágenes. Dependiendo de las características de las imágenes y del tipo de información buscada, se distinguen diferentes ámbitos de actuación (figura 1).

Un primer gran ámbito de la visión por ordenador lo constituye la determinación de formas y la percepción de movimiento cuando se parte de *escenas reales*. Es decir, cuando tenemos que interpretar fotografías, o nos enfrentamos al reto de ayudar a los robots a interactuar con el mundo “real” y en tiempo “real”. Para el ordenador, el mundo real es el que una o varias cámaras captan, a través de la identificación de los objetos presentes en la escena y de la determinación de como se mueven (o donde y

como están colocados). Se puede distinguir entre escenas fijas y escenas en movimiento, pero la distinción no es importante para la presente tesis.

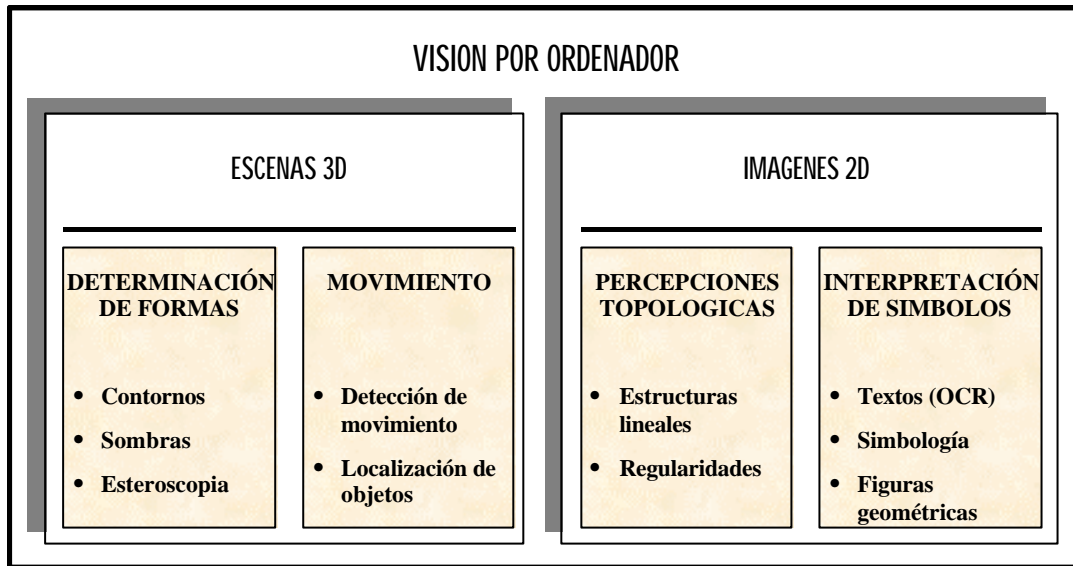


Figura 1. Campos de acción de la visión por ordenador

Debemos destacar que, en general, la identificación de los objetos no incluye una descripción detallada sobre su geometría. En otras palabras, el problema es determinar cual es el objeto, no saber su geometría exacta. Además, la estrategia de capturar la geometría de todos los objetos presentes en la escena donde un robot se mueve sería muy ineficiente, porque de los robots se espera que actúen como hacen los humanos: consiguiendo la mínima información necesaria para evitar colisiones con objetos y sin alcanzar un conocimiento detallado de su geometría.

El segundo gran ámbito es el de las imágenes 2D. Para situar el entorno de la presente tesis es conveniente dividir el problema de visión por ordenador de imágenes 2D en dos casos diferentes. En primer lugar está el tratamiento de imágenes. El objetivo es *filtrar* las imágenes obtenidas; bien para describirlas de una forma compacta o bien para destacar algunas partes de la información presente en ellas. Las imágenes pueden ser obtenidas a través de toda clase de cámaras, escaners y sensores, porque se parte de aceptar que el modo en que se obtiene la imagen (es decir, la relación entre el aparato que obtiene la imagen y el mundo real) carece de importancia. En segundo lugar, podemos encontrar las técnicas encaminadas a añadir las capacidades que los

ordenadores necesitan para “leer” las imágenes de la forma en que lo hacen los humanos. Es decir, que el ordenador debe ser capaz de “interpretar” símbolos y signos, e “interpretar” imágenes 2D; extrayendo la misma información que puede extraer un ser humano. La denominación más acuñada y establecida para este último problema es Reconstrucción (o más precisamente “Reconstrucción Geométrica”), e implica la *determinación de relaciones geométricas y topológicas de todas las partes atómicas de un objeto a partir de una o varias imágenes del mismo*.

Por tanto, el campo del reconocimiento automático de modelos geométricos en 3D implícitamente contenidos en figuras geométricas en 2D es conocido como “Reconstrucción Geométrica”.

La *reconstrucción de dibujos lineales* (o reconstrucción de dibujos técnicos) se entiende algunas veces como sinónimo de la Reconstrucción Geométrica. En otros casos se establece una diferencia consistente en entender que la reconstrucción de dibujos lineales es un caso particular; aquel en el que la imagen sólo contiene entidades geométricas simples, y no contiene símbolos complejos.

En relación con el estado de la cuestión, podemos adelantar que la mayoría de los trabajos de reconstrucción conocidos hasta la fecha se encuentran en fase experimental. Lo que equivale a decir que son desarrollos académicos, carentes de las características necesarias para ser usados como herramientas cómodas y robustas en los procesos que requieren descripciones rigurosas de objetos. En efecto, las aplicaciones existentes tienen importantes limitaciones relacionadas con la tipología de los objetos que son capaces de reconstruir. También tienen las dificultades de uso asociados a cualquier herramienta informática carente de robustez. Además las aplicaciones de reconstrucción geométrica sólo funcionan automáticamente en algunos casos sencillos. Por tanto, pueden interpretar (con algunos errores) toda clase de objetos poliédricos e incluso algunas de las primitivas 3D más usuales (como cilindros, esferas, etc.). Pero los procesos pasan a ser semiautomáticos cuando aumenta la complejidad de las imágenes a reconstruir. Y no son capaces de reconstruir objetos muy complejos, ni siquiera con ayuda del usuario.

Como ya se ha dicho, el objetivo de la visión por ordenador en el ámbito del Diseño Asistido por Ordenador es claro, y se resume en la opinión, muy extendida, de que “los sistemas CAD basados en el modelado de sólidos no tienen el interface correcto” [Cugi91]. Por ello, el propósito final es conseguir una máquina para idear, que

ayude a los diseñadores en las etapas iniciales y en los pasos más conceptuales del proceso de diseño.

En términos de *lenguaje* se puede expresar el problema diciendo que en la actualidad los sistemas CAD tienen salidas no secuenciales, pero sólo aceptan entradas secuenciales. Esto es una consecuencia directa del estado actual de la evolución en Gráficos por Ordenador, que es un campo restringido a su vez por la naturaleza secuencial de los lenguajes de algoritmos usados para tareas de programación. En consecuencia, desde el punto de vista del programador, se considera conveniente que el usuario deba describir el objeto que pretende modelar de forma secuencial y ordenada. Por el contrario, ya se ha dicho que el diseño conceptual no puede ser secuencial. De hecho se ha basado siempre en la utilización de un lenguaje gráfico, es decir, no secuencial. En otras palabras los ordenadores deben interactuar con los diseñadores de una manera amistosa y desordenada durante todo el ciclo de diseño. Consecuentemente, debe definirse un lenguaje nuevo, o debe adaptarse un lenguaje ya existente para mejorar la comunicación actual entre diseñadores y sistemas CAD. Esto significa que *los sistemas CAD deben “comunicarse” en la manera no secuencial en la que los diseñadores lo hacen.*

1.2 Ámbito de la Reconstrucción Geométrica.

Se ha justificado que los diseñadores e ingenieros estamos familiarizados con el uso de planos de ingeniería, que nos permiten establecer una comunicación técnica entre seres humanos (hombre/hombre). En dichos planos es habitual la representación en dos dimensiones de objetos tridimensionales. En consecuencia, parece lógico mantener ese lenguaje para la comunicación entre seres humanos y máquinas (hombre/máquina).

La utilidad de la comunicación hombre/máquina es obvia en el Diseño Asistido por Ordenador. Y también es evidente que conseguir sistemas capaces de interpretar la información técnica contenida en los planos de ingeniería mejoraría la comunicación entre los diseñadores y los sistemas CAD.

Pero además debemos recordar que, en el caso más general, la información técnica contenida en el plano que servirá de entrada a una aplicación de Diseño Asistido por Ordenador será una mezcla de figuras geométricas en 2D y símbolos normalizados. La razón es que los planos de ingeniería incluyen, además de una descripción

geométrica de los elementos que en ellos se representan, información adicional en forma textual (cifras de cota, leyendas, etc.) y simbólica (líneas de acotación, líneas auxiliares de cota, patrones de sombreado, etc.).

Por tanto, el proceso de información de los planos técnicos puede tener hasta tres tareas diferenciadas [Gomi99(1)]:

- Obtención de entidades geométricas.
- Discriminación e interpretación de textos.
- Discriminación e interpretación de entidades simbólicas.

En principio hay que notar que algunas de estas tareas pueden estar embebidas en el software de ciertos periféricos (como los “OCR” que identifican textos en imágenes obtenidas por escaners; o las tabletas digitalizadoras que identifican ciertos movimientos y los interpretan como lenguajes gestuales). Pero, lo que se quiere destacar no es que estas tres tareas pueden ser necesarias. Ni, por supuesto, se pretende decir que sean tres tareas no estudiadas que se deban desarrollar *ex novo*. Lo único que se pretende constatar es que estas tres tareas están relacionadas, y deberán resolverse conjuntamente para obtener una interpretación semántica coherente de los planos de ingeniería más complejos.

Probablemente, el ejemplo más claro de esta relación se presente en las cotas, ya que en ellas debemos distinguir entre la cifra de cota (texto), la línea de cota y líneas auxiliares de cota (entidades simbólicas) y la parte del dibujo que se está acotando (entidad geométrica). Por tanto, para abarcar la semántica completa de una acotación, en primer lugar debemos identificar las líneas de cota y asociarlas con sus correspondientes cifras de cota y con sus líneas auxiliares de cota para, más tarde, relacionar este conjunto con la entidad geométrica que se está acotando y de este modo poder interpretar la cota como una restricción geométrica que se aplica a la forma geométrica acotada. Una dificultad añadida a este problema es que para poder reconocer todo este tipo de características será necesario identificar previamente el estándar de dibujo que se está utilizando; dado que las simbologías de las diferentes normas (ISO, ANSI o UNE) pueden presentar diferencias notables.

Dotar a un sistema informático de la capacidad de leer dibujos de ingeniería, es decir, de interpretar la información geométrica y simbólica que en ellos se representa,

supone por tanto que dicho sistema informático sea capaz de realizar las tareas de vectorización (en el caso de partir de bocetos contenidos en soporte papel), reconstrucción geométrica 2D, reconstrucción geométrica 3D y refinado del modelo. Las tareas descritas se ilustran en la figura 2.

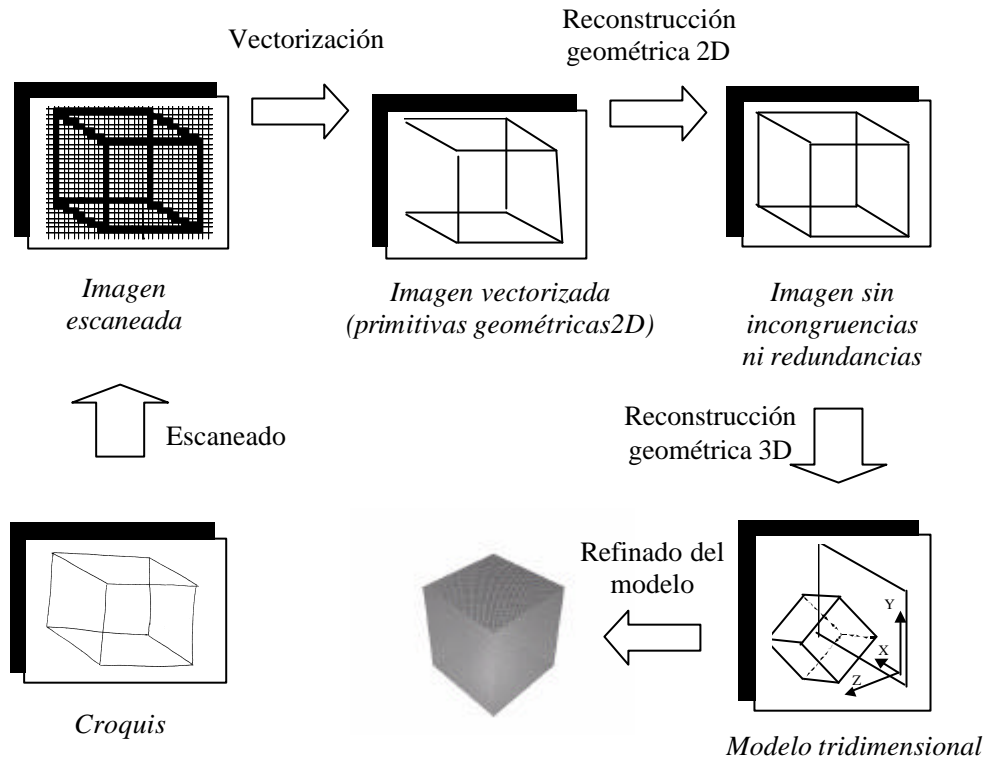


Figura 2. Fases de la reconstrucción de dibujos de ingeniería

1.2.1 Vectorización.

En la actualidad, las aplicaciones de manipulación de imágenes están orientadas a trabajar con las imágenes almacenadas en formato matricial, o mapa de píxeles, mientras que las aplicaciones basadas en la manipulación de geometría (que básicamente son las aplicaciones CAD) utilizan el almacenamiento vectorial (es decir, elementos primitivos tales como puntos, líneas y curvas).

La inmensa mayoría de los periféricos de representación tienen un funcionamiento orientado al almacenamiento en mapa de píxeles. Por ejemplo, los escaners y cámaras fotográficas digitales permiten adquirir imágenes utilizando un

almacenamiento de la imagen matricial. Por ello, las aplicaciones CAD suelen incorporar algunas capacidades de tratamiento de información matricial, de forma que, la imagen matricial resultante de escanear un plano delineado por medios tradicionales puede ser manipulada, e incluso vuelta a imprimir, por una aplicación CAD. Pero dicha aplicación no podrá utilizar ningún recurso que requiera conocer las propiedades geométricas de la imagen de partida.

La vectorización permite convertir una imagen en mapa de bits en figuras geométricas. Es decir, permite identificar puntos, rectas y curvas en una imagen dada en mapa de pixels. Es un problema que no está totalmente resuelto en la actualidad; a pesar de que se han dedicado muchos esfuerzos, dada la indudable importancia práctica que tiene. De hecho, existen en el mercado diferentes programas de vectorización. Una gran parte de ellos están enfocados a simplificar el tratamiento de las representaciones artísticas, otros están orientados al GIS. Y en lo que respecta a los orientados a dibujos de ingeniería estos aún necesitan de un grado de iteración notable por parte del usuario [Gomi98].

1.2.2 Reconstrucción geométrica 2D.

En esta fase se enmarcan las técnicas de filtrado de aquellos planos, croquis o bocetos generados por el diseñador que han podido adquirirse tras escanear las figuras en soporte papel, o mediante la delineación con algún software apropiado, con hardware específico (tales como pantallas LCD e interfaces caligráficas), o con ambos [Kuzo99].

Se contemplan todas las técnicas que se pueden utilizar para convertir los bocetos en figuras geoméricamente consistentes, que sirvan a los algoritmos de reconstrucción para generar modelos tridimensionales.

Algunas de las técnicas que se utilizan se toman de otros campos afines; como el tratamiento de imágenes o la detección automática de caracteres. De hecho, parte de los aspectos que se consideran en esta fase están tradicionalmente contemplados en muchos sistemas de vectorización. Es el caso de la segmentación de líneas, que da como resultado en el proceso de reconstrucción la interpretación de multitud de aristas cuando en realidad sólo habría una. El de la detección y eliminación de las líneas provocadas por los pliegues del papel; o las sombras causadas por su envejecimiento.

Pero la reconstrucción geométrica 2D no sólo debe eliminar las imperfecciones provocadas por el proceso de vectorización o los errores de la delineación por ordenador. Su tarea principal (que es mucho más compleja que la anterior), es filtrar todas las imperfecciones y errores que contienen los bocetos de partida y que son consustanciales a su propia naturaleza de dibujos imprecisos e incompletos. El problema se puede ilustrar con el caso de las líneas que se “perciben” paralelas, aunque geoméricamente se constate que son convergentes; o las líneas cuyos extremos no definen un vértice, porque determinan puntos muy próximos pero no completamente coincidentes. Se trata por tanto de un filtrado basado en criterios de psicología de la percepción, de los cuales apenas existen antecedentes en los ámbitos de la informática y la ingeniería. En este campo se pueden citar, no obstante, trabajos como los de De Bonet [DeBo95].

1.2.3 Reconstrucción geométrica 3D.

En esta fase se trata de obtener un modelo geométrico tridimensional a partir de una o más figuras vectoriales, utilizando la información sobre geometría tridimensional de los objetos que está implícita o explícitamente contenida en las figuras de partida.

La información explícita es principalmente de naturaleza geométrica. Y el vínculo entre la figura 2D y el modelo 3D se establece partiendo de la suposición de que la figura es algún tipo de proyección del modelo. Por tanto, son las leyes que rigen los distintos sistemas de proyección las que permiten utilizar la información explícita para reconstruir el modelo proyectado.

La información implícita es casi exclusivamente perceptual. Por ende, las leyes que gobiernan la percepción visual sirven para aprovechar la información implícita para reconstruir el modelo percibido.

Los distintos métodos ensayados hasta la fecha y las aportaciones particulares de este trabajo serán ampliamente expuestas en el desarrollo de esta tesis.

1.2.4 Refinado del modelo.

En la fase de refinado incluimos todas las técnicas precisas para obtener un modelo que sea acorde con todas las especificaciones dadas en la imagen. Dichas

técnicas deben mejorar al modelo obtenido mediante la reconstrucción geométrica 3D. Dos son las especificaciones fundamentales que debe aportar este proceso:

- Como se mostrará en el desarrollo de esta tesis, mediante la Reconstrucción geométrica 3D es posible conseguir modelos que sean acordes con la imagen de partida desde el punto de vista topológico. Sin embargo, en algunas ocasiones la representación del modelo no resulta ser perfecta y se distinguen claras distorsiones tales como que los vértices de una cara no resulten coplanarios. El refinado del modelo tiene como uno de sus objetivos el paliar estas deficiencias para conseguir modelos perfectos.
- Pero además, y como ya ha sido comentado, un dibujo de ingeniería lleva asociados un conjunto de información que es expresada por medio de símbolos normalizados. El proceso de refinado debe restringir el modelo obtenido en la fase de reconstrucción geométrica 3D, exigiendo que el modelo final cumpla las restricciones especificadas por dichos símbolos. El ejemplo más claro de estos símbolos normalizados lo constituyen las cotas que expresan las dimensiones reales del modelo final.

1.3 Clasificación de los métodos de reconstrucción.

El problema de la reconstrucción es inabordable en su totalidad, por lo que todas las aproximaciones realizadas hasta la fecha son parciales. Para delimitar los límites del problema a resolver, los diferentes autores han planteado distintas hipótesis de trabajo. A groso modo, estas hipótesis se pueden clasificar atendiendo a tres criterios:

- Las que limitan los tipos (o la naturaleza) de las imágenes de partida.
- Las que limitan la naturaleza de los modelos generados.
- Las que limitan los procedimientos de reconstrucción a emplear.

1.3.1 Número de vistas.

Los métodos de reconstrucción pueden clasificarse en función de los tipos de imágenes que constituyen la información de entrada del proceso, atendiendo a diferentes

critérios. Así por ejemplo, si bien es cierto que la mayoría de estos métodos de reconstrucción parten de figuras delineadas que deben verificar el estricto cumplimiento de las reglas de la geometría, existen otras estrategias de reconstrucción que por la metodología empleada permiten ciertas imperfecciones del dibujo, e incluso en ocasiones, partir de un croquis, es decir, que de alguna manera integran la fase anteriormente descrita como reconstrucción geométrica 2D.

Otros criterios que deben ser tenidos en cuenta en esta clasificación pueden ser la necesidad de no representar en la imagen las aristas ocultas del modelo, o la propia naturaleza de las primitivas que componen la imagen, diferenciando en este caso entre modelos poliédricos y modelos que presentan superficies delimitadas por curvas.

Pero la clasificación más extendida hasta el momento distingue entre dos grandes grupos (figura 3).

- Métodos de múltiples vistas: cuando la entrada al algoritmo de reconstrucción son dos o más imágenes (que generalmente representan las vistas ortográficas alzado, planta y perfil).
- Métodos de vista única: cuando la entrada en los algoritmos de reconstrucción está compuesta por una sola imagen (que generalmente representan una proyección directa axonométrica).

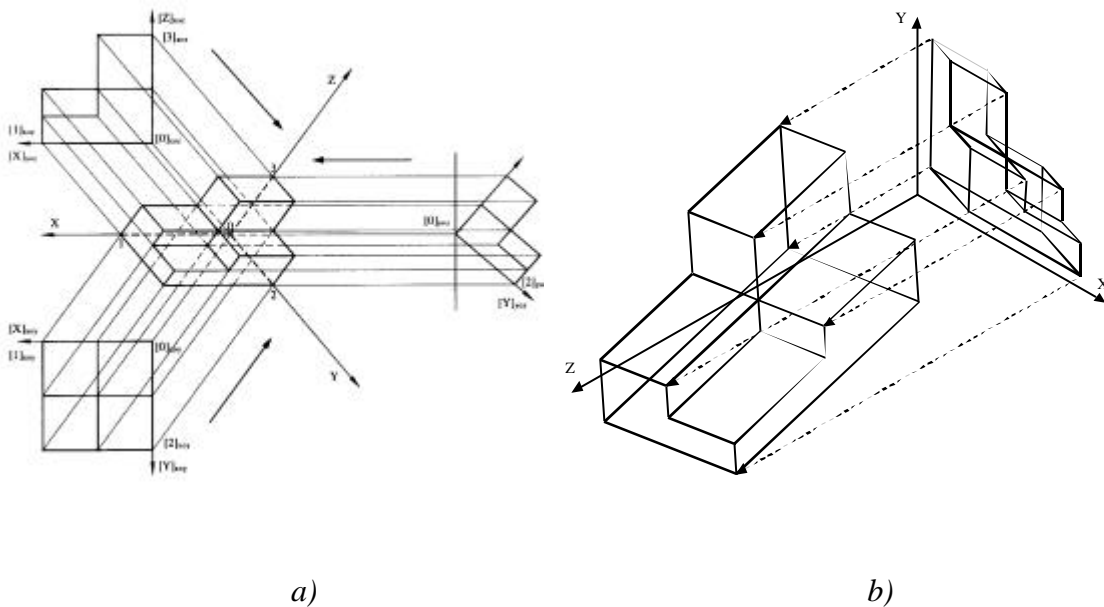


Figura 3. Reconstrucción mediante métodos de
 a) Múltiples vistas b) Vista única

Los métodos de múltiples vistas están actualmente más avanzados, la razón es obvia si comparamos la mayor información geométrica dada por un conjunto de varias vistas frente a las indefiniciones que presenta una única vista. No obstante, ni siquiera los métodos de vistas múltiples aceptan convencionalismos normalizados tales como cortes, vistas particulares, etc.

Hay más limitaciones que se aplican a la imagen de partida. Cabe destacar, por ejemplo, que no parece viable la consideración de métodos de vista única sin tener en cuenta el “*Supuesto de Punto de Vista General*” descrito por Sugihara [Sugi86]. Dicho supuesto admite como limitación adicional que la proyección no debe ser paralela a ninguna cara, ni coplanaria con ningún par de aristas no colineales. Mediante este supuesto se eliminan claros casos de degeneración tales como que una superficie plana sea representada por una arista, que un par de aristas se puedan proyectar sobre una única línea, o que una arista quede proyectada en un punto. En realidad se trata de una reformulación en términos proyectivos de uno de los principios básicos de la percepción visual. No se trata de un caso aislado. Dicho en otras palabras: muchos de los principios de la percepción se han venido introduciendo como requisitos, limitaciones o reglas heurísticas que la imagen de partida debe cumplir.

1.3.2 Modelos generados.

En cuanto a la naturaleza de los modelos generados hay que resaltar que las representaciones comúnmente utilizadas en el problema de reconstrucción son la CSG y la BRep.

La CSG, introducida por primera vez en la reconstrucción por Requicha [Requ92], consiste en la representación de un modelo 3D a partir de una colección de primitivas, una serie de transformaciones y un conjunto de operaciones booleanas. Las primitivas son sólidos simples concretos, tales como cubos, pirámides, cilindros, conos y esferas. Las transformaciones incluyen la traslación, rotación y escalado y se usan para definir la posición, orientación y tamaño arbitrario de las primitivas. Las operaciones booleanas consisten en un conjunto de operaciones de unión, intersección y diferencia (regularizados para producir objetos con interiores no vacíos) y se usan como operadores para generar objetos más complejos.

La composición se puede representar por una estructura jerárquica de datos, denominada *árbol CSG*, cuyas ramas representan las primitivas y cuyos nodos internos quedan etiquetados con las operaciones booleanas regularizadas con sus respectivas transformaciones. La raíz del árbol CSG se identifica como el objeto deseado y recorrer el árbol proporciona un mecanismo de representación o análisis del objeto (figura 4). Este método está conceptualmente próximo a las técnicas que actualmente se usan en ingeniería para el diseño de piezas mecánicas y, a menudo, da una representación compacta del objeto.

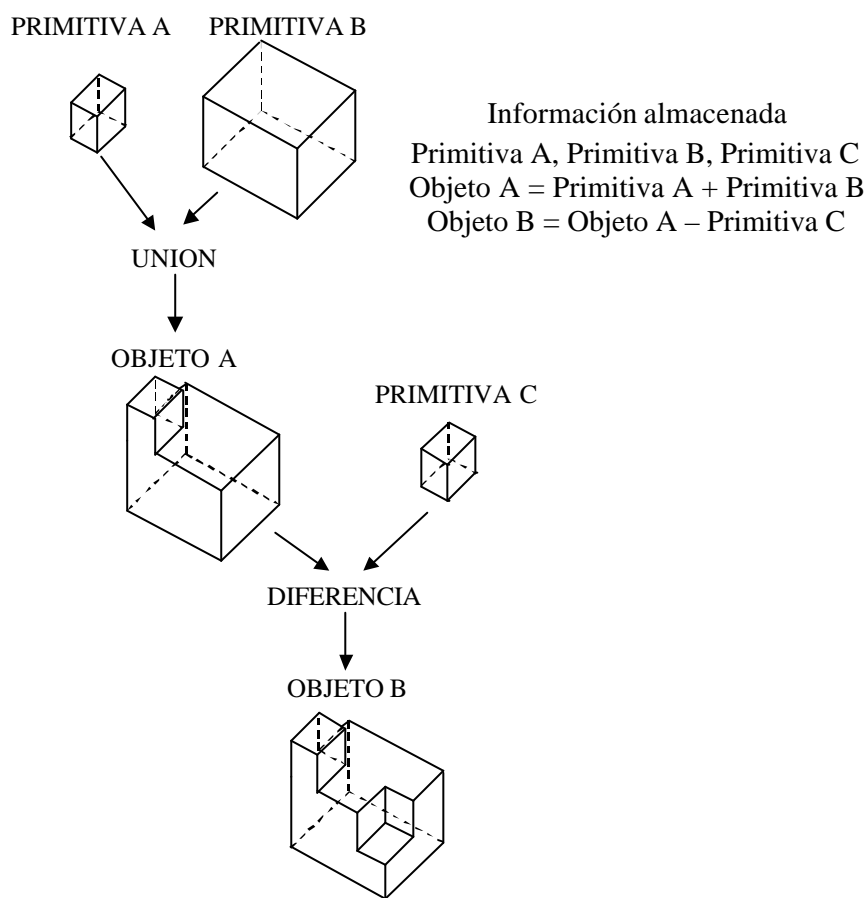
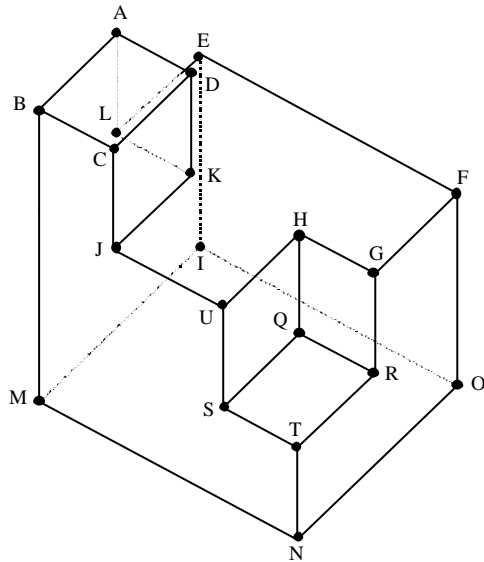


Figura 4. Modelo representado por CSG

La representación BRep (figura 5) ofrece una representación del objeto más general pero no necesariamente más compacta. Un sólido se modela por un número finito de superficies delimitadas, cada una de las cuales se representa por un conjunto de aristas orientadas que la limitan. Cada arista se representa por dos vértices, definidos en algún sistema de coordenadas. Las superficies que se usan normalmente para describir

los objetos sólidos incluyen superficies planas (polígonos), superficies cuadráticas (cilíndricas, cónicas y esféricas) y superficies CAGD (tales como spline, Bezier, NURBS). Las operaciones de construcción en BRep aseguran que el contorno esta bien formado, lo que incluye requisitos tales como que el contorno este cerrado.



Información almacenada

Lista de vértices

A, B, C, D, E, F, G, H, I, J, K, L,
M, N, O, P, Q, R, S, T

Lista de Aristas

AB, AD, AL, BC, BM, DK, DC, LE,
LK, MI, MN, EI, EF, NT, NO, FO,
FG, IO, CJ, JK, JU, US, UH, HQ,
SQ, QR, RG, HG, ST, TR

Figura 5. Modelo representado por BRep alámbrico (sin caras)

1.3.3 Metodología y número de soluciones.

Los distintos procedimientos empleados en la reconstrucción también permiten agrupar las estrategias conocidas hasta el momento atendiendo a diferentes razones. De hecho, los primeros procedimientos de reconstrucción propiciaron una confusión entre los conceptos de Reconstrucción Geométrica y *Restitución*. La Reconstrucción Geométrica no debe ser confundida con la Restitución que implica la identificación de un objeto mediante algún sistema de acoplamiento de plantillas.

De igual forma hay que resaltar que muchos métodos cuyo objetivo real es validar una imagen como proyección de un modelo, han sido también considerados como métodos de reconstrucción.

Los distintos procedimientos empleados en la reconstrucción pueden agruparse en función de la metodología adoptada diferenciándose entre:

- Métodos basados en la formulación de reglas heurísticas.
- Métodos basados en reglas de la Geometría proyectiva.
- Métodos basados en la formulación de reglas de la percepción visual humana.

Pero la clasificación más extendida es aquella que diferencia los distintos métodos de reconstrucción en función del número de soluciones obtenidas a partir de las imágenes de partida. Así, cuando existan casos en los que más de un objeto 3D puede corresponder con las proyecciones 2D de partida como consecuencia de ambigüedades de la imagen, los métodos pueden definirse como de *solución única*, si tras encontrar una solución finalizan su ejecución, o de *múltiple solución* si muestran todas las posibles soluciones al problema planteado (figura 6).

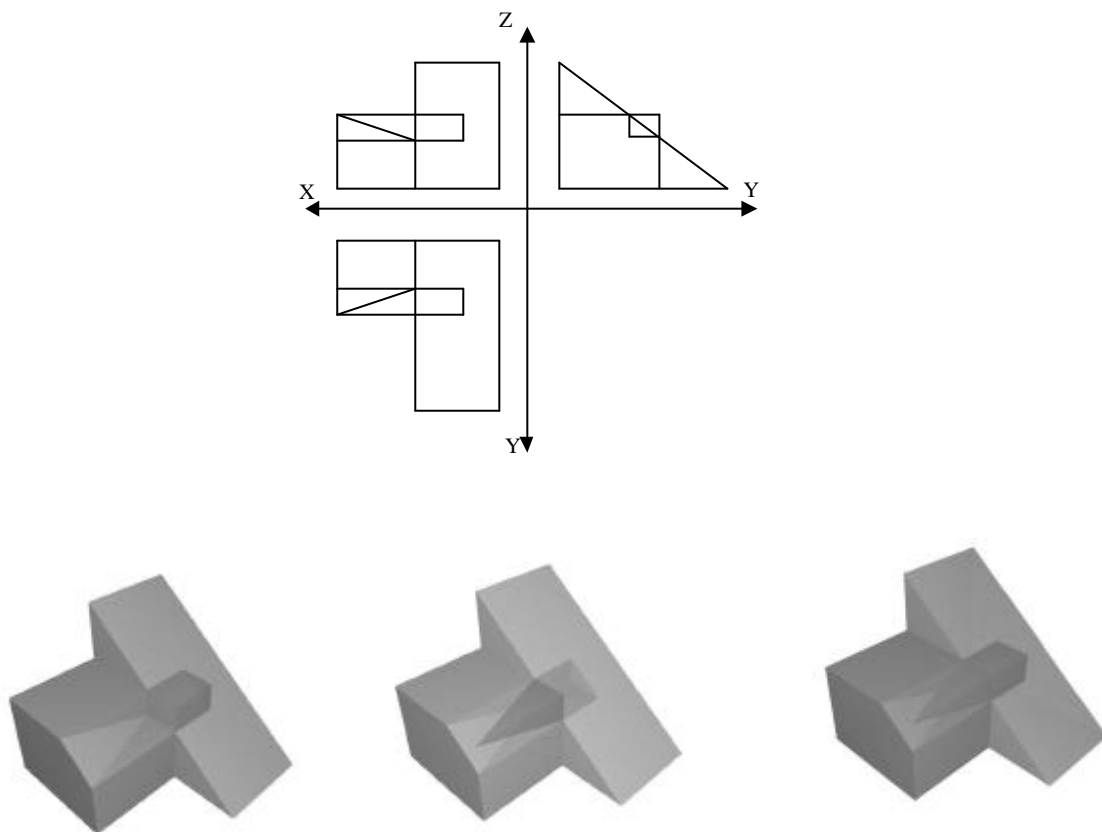


Figura 6. Imagen que puede corresponder a más de un modelo tridimensional

1.4 Objetivos del trabajo.

El objetivo principal de este trabajo, que forma parte del conjunto de desarrollos realizado en el grupo de investigación *Regeo* [REGE00], se orienta a la creación de una herramienta de diseño que permita modelar objetos a partir bocetos, permitiendo integrar las fases iniciales de diseño en el entorno de los sistemas CAD/CAM/CAE.

El sistema permitiría actuar sobre el boceto, sobre el modelo 3D, o sobre ambos, iniciando un proceso de refinamientos sucesivos que no se detiene hasta validar el diseño definitivo.

Dada la amplitud del ámbito de trabajo, puesta de manifiesto en 1.2, esta tesis marca sus objetivos específicos en la fase de reconstrucción geométrica 3D, es decir, en la obtención de manera automática de modelos geométricos, a partir de la información contenida en una imagen vectorial y geoméricamente consistente. Si bien, transcurre en paralelo con otros trabajos de investigación del grupo REGEO que abordan otros aspectos de nuestro objetivo general.

1.5 El programa REFER.

Dada la naturaleza del trabajo propuesto, la validación de las conclusiones requiere una fase experimental, consistente en la implementación de las metodologías y los algoritmos propuestos en una aplicación prototipo (o académica). La aplicación utilizada ha sido REFER, que es el programa que se desarrolló como paso inicial en la estrategia del grupo REGEO de disponer de un “estado del arte práctico”. Es decir, que entendiendo que conocer el estado del arte por medio de documentos “estáticos” no es suficiente para introducirse en un trabajo de desarrollo de herramientas informáticas, el grupo ya había desarrollado una aplicación experimental. La primera versión de REFER se desarrolló con el Proyecto Final de Carrera ‘Desarrollo, implementación y prueba de un algoritmo de reconstrucción de objetos a partir de una representación axonométrica, utilizando técnicas de optimización’ [And00].

Se ha implementado una segunda versión del programa para validar los métodos propuestos en esta tesis y comparar con otros métodos de planteamiento análogo propuestos por otros autores.

REFER ha sido desarrollado utilizando Microsoft’s Developer Studio, para ejecutar en ordenador personal (plataforma PC) y bajo sistema operativo Windows NT o

Windows 98. La programación se ha realizado en Visual C++ y la interface gráfica con el usuario se ha realizado mediante las librerías de win32 y Open GL [Open].

ESTADO DEL ARTE EN LA RECONSTRUCCIÓN GEOMÉTRICA

2

La descripción de objetos tridimensionales en un plano, utilizando proyecciones bidimensionales, se remonta a más de dos mil años. Fue **Monge** el primero que sistematizó y simplificó los métodos existentes, dando lugar al nacimiento de la geometría descriptiva. El problema contrario de cómo reconstruir automáticamente la estructura de un objeto tridimensional (estructura geométrica y topológica) a partir de su proyección, empezó a atraer la atención sólo a finales de los 60, motivado por el desarrollo de los ordenadores digitales.

El primer método de reconstrucción de objetos fue realizado por **Roberts** [Robe65], quién fundamentó su estudio en una serie de modelos predefinidos que mediante la unión de los mismos originaban un modelo cuya proyección era contrastada con la imagen de partida. Puede por tanto considerarse como una aproximación inicial al método de representación por CSG.

La evolución de la Reconstrucción Geométrica puede encontrarse resumida en unas pocas referencias. El libro de Sugihara [Sugi86] es la referencia más completa a la historia inicial de la interpretación automática de dibujos técnicos. Nagendra y Gujar [Nage88] publicaron un resumen de varios artículos que trataban la reconstrucción de objetos tridimensionales a partir de sus vistas 2D. Wang y Grinstein [Wang93] completaron el trabajo, realizando una taxonomía de la reconstrucción de objetos 3D a partir de dibujos lineales de una proyección bidimensional.

Un último resumen del estado actual y las tendencias de la reconstrucción es el trabajo de Company [Comp97], quien muestra los resultados obtenidos mediante los sistemas de reconstrucción de única vista y múltiples vistas, comparando las etapas del diseño conceptual mediante métodos clásicos y CAD.

Los trabajos realizados hasta la fecha sobre Reconstrucción Geométrica a partir de vista única y vistas múltiples han transcurrido paralelamente, si bien, existen claras diferencias en cuanto a los métodos ensayados.

2.1 Análisis de los métodos de vistas múltiples.

Hasta el momento se han desarrollado distintos métodos para obtener modelos a partir de vistas múltiples. Algunos de ellos operan de forma automática, pero la mayoría, requieren mucha participación del usuario. En este último caso, se trata de métodos de reconstrucción asistida más que de reconstrucción automática.

A continuación se exponen de manera agrupada los distintos métodos de reconstrucción de vistas múltiples a partir de las referencias que hemos podido encontrar.

2.1.1 Métodos basados en la identificación de primitivas.

Los métodos basados en la identificación de primitivas deben diferenciarse según el modo de representación del modelo. Los métodos que utilizan la CSG para la representación del modelo tienen en común el asumir que un objeto 3D puede ser construido a partir de cierto conjunto de primitivas (prismas, conos, cilindros y esferas) combinadas según una cierta jerarquía. Las diferencias entre los métodos propuestos se establecen en las estrategias para “extraer” dichas primitivas de las vistas de partida (figura 7).

Aldefeld [Alde83] partiendo de las ideas de Idesawa basadas en formulación analítica, fue el primer autor en desarrollar un método de reconstrucción utilizando la representación por CSG. Se trataba de un algoritmo automático pero que presentaba grandes limitaciones. Posteriormente **Aldefeld y Richter** [Alde84] desarrollando un nuevo algoritmo de reconstrucción en el que, si bien eliminaron parte de las limitaciones del método anterior, precisaba de un alto grado de interacción con el usuario.

De acuerdo con la publicación de Wang y Grinstein [Wang93], **Ho** propuso un algoritmo bastante parecido al de los autores anteriores pero necesitaba menor interacción del usuario y trataba una clase más amplia de objetos de ingeniería.

Ambos presentan como característica principal la interacción con el usuario que debe introducir cada una de las primitivas así como validar en cada uno de los pasos del algoritmo la correcta evolución del problema. Se tratan de programas más propiamente de CAD que de Reconstrucción Geométrica.

En la figura 7 se muestra un cuadro evolutivo de los algoritmos de reconstrucción basados en la identificación de primitivas a partir de vistas múltiples. Todos ellos utilizan la CSG como método de representación del modelo.

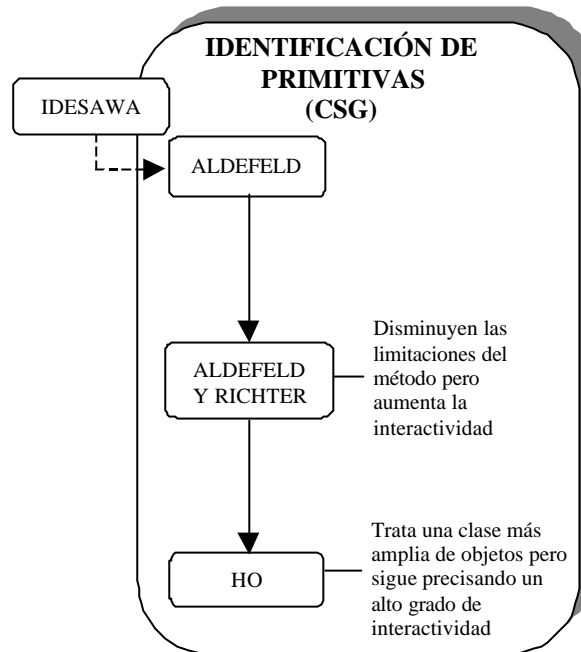


Figura 7. Evolución de los métodos basados en identificación de primitivas a partir de vista múltiples. Representación CSG

En la representación por BRep (figura 8), las estrategias basadas en la identificación de primitivas (vértices, aristas y caras), establecen una comparación y correspondencia entre el modelo y las vistas ortográficas.

Las primeras aportaciones se deben a **Wesley y Markowsky** [Wesl81]. Estos autores identificaban las distintas primitivas en una jerarquía ascendente, es decir, se determinaban en primer lugar los vértices, posteriormente las aristas y por último las caras. El mayor inconveniente es que su método sólo puede ser aplicado a modelos poliédricos.

Diversos trabajos de identificación de primitivas partieron de las ideas propuestas por Wesley y Markowsky, tales como los realizados por **Haralick y Queeney**, o **Richards y Onwubolu**. Cabe destacar las aportaciones de **Sakuari** que mejoró el algoritmo de Wesley y Markowsky permitiendo reconstruir modelos de simetría axial.

Posteriormente **Gu y otros** [Gu86] ampliaron el método de Sakuari para tratar elipses, hipérbolas y curvas regulares de orden superior en proyecciones ortográficas. Otros autores ([Guja89], [Chen92], [Chen96], [Meer93], [Masu97], [Kuo98], y [Shin98]) basados en las ideas de Gu y Sakuari trabajaron sobre distintos métodos de identificación de primitivas con objetivos comunes: ampliar gradualmente los algoritmos previos para poder tratar un mundo de objetos más general, o incluso para intentar obtener información adicional de los dibujos.

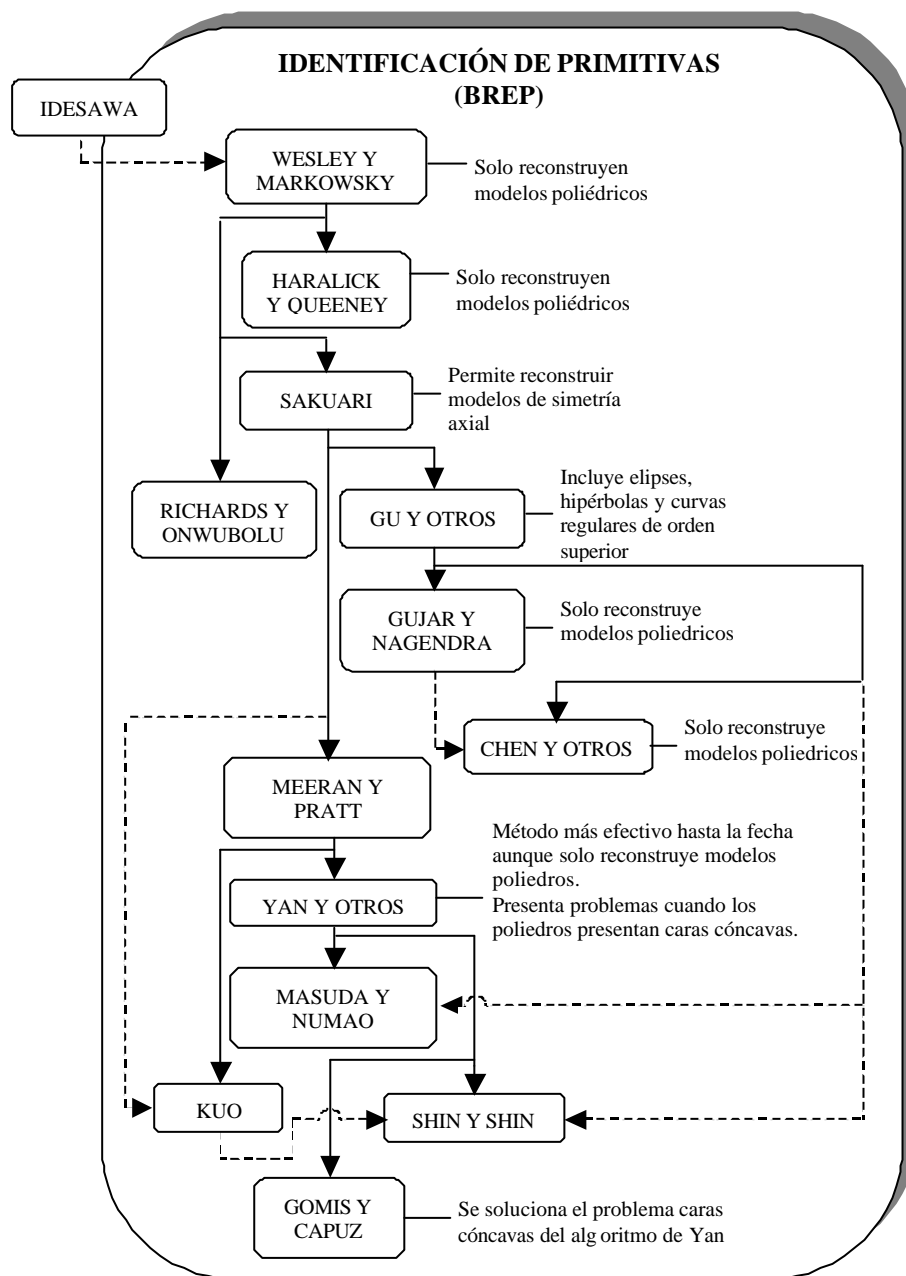


Figura 8. Evolución de los métodos basados en identificación de primitivas a partir de vista múltiples. Representación BRep

Pero los mejores resultados obtenidos en la Reconstrucción Geométrica de Sólidos a partir de vistas múltiples fueron los presentados por **Yan y otros** [Yan94]. Estos autores completaron y sistematizaron uno de los métodos más desarrollados y efectivos hasta la fecha para reconstrucción de poliedros a partir de vistas múltiples. Su método, que presentaba soluciones múltiples, tenía aportaciones fundamentales en la evaluación conjunta de las primitivas vértices, aristas y caras, reduciendo notablemente el tiempo de computo. Sin embargo sólo era capaz de reconstruir modelos poliédricos.

Por su parte el grupo de investigación REGEO, detectó problemas en la reconstrucción de modelos poliédricos en el caso de que estos presentasen caras cóncavas. Desarrollando un algoritmo basado en este mismo método en el que se resolvían dichos problemas [Gomi99(2)].

2.1.2 Métodos de formulación analítica.

Sólo hemos podido encontrar dos trabajos que plantean la Reconstrucción Geométrica a partir de vistas múltiples como solución a un problema de formulación analítica (figura 9).

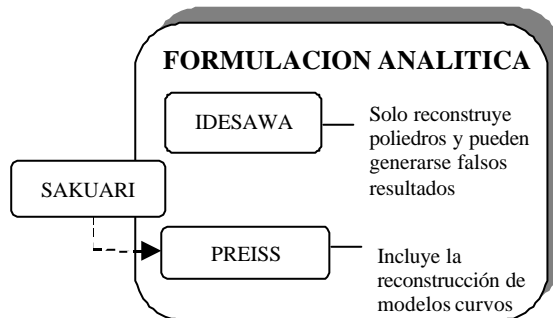


Figura 9. Evolución de los métodos basados en formulación analítica

El primer trabajo de Reconstrucción Geométrica de Sólidos a partir de vistas ortográficas fue propuesto por **Idesawa** [Ides73]. Se trataba de un método matemático que establecía una función analítica que relacionaba un modelo con sus proyecciones. Aplicando la inversa de la función se obtenía una representación BRep del modelo

tridimensional. Sin embargo, su método, que solamente trabajaba con poliedros, podía generar falsos resultados.

Posteriormente **Preiss** [Prei84] utilizó la formulación analítica para definir un conjunto de restricciones que mejoraban los métodos de identificación de primitivas a partir de las ideas propuestas por Sakuari.

2.1.3 Métodos basados en el Teorema de Pohlke.

Un método basado en el teorema de Polhke para la reconstrucción de axonometrías fue presentado por **Gomis y otros** [Gomi96], [Gomi97(1)]. Aunque la reconstrucción final se realizaba mediante métodos de identificación de primitivas similares a los de Meeran y Yan. Una vez reconstruido el modelo alámbrico se determinaban sus las caras. Sus mayores inconvenientes eran la alta interacción del usuario (que debía definir las direcciones de la axonometría) y que sólo podía reconstruir modelos poliédricos eulerianos.

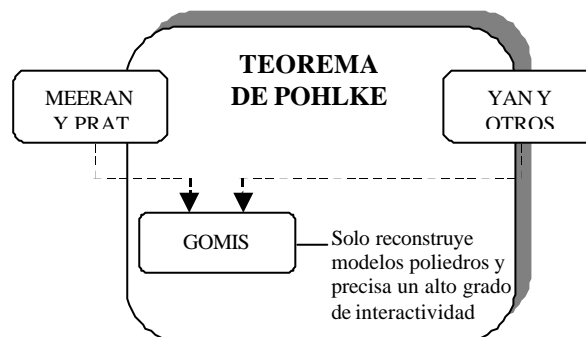


Figura 10. Evolución de los métodos basados en el teorema de Pohlke

En la Figura 11 se muestra un cuadro resumen de la evolución de los distintos métodos de reconstrucción de vistas múltiples

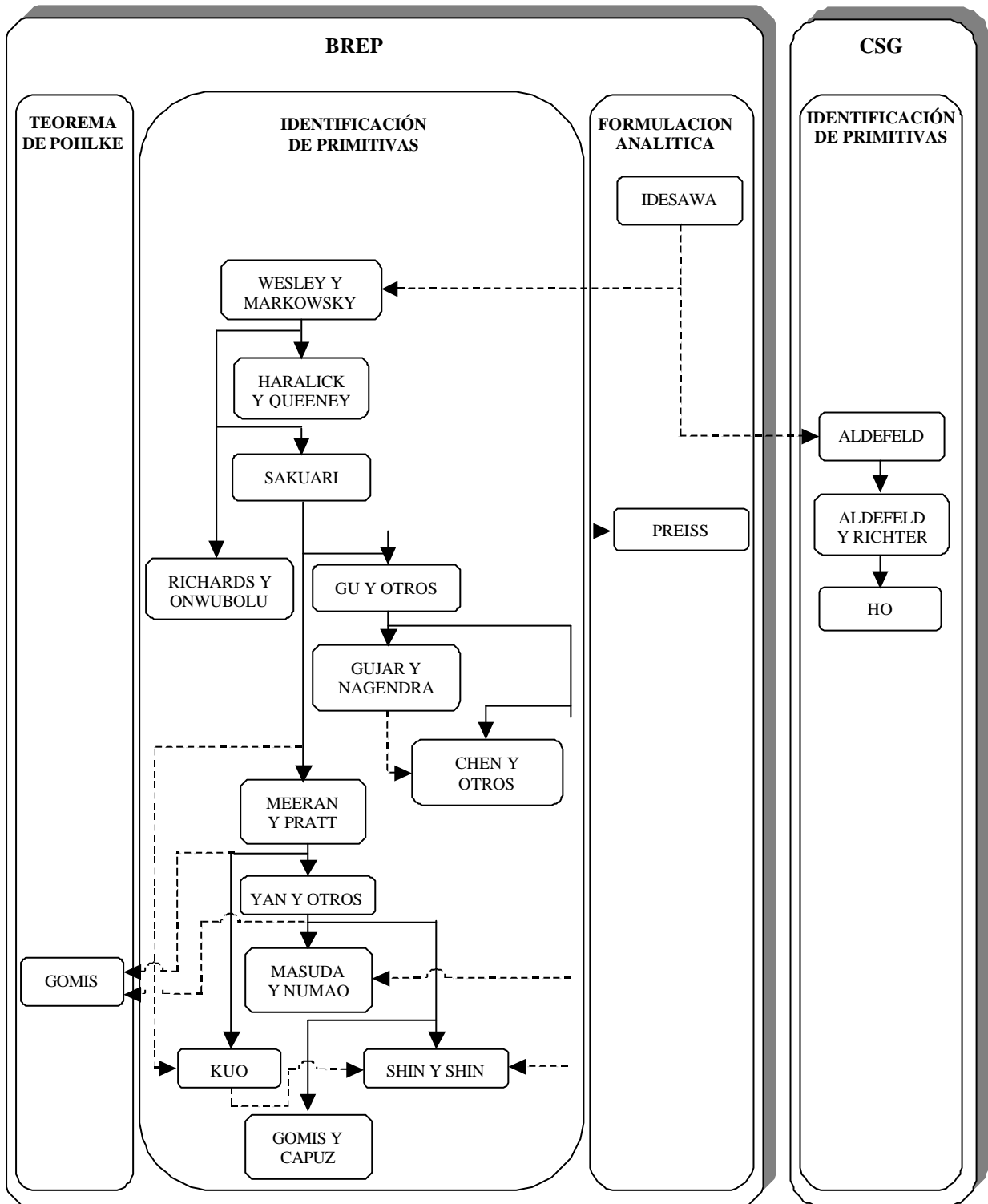


Figura 11. Evolución de los métodos de reconstrucción de vistas múltiples

2.2 Análisis de los métodos de vista única.

En el campo de la reconstrucción con representaciones por CSG, se ha avanzado poco hasta fechas recientes (figura 12). La aportación más importante es la de **Wang y Grinstein** [Wang93] cuyo método se basaba en la identificación de primitivas.

Dichos autores establecieron, a partir de las ideas de Roberts, un algoritmo para resolver reconstrucciones de única vista de objetos poliédricos. Su algoritmo partía del supuesto de vista general utilizando el sistema de etiquetado de Huffman. El algoritmo, basado en la identificación de primitivas, detecta esquinas de poliedros y determina su coordenada Z agrupándolas en distintos niveles según el contacto existente entre las distintas primitivas evaluadas. Sus mayores limitaciones se basan en la naturaleza de los objetos poliédricos que puede manejar, y en que los dibujos lineales deben ser perfectos.

Posteriormente Wang amplió su algoritmo, introduciendo primitivas curvas para trabajar con cilindros, y suavizando las presunciones de rectilinearidad a semirectilinearidad. Sin embargo, el introducir nuevas primitivas produjo ambigüedades en las interpretaciones. En 1993, Wang completó y amplió los algoritmos rectilíneos y semirectilíneos y desarrolló un sistema completo de identificación para reconstrucción por CSG basado en un nuevo sistema de etiquetado que establecía condiciones necesarias y suficientes para que un dibujo lineal representase un objeto 3D válido.

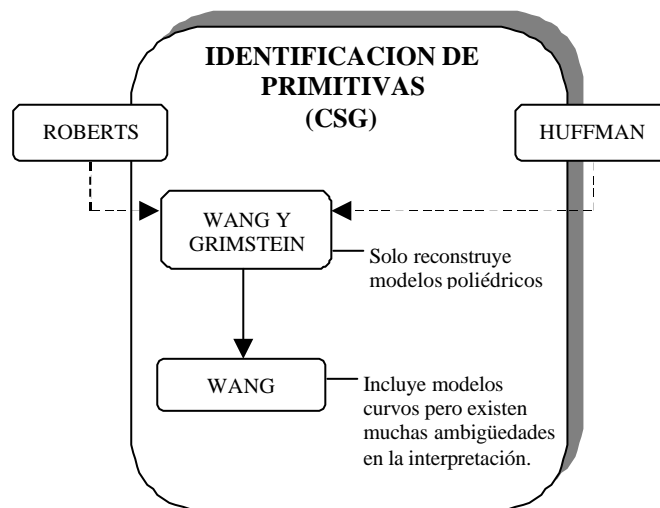


Figura 12. Evolución de los métodos basados en identificación de primitivas a partir de vista única. Representación CSG.

También en la representación de modelos por BRep, ha sido utilizada la identificación de primitivas como método de reconstrucción.

Gomis y otros [Gomi97(2)] desarrollaron un algoritmo de reconstrucción de objetos 3D a partir de la delineación de axonometrías bidimensionales utilizando BRep para la representación del modelo. Su método se basaba en la identificación de caras en la imagen, la reconstrucción del modelo alámbrico y la posterior identificación de caras 3D. Se caracteriza por reconstruir una axonometría cualquiera, resolviendo incluso algunas representaciones que no cumplen el supuesto punto de vista general, si bien el objeto a reconstruir debe ser un poliedro euleriano, es decir, debe ser un poliedro cuya superficie sea bilateral y simplemente conexa. Resalta también su alto grado de interacción con el usuario, que debía confirmar ciertas decisiones tales como si un polígono dado por un conjunto de aristas define el contorno de una cara.

Una de las características más destacables de este algoritmo se derivaba de la libertad por parte del usuario para fijar las escalas axonométricas, rescatándose de este modo, una de las características esenciales del dibujo mediante croquis.

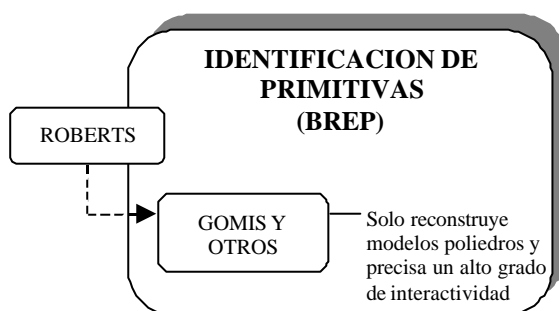


Figura 13. Evolución de los métodos basados en identificación de primitivas a partir de vista única. Representación BRep.

Pero los métodos de reconstrucción más utilizados en la representación de modelos por BRep, están basados fundamentalmente en el etiquetado de vértices, el espacio gradiente, la programación lineal y la formulación de reglas perceptuales.

2.2.1 Métodos basados en el etiquetado de vértices

Los métodos de etiquetado se basan en definir circuitos con todos los enlaces de la figura 2D, que son candidatos a corresponder con aristas del modelo 3D. Se trata más

propiamente de métodos de interpretación que de reconstrucción: ofrecen sólo condiciones necesarias para que un dibujo lineal 2D represente un sólido 3D válido. Además, un objeto lineal que se puede etiquetar adecuadamente no necesariamente representa un sólido 3D verdadero.

Las primeras aproximaciones a los métodos de etiquetado parten de las ideas propuestas por **Guzman** [Guzm]. **Huffman** [Huff71] y paralelamente **Clowes** [Clow71] establecieron por separado el primer esquema de etiquetado válido para poliedros. Clasificaron las esquinas de un cubo por el número de octantes que rodean y tras considerar todas las posibles esquinas desde todos los puntos de vista posibles, observaron que si no existen elementos ocultos, las esquinas de un poliedro podían resumirse en cuatro tipos (figura 14).


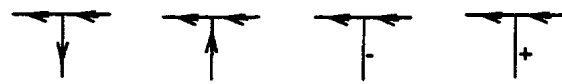
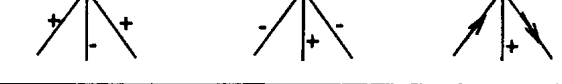
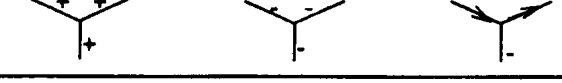
L Type	
T Type	
W Type	
Y Type	

Figura 14. Etiquetado de vértices propuesto por Huffman-Clowes

Las limitaciones más importantes de este método eran que sólo podían reconstruir poliedros y que estos no debían presentar aristas ocultas.

Otros algoritmos de etiquetado surgieron a partir del método de Huffman y Clowes, como los presentados por **Waltz** [Walt] y **Thorpe y Shafer** [Thor83]. También **Marti y otros** [Mart93] desarrollaron un esquema de etiquetado que se caracterizaba por que precisaba distinguir entre aristas visibles y ocultas de la imagen.

De acuerdo con la publicación de Wang y Grinstein [Wang93] **Lee, Haralick y Zhang** trataron de ampliar el etiquetado de Huffman-Clowes a casos más generales, incluyendo dibujos con líneas de trazos que representaban líneas ocultas, superficies sin

espesor y dibujos con objetos curvos. Pero todos los métodos propuestos se basaban en reglas heurísticas, hasta que **Malik** [Mali87] publicó su trabajo.

Su método trata los dibujos lineales de escenas formadas por objetos sólidos regulares opacos delimitados por piezas con superficies suaves, con lo que solucionó el problema de etiquetado general.

Malik desarrolló y demostró un nuevo esquema de etiquetado de dibujos de una clase general de objetos, no necesariamente poligonales. Desde un punto de vista general clasificó todas las uniones posibles en una proyección ortográfica (figura 15).

Sin embargo, una proyección se podía etiquetar de más de una manera, lo que contrasta con la percepción humana en la que la proyección tiene muy pocas interpretaciones 3D.

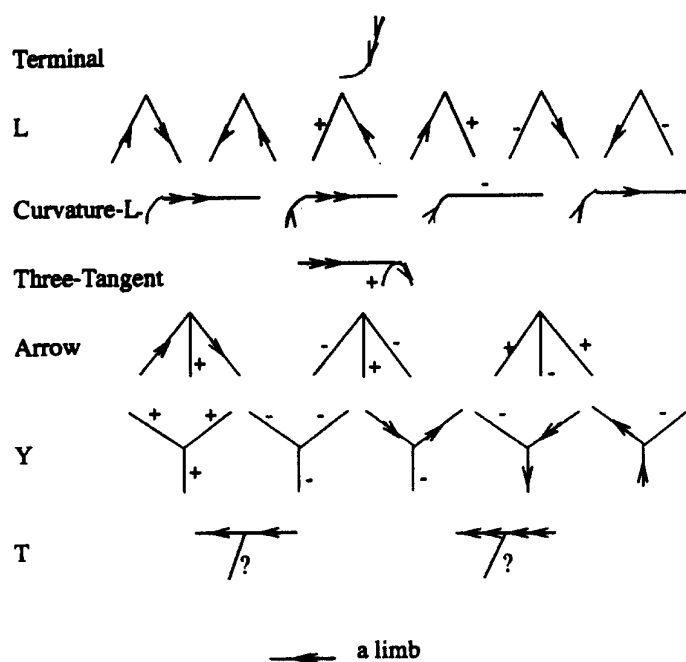


Figura 15. Etiquetado propuesto por Malik

Las últimas aportaciones de los métodos de etiquetado se deben a **Varley y Matin** [Varl00(1)] [Varl00(2)], que aplican dichos métodos para establecer la geometría de las partes ocultas y definir la geometría frontal de un modelo.

En la figura 16 se muestra un diagrama resumen de la evolución de los métodos de reconstrucción basados en el etiquetado de vértices.

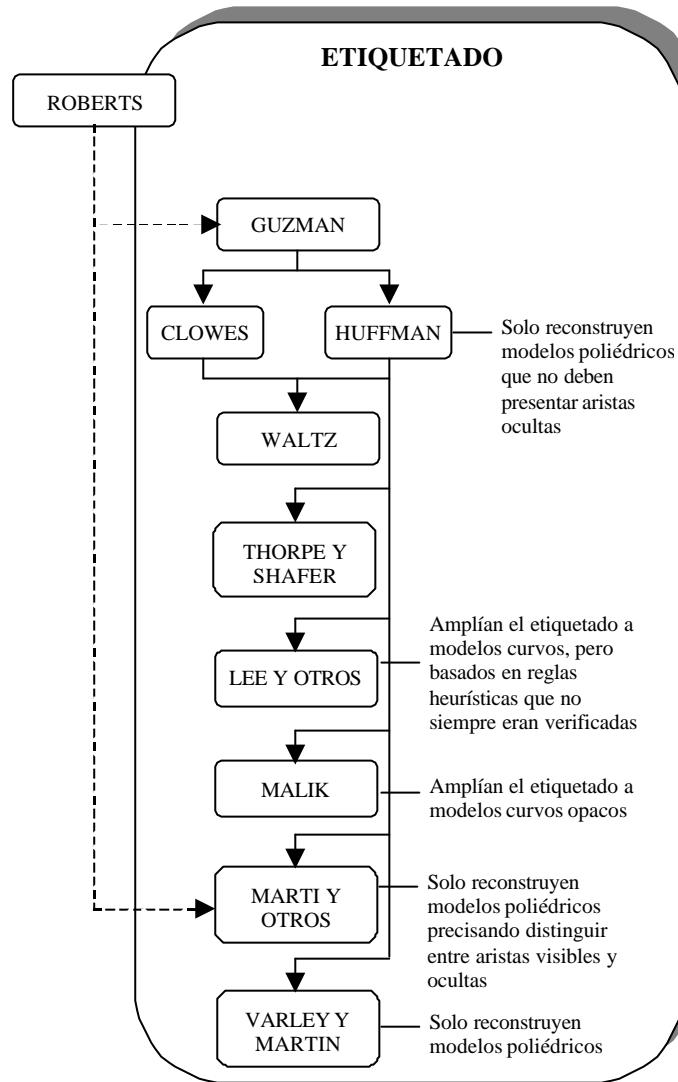


Figura 16. Evolución de los métodos de etiquetado.

2.2.2 Métodos basados en el espacio gradiente.

Estos métodos son también generalmente considerados como procesos de interpretación. **Mackworth** [Mack73] estableció el primer método de gradientes para la resolución de procesos de reconstrucción. Su método se basa en una correspondencia establecida entre los gradientes de las superficies poliédricas con las líneas que componen la imagen. Mackworth interpreta los dibujos lineales construyendo la imagen de cada plano en el espacio de gradientes.

Su método se ha ampliado para determinar si un dibujo lineal es realizable. Es decir, el método del espacio de gradientes se puede usar para detectar otra clase de dibujos lineales no realizables.

La existencia de una imagen de gradientes es aun únicamente una condición necesaria para que los dibujos lineales sean realizables. Algunos dibujos lineales no son realizables incluso aunque se pueda construir sus imágenes gradientes. Algunos ejemplos pueden encontrarse en Sugihara [Sugi86].

De acuerdo con otras publicaciones que hacen referencia a los trabajos de **Wei**, este autor, basándose en la idea de Mackworth, amplió el espacio dual gradiente estableciendo cuatro condiciones que debían cumplir los vértices en proyección para representar un posible diedro. Sin embargo, el método de Wei estaba limitado a ciertos poliedros.

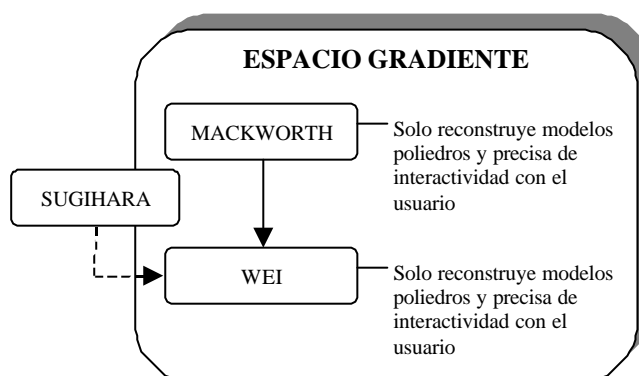


Figura 17. Evolución de los métodos basados en el espacio gradiente.

2.2.3 Métodos basados en la programación lineal.

Sugihara [Sugi82] presentó un mecanismo computacional basado en la programación lineal para extraer estructuras tridimensionales poliédricas a partir de dibujos lineales bidimensionales. Sugihara [Sugi] consiguió dar una condición necesaria y suficiente que permitía que un dibujo lineal representase un objeto poliédrico en términos del problema de programación lineal. Su formulación permite resolver el problema de discriminar entre dibujos lineales correctos e incorrectos. Sin embargo, la

condición es tan precisa matemáticamente que algunos dibujos son rechazados simplemente porque los vértices se desvían ligeramente de las posiciones correctas.

También **Grimstead y Martin** [Grim95] [Grim96] plantearon la Reconstrucción Geométrica en términos de programación lineal, si bien su método partía del esquema de etiquetado presentado por Martí y otros.

Las aportaciones más recientes a la Reconstrucción Geométrica están basadas en las ideas de **Varley y Martin** [Var100(3)]. Estos autores tratan el problema de la Reconstrucción Geométrica basándose en reglas de la geometría. Su método establece la formulación matemática de un sistema de ecuaciones que representa un conjunto de condiciones geométricas que deberá verificar el modelo 3D, a partir del análisis de una imagen o de una porción de la misma. Su método asume que topológicamente todos los vértices del objeto son triedros y que la imagen representa siempre un poliedro euleriano.

La figura 18 muestra un diagrama resumen de la evolución de los métodos basados en programación lineal.

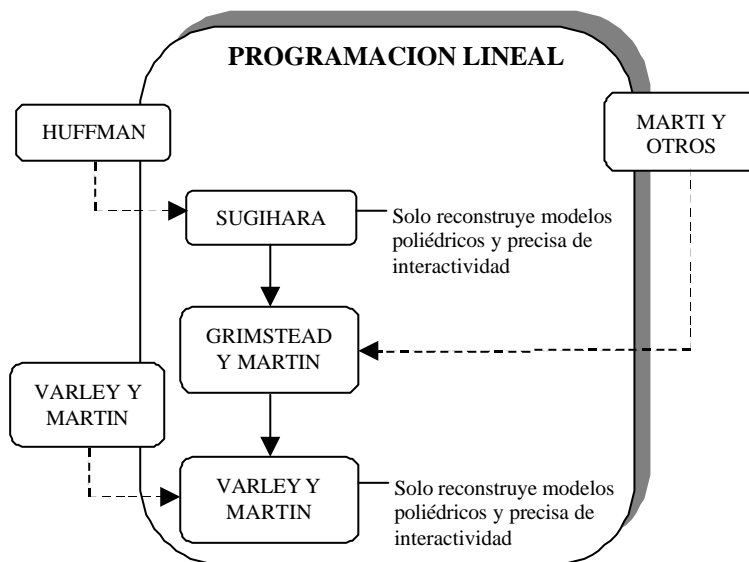


Figura 18. Evolución de los métodos basados en programación lineal.

2.2.4 Métodos perceptuales.

Los métodos perceptuales se distinguen del resto de los métodos por intentar implementar, a través del lenguaje secuencial de los ordenadores, la forma en que percibe el ser humano.

Lamb y Bandopadhyay [Lamb90] diseñaron un primer método perceptual cuyas características principales se basaban en el alto grado de interacción del usuario que debía designar los ejes principales (intersección de los planos de proyección), si bien, siendo un algoritmo matemático basado en la proporcionalidad de la imagen con el modelo 3D, permitía cierta flexibilidad a imprecisiones del dibujo. En su algoritmo las líneas paralelas de la imagen aparecen paralelas en el modelo y las aristas paralelas a los ejes principales se dibujan con longitudes proporcionales a las dimensiones reales permitiendo definir paralelogramos con caras paralelas a los planos de proyección. Aplicando el etiquetado de Waltz se obtiene un gráfico de adyacencia que permite definir la orientación de cada arista con respecto a los ejes principales. La característica que diferencia a éste método de otros métodos de etiquetado es la aplicación de una regla de la percepción visual humana: la ley de la simetría permite reconstruir modelos en los que no todas sus aristas resultan paralelas a tres direcciones principales. Su principal inconveniente es que los modelos a reconstruir no deben contener aristas ocultas dada la ambigüedad que la representación de dichas aristas ocasionan.

Marill [Mari91] presentó un primer algoritmo capaz de reconstruir objetos poliédricos basado en la estrategia de inflado, consistente en asignar coordenadas “z” a todos los vértices del modelo mientras se mantienen las coordenadas (x,y) de la imagen.

Las coordenadas “z” eran modificadas mediante la optimización de una función objetivo que estaba formada por un único componente, el MSDA (mínima desviación estándar de ángulos). Es decir, minimiza las diferencias de los ángulos entre cada pareja de aristas que concurren en cada uno de los vértices, obteniendo como resultado estructuras alámbricas tridimensionales.

Su mayor inconveniente proviene de la definición de la función objetivo, basada en una regla heurística que únicamente resulta válida para modelos regulares, por lo que sus resultados presentaban un alto índice de fallos.

El método establecido por Marill basado en el inflado y la optimización ha sido desarrollado posteriormente por diferentes autores como **Baird y Wang** [Bair91], si

bien los métodos perceptuales más prometedores se mostraron a partir de **Leclerc y Fischler** [Lecl92].

Su método establece una función objetivo que es suma de una serie de términos que representan distintas reglas de la percepción a partir de una serie de atributos propios del esquema de la imagen entrante. La función objetivo es definida en la forma:

$$E(\lambda) = \lambda SDA^2 + (1-\lambda) DP \quad (1)$$

donde el término “DP”, que denominan desviación de planitud, mide la desviación de los vértices del modelo obtenido respecto a las caras planas a las que el algoritmo supone que deberían pertenecer y SDA hacia referencia a la desviación de ángulos propuesta por Marill. Dicha suposición se basa en un algoritmo de detección de caras planas que se aplica a la figura de partida, y que también es descrito por los autores.

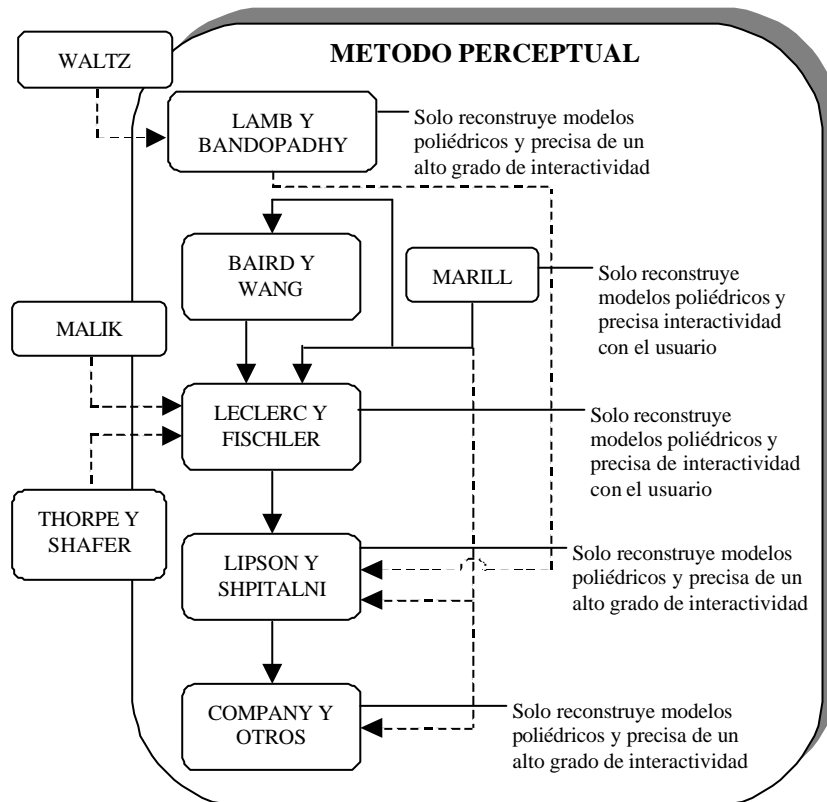


Figura 19. Evolución de los métodos basados en reglas de la percepción.

Posteriormente otro método de percepción basado optimización fue desarrollado por **Lipson y Shpitalni** [Lips95] [Lips96] [Sphi96] [Sphi97]. En sus trabajos proponen la minimización de una función suma de sub-funciones que representan a las regularidades que pueden apreciarse en una imagen bidimensional que representa un objeto 3D. Dichas regularidades imitan la sensibilidad perceptual del ser humano al observar un boceto y que agrupaban en tres tipos:

- 1) Las que reflejan alguna relación espacial entre entidades individuales (por ejemplo el paralelismo entre aristas).
- 2) Las que reflejan alguna relación espacial entre un grupo de entidades (por ejemplo una simetría oblicua en las aristas que definen el contorno de una cara).
- 3) Las regularidades que afectan a todo el dibujo (como la proporcionalidad entre las longitudes del dibujo y las longitudes reales).

Su método tolera imperfecciones y permite la reconstrucción de una gran variedad de objetos, incluyendo caras planas y cilíndricas, pero el porcentaje de fallos aumenta al considerar objetos complejos o con superficies curvas.

Posteriormente **Company y otros** [Comp99(1)] [Comp99(2)] han trabajado en la reconstrucción aplicando procesos de optimización, intentando disminuir el alto índice de errores producidos.

En la figura 20 se muestra un cuadro resumen de los distintos métodos de reconstrucción a partir de una única vista.

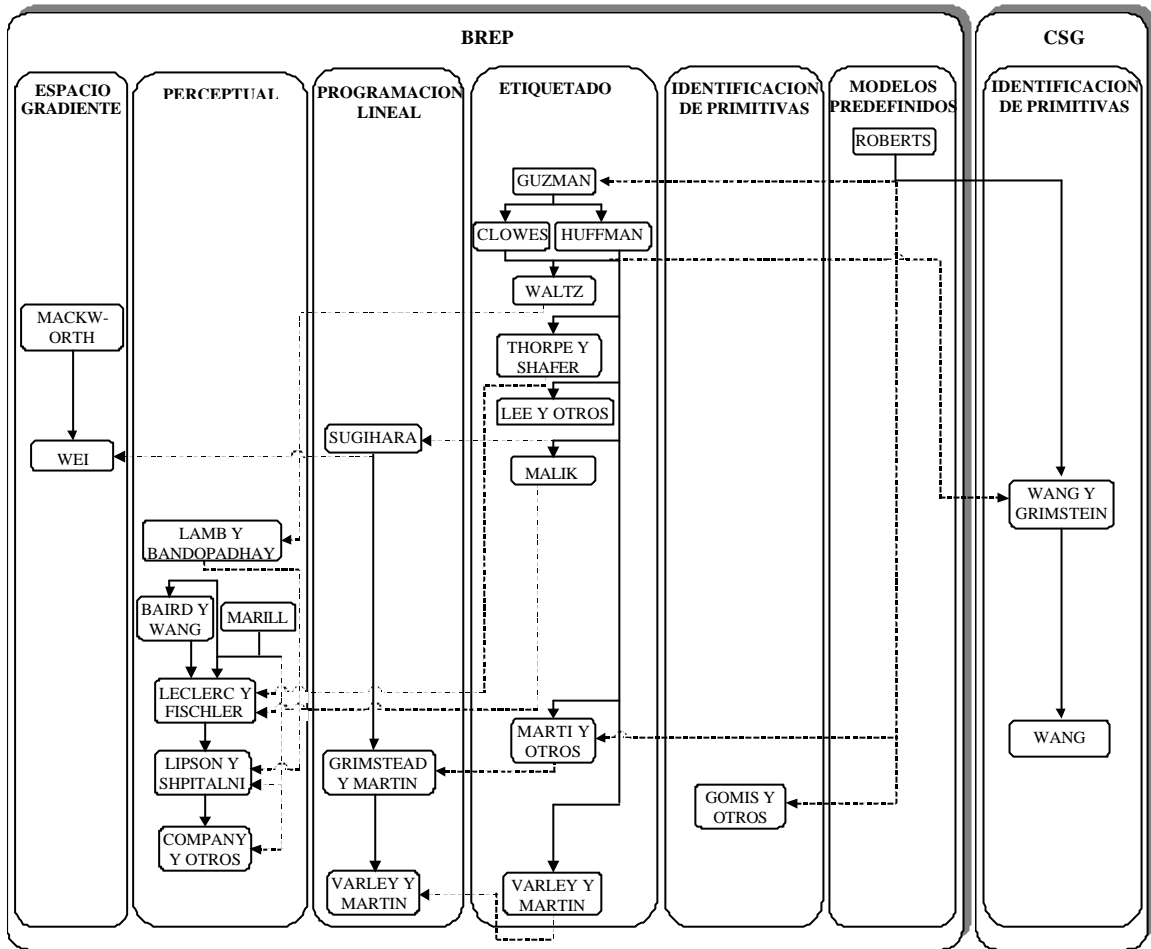


Figura 20. Evolución de métodos de reconstrucción de vista única

En el diagrama de la figura 21 se resume el estado del arte descrito en este capítulo. Pretende ser una visión de conjunto de las que, a nuestro entender, son las aportaciones más significativas en la reconstrucción geométrica. El diagrama sigue el mismo criterio que todos los diagramas parciales presentados en éste mismo capítulo: ordena las aportaciones de los autores por columnas según las naturaleza de los métodos empleados, y establece una secuencia temporal por filas; tan exacta como la búsqueda de un diagrama compacto ha permitido.

METODOS DE ÚNICA VISTA

METODOS DE VISTAS MÚLTIPLES

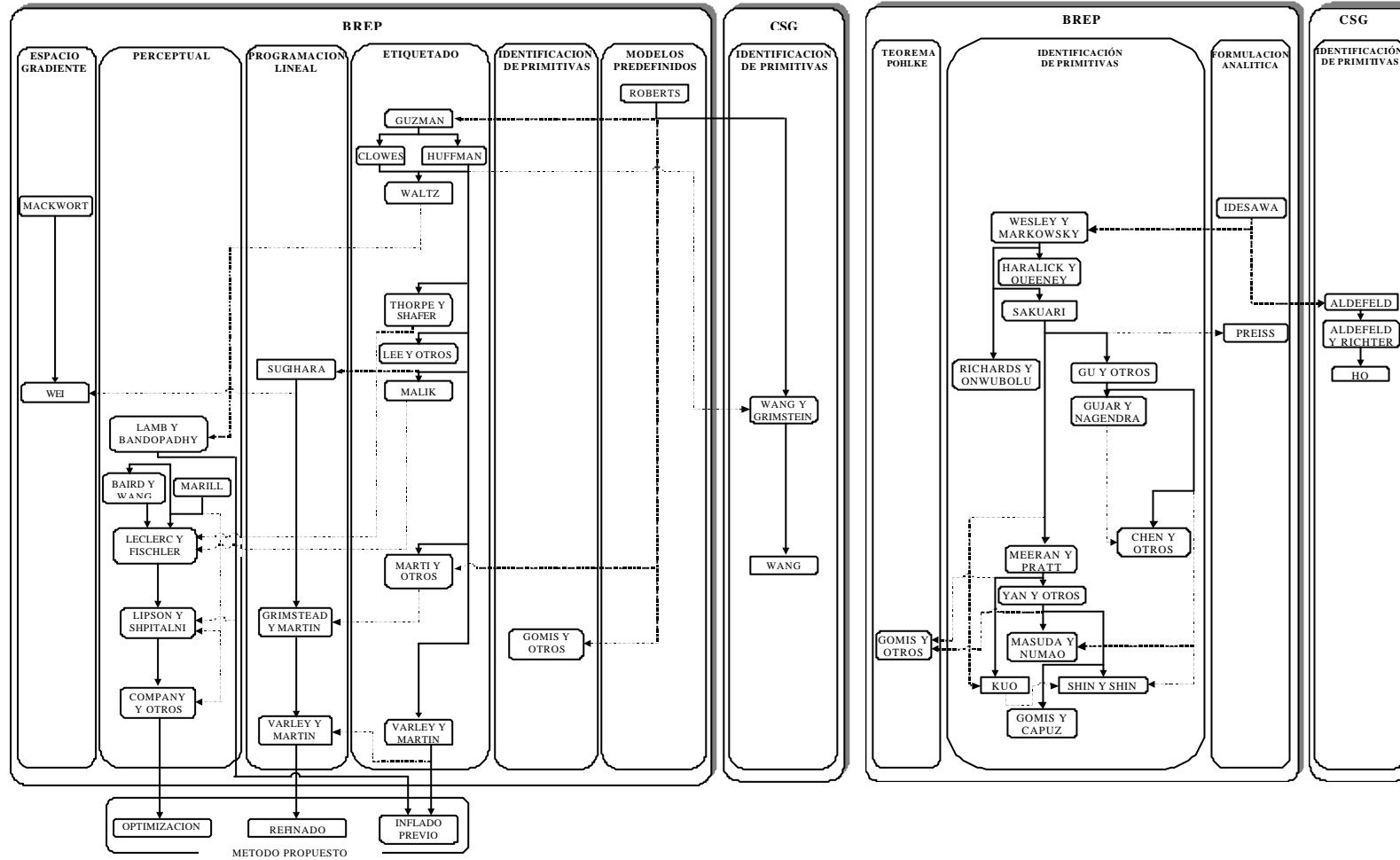


Figura 21. Resumen cronológico de los antecedentes de la Reconstrucción Geométrica

PERCEPCIÓN VISUAL Y RECONSTRUCCIÓN GEOMÉTRICA

3

El ser humano pasa la vida construyendo profundidad a partir de imágenes 2D. La construcción de la profundidad implica considerar diversas propiedades visuales y conseguir que todas ellas sean mutuamente coherentes. Organizamos el mundo visual en objetos, a los que asignamos formas tridimensionales. Colocamos fuentes de luz que iluminan esos objetos, y asignamos color tanto a las fuentes luminosas como a los objetos.

El proceso es muy complejo, porque las imágenes que visualizamos son infinitamente ambiguas. Existen incontables maneras en las que se puede interpretar una imagen en términos de objeto, de forma, de colores y de iluminadores. No obstante, todos los videntes parecen estar de acuerdo sobre cuáles son las propiedades ha definir ante una imagen.

Esto es debido a que el ser humano percibe según una serie de reglas muy sofisticadas que los investigadores apenas han empezado a desvelar, y por las cuales elegimos una interpretación de entre todas las incontables posibilidades [Hoff00].

Las reglas de la percepción son muy variadas, porque se aplican a la forma, la textura, el color, el movimiento, etc., pero en esta tesis solamente se tratarán aquellas reglas que afecten más directamente a la geometría del modelo.

3.1 La Reconstrucción Geométrica como proceso de optimización.

Desde el punto de vista de la Geometría, ha sido siempre bien conocido que no es posible recuperar totalmente un modelo geométrico en 3D a partir de una única proyección del mismo. También desde el punto de vista de la psicología se hace referencia al problema fundamental de captar la profundidad: *la imagen que recibe el ojo tiene dos dimensiones; por consiguiente, tiene incontables interpretaciones tridimensionales* [Hoff00].

Sin embargo, desde el punto de vista psicológico es también bien sabido el hecho de que los humanos no parecen tener problemas para identificar modelos en 3D

en imágenes en 2D. Lo que es más, parece existir un gran consenso sobre cual es el modelo “correcto” y “simple” que todos los humanos ven en cada dibujo.

Esto es debido a que el ser humano no construye la profundidad de cualquier manera, sino que lo hace obedeciendo unas reglas. El papel de las reglas visuales es fundamental, porque, según Hoffman [Hoff00]: *usted construye los mundos visuales a partir de imágenes ambiguas, según unas reglas visuales.*

En otras palabras, nuestro sistema visual tiene prejuicios. Sólo construye aquellos mundos tridimensionales que se adaptan a sus reglas; a la mayoría de los demás simplemente no les hace caso.

La percepción visual, por tanto, actúa de una manera tentativa e iterativa. Es decir, algunas fuerzas guían al cerebro para interpretar la figura de acuerdo con determinadas reglas, llevando a cabo sucesivos intentos de organización hasta que se minimiza el incumplimiento de dichas reglas. El cerebro busca la “mejor forma”, o forma prevalente, mediante un proceso de descarte de aquellas que son incompatibles con ella.

Por consiguiente, un proceso iterativo donde alguna solución inicial es refinada de acuerdo con algunas reglas visuales, o características perceptivas, parece ser una buena estrategia para conseguir lo que la geometría por sí sola no pueden obtener: *un modelo en 3D psicológicamente aceptable.*

Un sistema automático de reconstrucción se debe implementar de modo que sea capaz de decidir las modificaciones que deben hacerse para conseguir el modelo deseado. Decidir las modificaciones significa elaborar una estrategia para seleccionar las modificaciones a realizar, y otra estrategia para *evaluar* la bondad de los modelos resultantes. Por consiguiente, se debe definir tanto un criterio de búsqueda, válido para un proceso iterativo de búsqueda de nuevas soluciones, como un criterio que permita evaluar objetivamente las diferentes soluciones.

La evaluación puede quedar resuelta si se define una cifra de mérito. Es decir, una calificación (lo más objetiva posible) que permita clasificar todas las alternativas estudiadas con arreglo a un mismo criterio. En nuestro caso, dicha cifra de mérito puede establecerse como un factor que mida el cumplimiento de las reglas de percepción y que constituya el criterio para decidir la bondad del modelo generado.

Si el criterio de búsqueda, el de evaluación, o ambos, pueden ser automatizados, el grado de asistencia que puede prestar un ordenador aumentará notablemente, tendiendo hacia la total automatización del proceso de diseño.

La estrategia así descrita copia el proceso de percepción. Es por ello que la Reconstrucción Geométrica, entendida como un problema de percepción, puede ser descrita en términos de un problema de optimización matemática (ver figura 22).

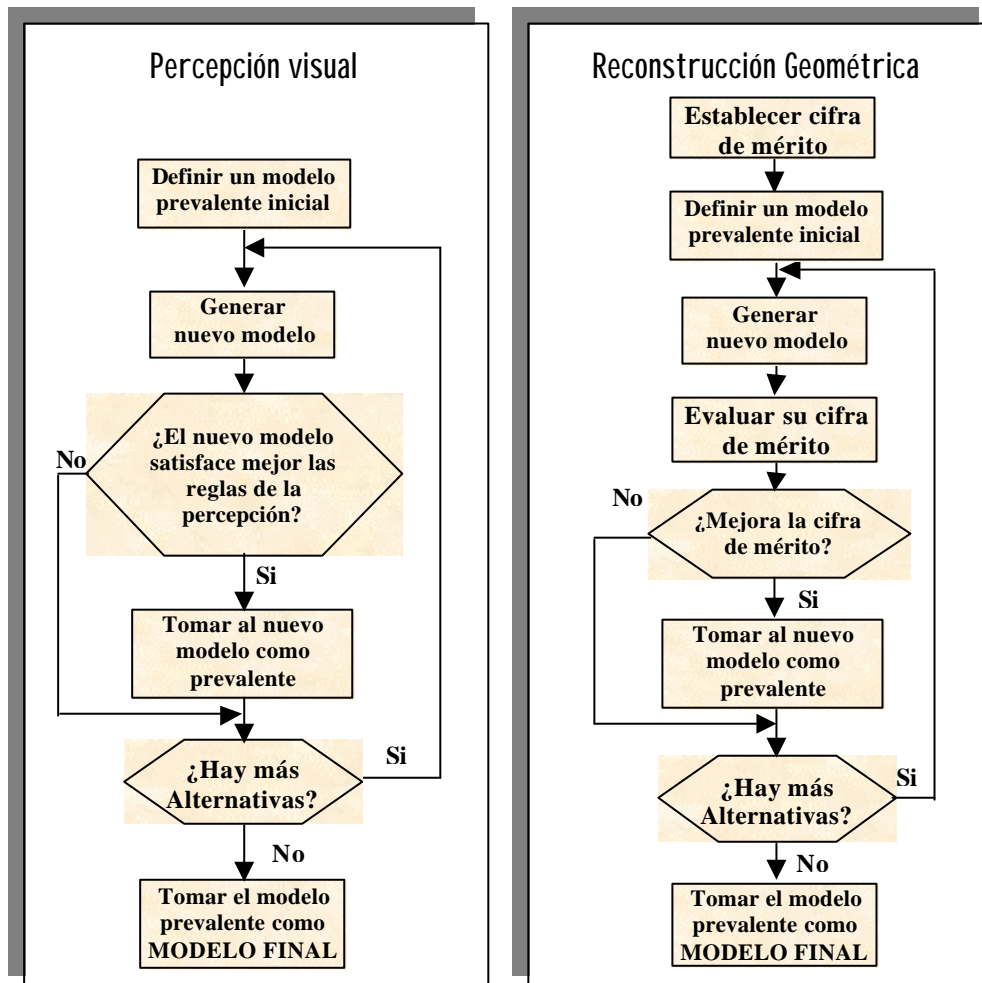


Figura 22. Analogía entre la percepción visual humana y el proceso de reconstrucción mediante métodos de optimización de regularidades.

3.2 Definición de las variables de reconstrucción.

Optimizar un problema significa determinar una *cifra de mérito*, que cuantifique la bondad de cada una de las posibles soluciones, y encontrar aquella solución que la minimice. Para el tratamiento informático, se requiere que la cifra de mérito sea una

función de un conjunto de variables independientes. Se denomina *solución* a un conjunto de valores válidos de dichas variables. Los problemas de optimización suelen tener un número “n” de variables bastante elevado. A pesar de ello, se suele establecer una analogía entre el conjunto de soluciones y los puntos de un espacio n-dimensional en el que cada dirección corresponde con una de las variables. De ahí que se utilice habitualmente la terminología de “espacio n-dimensional de soluciones” y “punto solución”.

La configuración del espacio de soluciones depende de la naturaleza del problema considerado. En el caso de la reconstrucción, la optimización se ha aplicado únicamente a *objetos poliédricos*, y la definición del espacio de soluciones se ha basado en lo que se denomina el *método de inflado* de un grafo para convertirlo en un modelo alámbrico del poliedro. El inflado es, a grosso modo, la operación inversa de la proyección y sección.

Recordemos que se llama proyección de una figura compuesta por los puntos A, B, C, D,... y las rectas m, n, s,... desde un punto fijo del espacio V, a la figura compuesta por las rectas proyectantes $r_A, r_B, r_C, r_D, \dots$ (determinadas por el punto V y cada uno de los puntos A, B, C, D,...) y los planos proyectantes $\sigma_m, \sigma_n, \sigma_s, \dots$ (que determina el punto V con cada una de las rectas m, n, s,...). Y recordemos que la sección que el plano de proyección π le produce a dicha *figura proyección* (tridimensional) da lugar a la *figura imagen* (bidimensional), compuesta por los vértices A', B', C',... y las rectas m', n', s',... (figura 23).

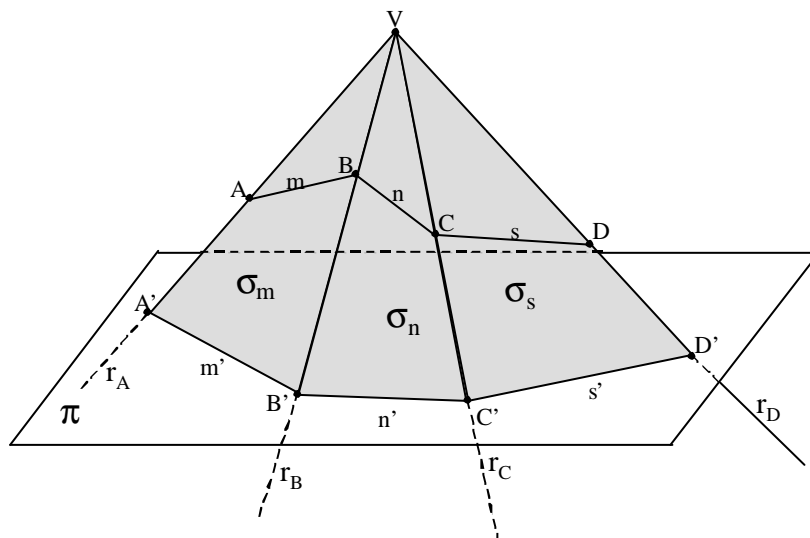


Figura 23. Concepto de proyección

El centro de proyección puede ser propio (figura 24a) o impropio (figura 24b). En el primer caso se dice que la *proyección es central*, y en el segundo, que la *proyección es paralela o cilíndrica*. En este último caso, los rayos proyectantes son todos ellos paralelos a una dirección (la del punto impropio) y cabe distinguir, a su vez, dos clases de proyección, en función de la relación existente entre la dirección de proyección y la orientación del plano de proyección.

Se llama *proyección ortogonal* de una figura sobre un plano a la proyección paralela de la figura sobre este plano, cuando la dirección de los rayos proyectantes es perpendicular al plano. Si la dirección no es ortogonal al plano diremos que es una *proyección oblicua*.

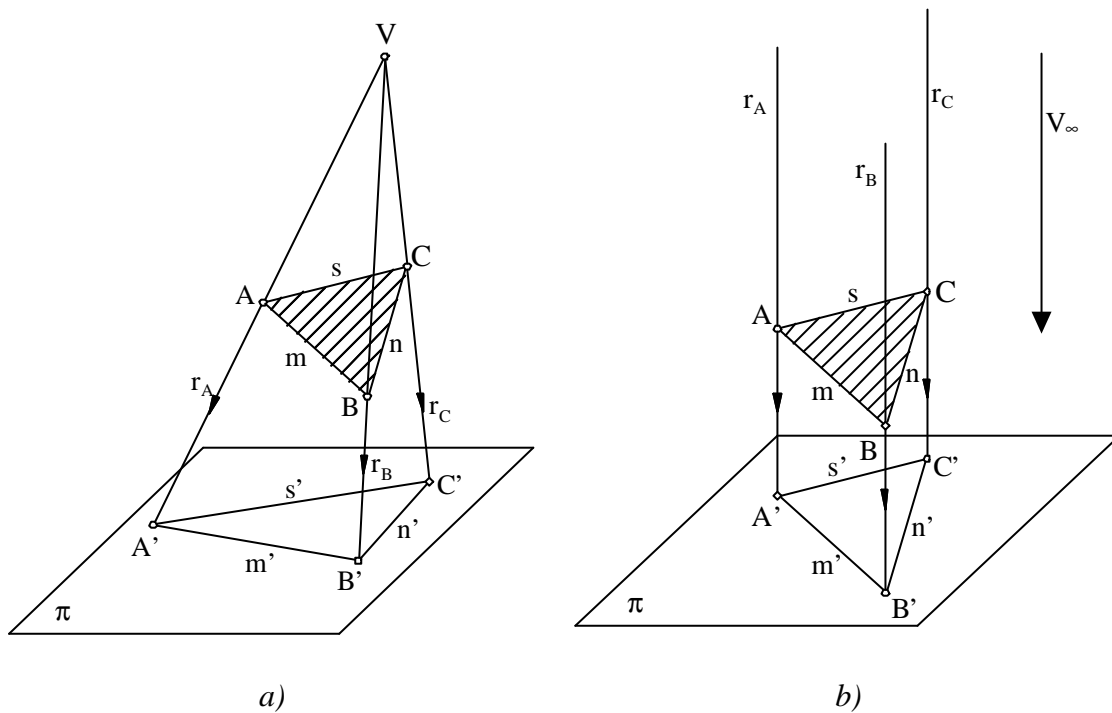


Figura 24. Clases de proyección
 a) Proyección cónica b) Proyección cilíndrica

Esta tesis se ha centrado en la reconstrucción de modelos a partir de una única proyección cilíndrica ortogonal de los mismos. En concreto, estas proyecciones son representaciones axonométricas ortogonales de dichos modelos, que son las representaciones de vista única de uso más habitual en los planos de Ingeniería.

En cuanto a la estrategia adoptada para configurar el espacio de soluciones se ha mantenido la empleada en estudios anteriores, que consiste en elegir un sistema de referencia de manera que los ejes X e Y del modelo quedan contenidos en el plano de proyección. Es decir, se asigna a cada vértice del modelo las mismas coordenadas (x,y) de su correspondiente nodo de la imagen (en la figura 25 se observa que $x_A = x_{A'}$, $x_B = x_{B'}$, $x_C = x_{C'}$ e $y_A = y_{A'}$, $y_B = y_{B'}$, $y_C = y_{C'}$). Las coordenadas z de la imagen son cero, y las coordenadas “z” de cada vértice del modelo son las variables de reconstrucción. En consecuencia, cada vértice del modelo se encontrará según una recta proyectante, perpendicular al plano de la imagen por el nodo proyección del vértice.

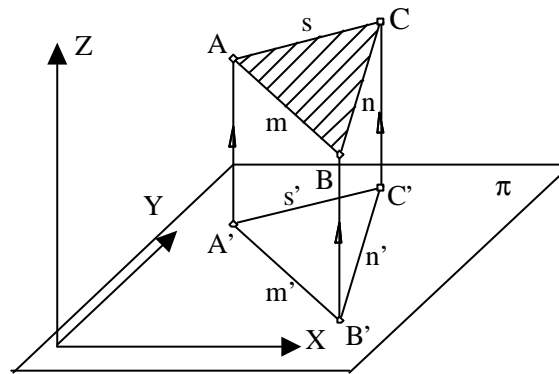


Figura 25. Sistema de referencia para inflado del modelo

De esta manera puede establecerse una doble correspondencia entre los modelos y las imágenes:

1. Los vértices del modelo tridimensional se corresponden biunivocamente con los nodos de la imagen.
2. Las aristas del modelo se corresponden biunivocamente con los enlaces de la imagen.

El método anteriormente descrito tiene su justificación a partir de una de las reglas de la percepción visual humana. Consideremos que deseamos conceder interpretación tridimensional al dibujo que se muestra en la figura 26a, es decir, que deseamos construir un objeto tridimensional que mirado desde cierta dirección nos proporcionará esta imagen.

El lector de este documento muy posiblemente se haya limitado a construir una “V”, lo cual funciona en el sentido de que hay muchas formas de mirar una “V” que dan la imagen representada. Pero también podría construir una interpretación tridimensional distinta: dos líneas rectas que se cruzan en el espacio. Esta interpretación también sería válida pero no resulta tan plausible como la “V”.

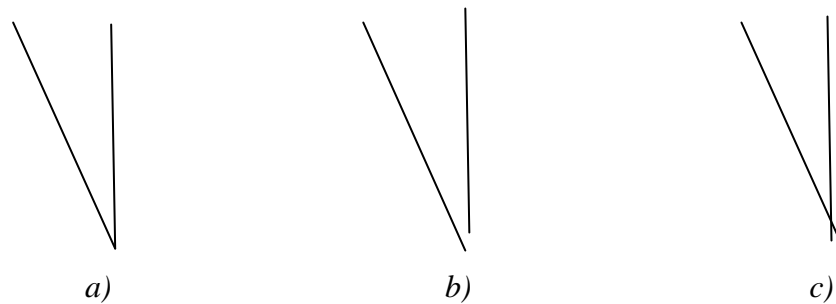


Figura 26. Reglas de visiones genéricas

Esto es debido a que si, en la segunda interpretación, se moviera ligeramente la cabeza en cualquier dirección, aparecería un espacio más amplio entre los extremos de las líneas (figura 26b), o bien un cruce (figura 26c). Es decir, la segunda interpretación se trata de un accidente relacionado con el punto de vista, ya que casi todas las visiones ofrecerían una imagen en la que esos extremos no coinciden.

Sin embargo, en la interpretación en “V”, los extremos no coinciden accidentalmente, sino que lo hacen independientemente del punto de vista que adoptemos.

Por consiguiente y dado que para considerar la segunda interpretación es preciso colocar nuestro punto de vista exactamente en una posición concreta, el vidente lo rechaza; igual que haría con cualquier interpretación que pueda ser resultado de un accidente.

En definitiva, el vidente percibe de acuerdo con la “**Regla de las visiones genéricas**” [Hoff00]:

Construir sólo aquellos mundos visuales para los cuales la imagen es una visión estable (es decir genérica).

A partir de la regla de visiones genéricas se establecen dos hipótesis acordes con el método de inflado y que constituyen la base de partida de la reconstrucción tridimensional de poliedros:

Hipótesis 1: Cualquier enlace en una imagen siempre será interpretado como una arista tridimensional (se descarta que pueda ser considerado como otra figura, tal como un arco, mirada desde cierto punto de vista).

Hipótesis 2: Cualquier nodo de una imagen siempre se corresponderá como un vértice tridimensional (se descarta que pueda corresponderse con un cruce ocasional, o una línea recta proyectada en un punto).

Estas hipótesis son unas normas sencillas pero muy eficientes, que eliminan incontables interpretaciones en 3D. Si observamos una de las figuras de Kopfermann (figura 27a), esta tiene tres líneas que cruzan su centro uniando los seis vértices del contorno hexagonal. Según la hipótesis 1, debemos interpretar cada línea como una recta. Esto explica la dificultad con la que nos encontramos para interpretar la figura como un cubo; dado que para ello cada una de las tres diagonales del hexágono debería descomponerse en una pareja de aristas. Basta una pequeña rotación del cubo para que su imagen cumpla las dos hipótesis anteriores, y, en consecuencia, sea fácil de reconstruir (figura 27b).

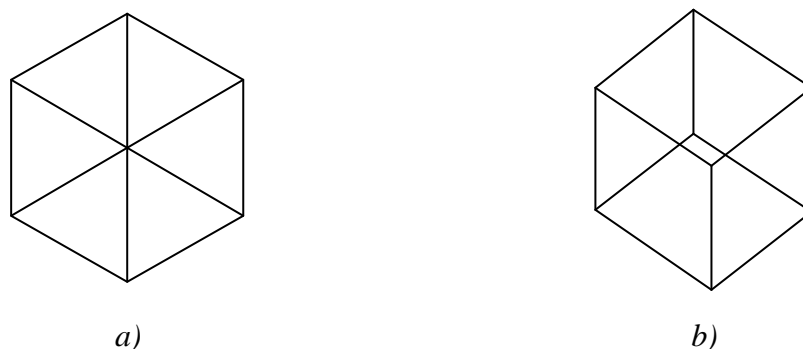


Figura 27. Hipótesis 1 en la figura de Kopfermann

En definitiva, nuestro espacio de soluciones queda definido por los valores que puedan ser asignados a las coordenadas “z” de cada uno de los vértices de la imagen. Es

decir, cada vértice del modelo mantendrá las mismas coordenadas (x,y) del correspondiente nodo de la imagen, mientras su coordenada “z” variará a lo largo de una recta perpendicular al plano que contiene la imagen. Por tanto, las variables de reconstrucción definen un vector n-dimensional, en la forma $Z = (z_1, z_2, z_3, \dots, z_n)$, siendo “n” el número de nodos de la imagen (figura 28).

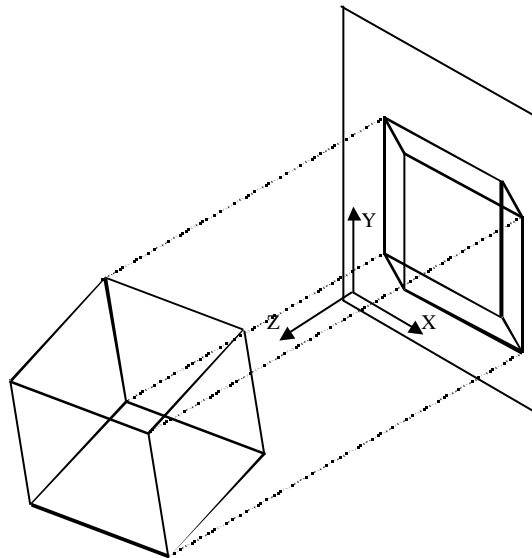


Figura 28. Método de inflado

Marill fue el primer autor que hemos podido encontrar que formuló el método de inflado en términos semejantes a los descritos arriba.

También cabe señalar que la regla de las visiones genéricas ha estado presente en la práctica totalidad de los métodos de reconstrucción a partir de vista única desarrollados hasta la fecha. Aunque ha sido justificada como un requisito derivado de las limitaciones del método, o una regla heurística, y no tenemos constancia de que haya sido anteriormente justificada en base a hipótesis perceptivas.

3.3 Formulación de la función objetivo.

Definida la configuración del espacio de soluciones, hay que volver a insistir en que el conjunto de variables de reconstrucción (coordenadas “z” de los vértices del modelo), podrían ofrecer un infinito número de *modelos geoméricamente válidos*. La validez geométrica se justifica porque cualquier modelo obtenido asignando un vector

válido de variables daría como proyección la figura de partida. Marill denominó *extensión ortográfica* al conjunto de los infinitos modelos tridimensionales que pueden obtenerse asignando valores válidos a las coordenadas z de todos los nodos (figura 29).

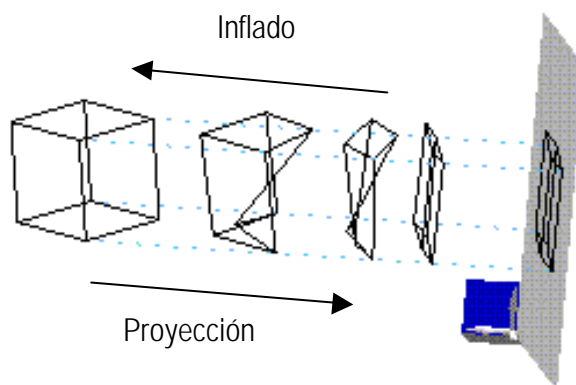


Figura 29. El conjunto de todos los modelos obtenidos por inflado, determinan la extensión ortográfica de una imagen

Obviamente, no todos los modelos de la extensión ortográfica cumplen las leyes de la percepción visual. Por ello se precisa desarrollar algún criterio para elegir un modelo particular entre los contenidos en dicha extensión ortográfica. Para ello se aprovecha el enfoque psicológico. Dado que sólo hay *una* solución *buena* en términos de verosimilitud psicológica.

Reglas como las “Visiones Genéricas”, que ya hemos presentado, son las que se deben implementar para seleccionar el modelo prevalente y llegar así a un método de reconstrucción geométrica próximo al proceso de percepción humano. La *sencillez* y la *simetría*, constituyen las reglas fundamentales que rigen la percepción visual humana [Hoff00]. Pero tales reglas resultan difíciles de formular analíticamente para objetos complejos. Tampoco es fácil descubrir tales características complejas a partir de la imagen. Es por ello que estas reglas han sido desglosadas en reglas más sencillas, a las que se suele denominar *señales perceptivas* o *características perceptivas*.

Obviamente nuestro problema se centra en la detección y formulación analítica de aquellas reglas que más perturben la forma prevalente. Maximizar o minimizar una función objetivo que cuantifique el cumplimiento de esas reglas es el criterio para

definir el problema de optimización que encuentre la solución correcta entre todas las de la extensión ortográfica.

Las *regularidades*, que posteriormente serán comentadas con mayor amplitud, son el camino para expresar analíticamente las reglas perceptuales que utilizamos en la interpretación de las imágenes.

Las regularidades deben ser expresadas en términos de las variables independientes “z”, y deben ser formuladas para ser iguales a cero cuando se verifica un completo cumplimiento de las reglas que representan, y muy diferente de cero para un claro incumplimiento. Se plantea así un problema de minimización de una Función Objetivo.

La Función Objetivo debe quedar definida por tanto, en términos o regularidades de la siguiente manera:

$$F = \sum \alpha_j R_j(z) \quad (2)$$

donde:

- α_j es el j-ésimo coeficiente de peso, y
- $R_j(z)$ es la j-ésima regularidad.

3.4 Regularidades de la imagen.

Como se ha comentado anteriormente, las regularidades son el camino para formular las reglas de la percepción visual humana. Las primeras apariciones de la palabra regularidad en el ámbito científico se deben al análisis geométrico de formas. Así se define una forma como *regular* cuando todos los elementos de idéntica categoría son iguales (equilateral, equiangular, ...). Muy posteriormente psicólogos de la Gestalt asociaron la palabra regularidad con la percepción de representaciones gráficas: llamaban regularidades a aquellas relaciones que no pueden ser accidentales. También en el área de la visualización científica, en el ámbito de la percepción semántica aparece el término de regularidad como sinónimo de forma que encaja en una *plantilla*.

Finalmente, en la Reconstrucción Geométrica las regularidades son interpretadas como aquellas propiedades del modelo que pueden ser detectadas en la imagen. Es

decir, que se supone que inspeccionando la imagen es posible deducir el número necesario de propiedades que describen un modelo.

A continuación se exponen las diversas regularidades que han sido implementadas durante el desarrollo de esta tesis. Todas ellas están tomadas de los trabajos de Lipson y Shpitalni [Lips96], aunque la justificación de las mismas a partir de reglas de la percepción es propia.

Pero, antes de pasar a la descripción y formulación detallada de las diferentes regularidades es necesario puntualizar algunos aspectos generales relacionados con el empleo de las regularidades para el inflado mediante optimización.

3.4.1 Tratamiento de las regularidades en la optimización.

El tratamiento de las regularidades es una de las características que hace diferir a la reconstrucción por optimización del resto de los métodos de reconstrucción. Es preciso tener presente la definición que anteriormente se apuntaba del concepto de regularidad en el sentido de la percepción de las representaciones gráficas: se llaman regularidades a aquellas relaciones que no pueden ser accidentales. Sin embargo, y como se mostrará en la formulación de las regularidades, tales accidentes son posibles. Es decir, que las regularidades permiten extraer propiedades que *lo más probable* es que se verifiquen en el modelo. Pero dicha probabilidad no tiene valor 1. En otras palabras, las propiedades que las regularidades extraen de la imagen, no siempre representan propiedades reales del modelo.

Es por ello que otros métodos de reconstrucción que resuelven analíticamente sistemas de ecuaciones creados a base de imponer el cumplimiento de dichas regularidades, en ocasiones pueden generar falsos modelos o pueden ser incapaces de resolver el problema.

Los procesos de optimización pueden provocar estos mismos errores en el caso de que las regularidades fuesen expresadas como restricciones, es decir como condiciones que el modelo debe verificar obligatoriamente para ser considerado válido. Y es por ello que las regularidades deben ser introducidas en el proceso de optimización como restricciones implícitas. De esta forma se consigue penalizar los modelos que no cumplan todas las regularidades, pero sin descartar aquellos modelos que incumplan alguna de ellas.

Pero incluso suponiendo que las regularidades expresasen propiedades del modelo con probabilidad 1, es decir que todos los modelos verificasen todas las regularidades, existe otra problemática que se deriva del rigor del análisis analítico. Los efectos pueden verse claramente en las imágenes que se ilustran en la figura 30 donde la interpretación usual de cualquier vidente (una pirámide truncada) no podría corresponder con una interpretación analítica dado que todas las caras planas que se interceptan entre sí, deberían hacerlo sobre aristas que tuviesen un punto en común (figura 30b).

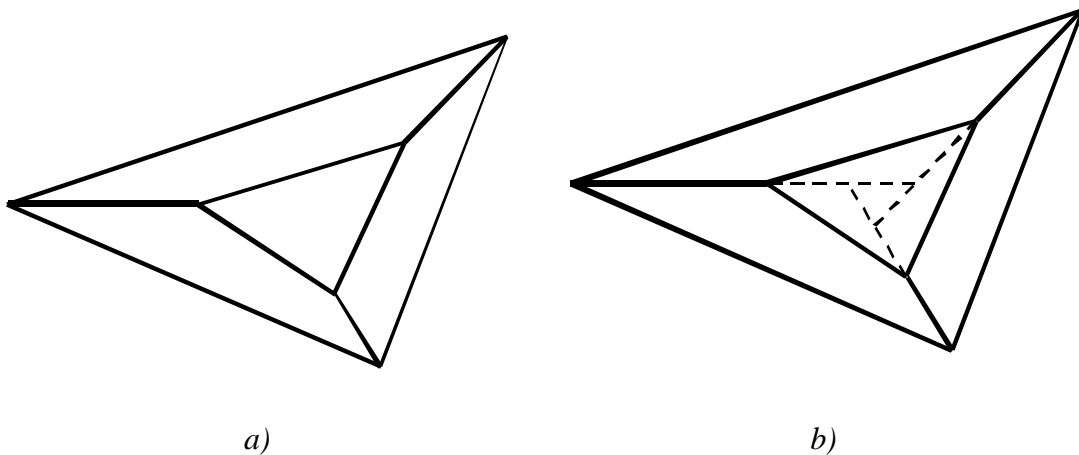


Figura 30. Efectos del rigor matemático en la formulación de regularidades

No obstante, algunos de los problemas del rigor matemático han sido estudiados para paliar este defecto. Tal es el caso de la desviación angular entre aristas: cuando dos aristas no son exactamente paralelas, pueden ser interpretadas por el observador humano como si lo fuesen, mientras que matemáticamente dichas aristas serían consideradas como concurrentes.

Lipson y Sphitalni [Lips96] incluyeron una importante aportación en su estudio que consistía en incluir un *factor de confianza*. Dicho factor de confianza está definido de manera que alcance valor unidad cuando la condición que pondera toma un *valor de referencia*, y decrece hasta cero de acuerdo a una curva de distribución normal, que toma valor nulo a partir de una *desviación máxima permisible* (figura 31).

La formulación general de la función de confianza viene dada por la expresión:

$$\mu_{a,b}(x) = e^{-((x-a)/b)^2} \quad (3)$$

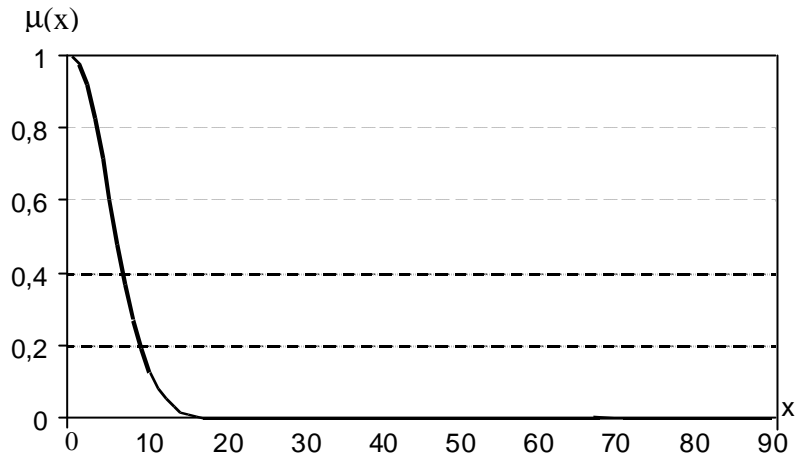


Figura 31. Representación gráfica de la función de confianza

donde:

- “x” representa el valor obtenido de la imagen (por ejemplo, el ángulo que forman dos rectas que deberían ser paralelas).
- “a” es un valor nominal de referencia (por ejemplo 0° para el paralelismo).
- “b” es la desviación límite permisible (por ejemplo 7° para el paralelismo).

Con propósitos prácticos, la ecuación dada en (3) puede ser modificada para eliminar los valores muy próximos a cero de acuerdo con la expresión:

$$\mu_{a,b}(x) = \max [0, 1.1 e^{-((x-a)/b)^2} - 0.1] \quad (4)$$

3.4.2 Regularidades de la imagen.

A continuación se van a presentar las distintas regularidades que han sido implementadas y que representan aisladamente reglas particulares de la percepción.

- **Regularidad de paralelismo de líneas.**

Esta regularidad esta basada en la hipótesis de que todos los pares de enlaces paralelos de la imagen deben corresponder a proyecciones de pares de aristas paralelas en el espacio. La expresión analítica para evaluar esta regularidad es formulada de la siguiente forma:

$$R_{\text{PARALELISMO}} = \sum_{i=1}^n \sum_{j=1}^m \omega_j [\cos^{-1}(u_1, u_2)]^2 \quad (5)$$

siendo:

$$\omega_j = \mu_{0,7^j}(\cos^{-1}(u_1', u_2')) \quad (6)$$

donde:

- n es el número direcciones, u orientaciones, definidas en la imagen.
- m es el número parejas de enlaces, orientadas en la dirección i-ésima, definida en la imagen.
- u_1 y u_2 representan los vectores unitarios tridimensionales según la primera y segunda arista de la pareja j-ésima de aristas evaluadas en la orientación i-ésima definida en la imagen.
- u_1' y u_2' representan los vectores unitarios de la imagen según el primer y segundo enlace de la pareja j-ésima de enlaces evaluados en la orientación i-ésima definida en la imagen.

Notar que la expresión dada en (6) es independiente del modelo generado durante el proceso de optimización, por lo que a efectos de reducir el coste computacional de algoritmo los valores ω_j deberán ser calculados exclusivamente al inicio del proceso de optimización. Es más, cuando ω_i sea nulo para un par de aristas, deberá omitirse el cálculo de la expresión $\cos^{-1}(u_1, u_2)$ para el par "i" de aristas.

También hay que destacar que esta regularidad, al igual que cualquier otra hipótesis basada en reglas perceptivas, no tiene porqué verificarse con probabilidad uno.

Una clara muestra de ello puede verse en la representación dada en la figura 32. Los enlaces $A'B'$, $C'D'$ y $E'F'$ de la imagen resultan ser paralelos en sí, y de acuerdo con la regla perceptiva anteriormente citada deberían corresponderse con tres aristas tridimensionales paralelas entre sí.

Sin embargo puede verse como, si bien dicha regla se verifica para las aristas AB y EF , no se cumplen con la arista CD .

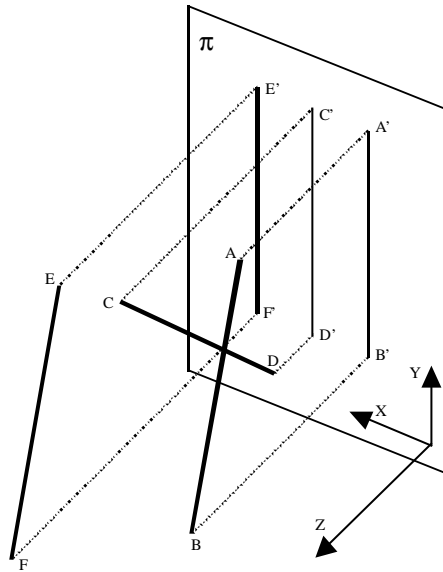


Figura 32. Regularidad de paralelismo

La regla de la percepción visual humana definida por algunos autores como la ley de la buena continuación, en la que nos basamos para definir esta regularidad, dice (regla 12 de Hoffman):

Si dos estructuras visuales mantienen una relación no accidental, agrúpelas y asígnelas un origen común

En la figura 33 se muestra un ejemplo de cómo este efecto provoca la creación de cuadros subjetivos. A la derecha, la mayor parte de los observadores ven un cuadrado blanco que tapa parcialmente a unos círculos negros. Sin embargo, a la izquierda se ven cuatro "L", y no se ve un cuadrado blanco que ocluye a cuatro cuadrados negros.

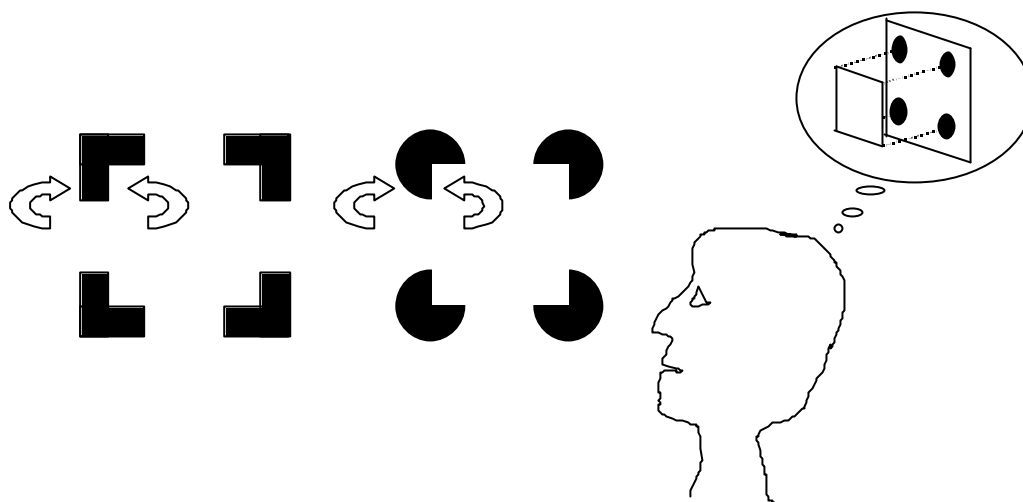


Figura 33. Ley de buena continuación. Paralelismo frente a simetría y simplicidad

La justificación se debe a que en la figura de la izquierda, los bordes marcados mediante dos flechas, por ser paralelos, se agrupan en un objeto único: una L. Esto quiere decir que los dos bordes, el interior y el exterior, pertenecen a la L. Dado que el borde interior pertenece a la L no puede pertenecer al mismo tiempo, excepto debido a un accidente visual, a otro objeto. En concreto no puede ser, excepto por un accidente visual, el borde de un rectángulo oclusivo. Por ello el vidente no construye un rectángulo subjetivo en la figura de la izquierda.

Pero además, el ejemplo representado en la figura 33, nos permite ofrecer un ejemplo de cómo algunas reglas de la percepción predominan frente a otras dependiendo de las características de la imagen. Una justificación posible se debe a que si suponemos que hay un cuadrado blanco oclusivo y cuatro cuadrados negros en la figura de la izquierda, reforzamos la simetría y la simplicidad; porque la L es una figura simétrica, pero es más compleja que un cuadrado parcialmente ocluido. Por otra parte, asumir que la oclusión ha generado una condición de paralelismo (entre las dos líneas marcadas con flechas) supone aceptar que el paralelismo es casual. El conflicto no aparece en la figura de la derecha, donde aceptar que hay un cuadrado oclusivo ayuda a interpretar como simétricas las cuatro figuras negras

En la forma de la derecha, los bordes señalados por las flechas distan mucho de ser paralelos. Tampoco se encuentran en ninguna otra relación no accidental, de modo que a priori no se agrupan como pertenecientes a un mismo objeto. Al final, el vidente

decide que es mejor que pertenezcan a un rectángulo oclusivo, en tanto que tiene la ventaja una vez más de agrupar en bordes paralelos.

Obviamente, también existe una justificación geométrica para esta regularidad. Se trata de que el tipo de representaciones en vista única más comunes en los dibujos de ingeniería son las axonométricas, que, por estar basadas en proyecciones paralelas, conservan invariante la condición de paralelismo.

- **Regularidad de planicidad de caras.**

En esta regularidad se propone que cada una de las caras del modelo tridimensional debe ser plana, es decir, todos los vértices que forman parte de una cara deben estar contenidos en un mismo plano.

Resulta evidente que la aplicación de esta regularidad precisa de una previa detección de las caras definidas en una imagen, que no es un problema trivial. El algoritmo de “detección previa de caras a partir de la imagen plana 2D” será tratado posteriormente en el capítulo “detecciones previas de la imagen”.

La evaluación de la planicidad de las caras que componen el modelo es realizada en dos pasos:

1. En primer lugar se busca el plano que mejor ajuste al conjunto de vértices que definen el contorno de la cara. La ecuación del plano vendrá dada por la expresión:

$$ax + by + cz + d = 0 \tag{7}$$

asumiendo arbitrariamente que $d = 1$. Los coeficientes a , b y c son obtenidos a partir de las coordenadas de los vértices que definen el contorno resolviendo el sistema lineal:

$$\sum_{i=1}^n \begin{bmatrix} X_i^2 & X_i Y_i & X_i Z_i \\ X_i Y_i & Y_i^2 & Y_i Z_i \\ X_i Z_i & Y_i Z_i & Z_i^2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \sum_{i=1}^n \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} \tag{8}$$

siendo “ n ” el número de vértices definen la cara.

Establecida la ecuación del plano se normalizan los coeficientes de manera que verifiquen:

$$\sqrt{a^2 + b^2 + c^2} = 1 \quad (9)$$

escalando d apropiadamente.

2. Obtenida la ecuación del plano con mejor ajuste, se obtienen las desviaciones en valor absoluto de cada uno de los vértices que definen el contorno mediante la expresión:

$$\Delta = \sum_{i=1}^n |aX_i + bY_i + cZ_i + d| \quad (10)$$

Las expresiones (8) y (10) son evaluadas para cada una de las caras definidas en el modelo. Y la suma de los valores obtenidos es el valor de la regularidad.

A modo de ejemplo vamos a evaluar la regularidad de planicidad de caras para un contorno cerrado formado por los vértices tridimensionales:

A (1, 0, 1)

B (0, 1, -1)

C (1, 1, 2)

D (1, -1, 0)

En primer lugar se obtiene el plano de mejor ajuste de acuerdo con la ecuación (8) y que para los vértices definidos quedaría planteado en la forma:

$$\begin{bmatrix} 3 & 0 & 3 \\ 0 & 3 & 1 \\ 3 & 1 & 6 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix}$$

de donde se obtiene los coeficientes:

$$a = 3/2$$

$$b = 1/2$$

$$c = -1/2$$

Normalizando vectores para verificar la ecuación (9) y escalando “d” apropiadamente, la ecuación del plano quedaría definida en la forma:

$$\frac{3}{\sqrt{11}} X + \frac{1}{\sqrt{11}} Y - \frac{1}{\sqrt{11}} Z + \frac{2}{\sqrt{11}} = 0$$

y aplicando la ecuación (10) para cada uno de los vértices se obtiene la evaluación de final de la regularidad:

$$\Sigma\Delta_i = \frac{16}{\sqrt{11}}$$

A efectos de la implementación informática, debe hacerse notar que existen términos a evaluar que resultan independientes del modelo generado en cada ciclo del proceso de optimización, en tanto que son evaluados exclusivamente a partir de las coordenadas (x,y) de los nodos de la imagen. Este es el caso de algunos de los coeficientes de la matriz dada en (8). Dichos términos deberán ser evaluados y memorizados una sola vez durante todo el proceso de optimización con el fin de reducir el coste computacional.

Forzar a que lo que *parecen* caras se comporten como caras (de ahí la exigencia de planicidad de caras), puede resultar tan evidente para el lector de este documento, que quizá es preciso remarcar que *vemos* caras porque así lo justifican las reglas de percepción visual humana. Cuando detectamos un circuito (un conjunto de líneas que pueden definir un camino) tendemos a considerarlo como una entidad que mantiene las mismas propiedades y que interpretamos de manera conjunta en el modelo. Es decir, tendemos a ver líneas y contornos del modo más continuo posible. Esto es debido a la

misma regla de la percepción visual (regla de buena continuidad) comentada con anterioridad (regla 12 de Hoffman) que se ve reforzada por una nueva regla de la percepción (regla 10 de Hoffman):

Construya superficies 3D, de modo que sean lo más lisas posibles

Esta norma se debe a reglas que rigen la interpretación de formas curvas tridimensionales, por lo que en este momento y dado que nuestro trabajo esta fundamentado en formas poliédricas, no entraremos en su justificación.

- **Regularidad de colinealidad de líneas.**

Esta regularidad establece que todos aquellos enlaces que resulten colineales en la imagen deben corresponder con aristas colineales del modelo. Los términos utilizados para la evaluación de esta regularidad son:

$$R_{\text{COLINEALIDAD}} = \sum_{i=1}^n \omega_i \max_{j=1, \dots, 4} \left[\frac{\det |v_j, v_{j+1}, v_{j+2}|}{\max (\|v_j - v_{j+1}\|, \|v_{j+1} - v_{j+2}\|, \|v_{j+2} - v_j\|)} \right]^2 \quad (11)$$

donde:

$$\omega_i = \mu_{0,7}^{\cos^{-1}(u_1', u_2')} \quad (12)$$

siendo:

- u_1' y u_2' los vectores unitarios del primer y segundo enlace de la imagen de la pareja i -ésima de enlaces colineales evaluados.
- $v_{j=1, \dots, 4}$ los cuatro vértices tridimensionales del par i -ésimo de aristas colineales.
- n , el número de pares de enlaces colineales.

La justificación de la ecuación (11) se puede encontrar en [Gusp00].

Notar desde el punto de vista de la implementación informática que la expresión dada en (12) es independiente del modelo generado durante el proceso de optimización, por lo que a efectos de reducir el coste computacional de algoritmo los valores ω_i deberán ser calculados exclusivamente al inicio del proceso de optimización. Es más,

cuando ω_i sea nulo para un par de aristas, deberá omitirse el cálculo de la expresión dada en (11) para el par “i” de aristas.

En la figura 34 se muestra un claro ejemplo de esta regularidad. Los enlaces $1',2'$ y $3',4'$ son colineales en la imagen, por lo que sus respectivas aristas tridimensionales 1,2 y 3,4 resultan colineales en el modelo. De igual forma, los enlaces $5',6'$ y $7',8'$ son colineales en la imagen, y sus respectivas aristas tridimensionales 5,6 y 7,8 resultan colineales en el modelo.

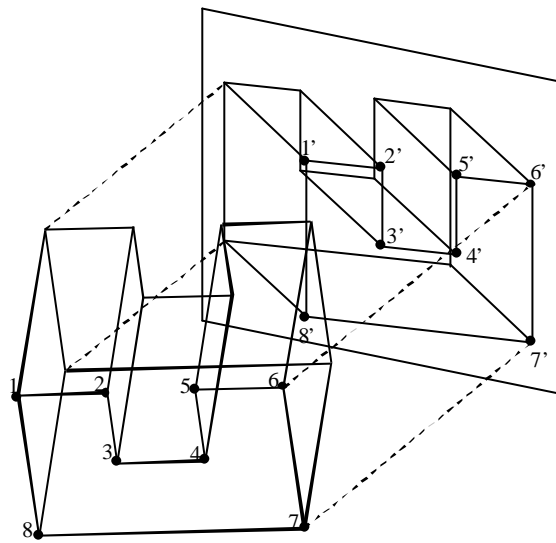


Figura 34. Regularidad de colinealidad

La regularidad de colinealidad también obedece a una regla de la percepción visual humana, la Regla de la colinealidad dice (regla 3 de Hoffman):

*Interpretamos siempre las líneas que son colineales en una imagen
como colineales en 3D*

No obstante debemos recordar una vez mas que esta regla debe representarse en términos de probabilidad, es decir, que representa propiedades que probablemente se verifiquen en un modelo, pero que ello no significa que tenga que verificarse estrictamente. En la figura 35a se muestra un contraejemplo de esta regularidad donde

dos enlaces colineales en la imagen (enlaces S_1 y S_2) no se corresponden con aristas colineales del modelo.

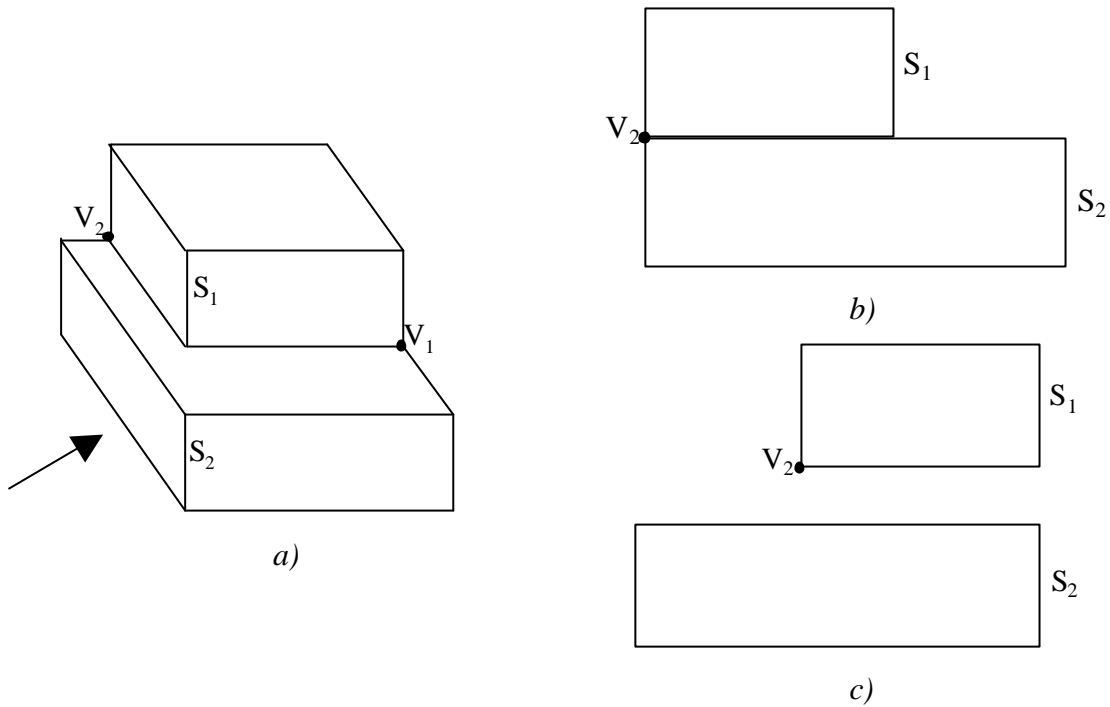


Figura 35. Contraejemplo de la regularidad de colinealidad

Para el lector de esta tesis resultará evidente que en el ejemplo mostrado en la figura 35a los enlaces S_1 y S_2 no son interpretadas como colineales en el modelo, sino que la aristas S_1 se encuentra bastante más alejada que la arista S_2 (figura 35b). Es decir, que estamos eliminando de nuestra interpretación una de las reglas de la percepción visual. Esto es debido a la influencia conjunta de todas las reglas de la percepción. Para que ambas aristas fuesen colineales en el espacio, la figura debería ser interpretada como dos cajas independientes, donde la caja superior floja sobre la inferior (figura 35c). Sin embargo, dicha interpretación estaría sujeta a un punto de vista accidental, es decir, si variásemos ligeramente el punto de vista del modelo la coincidencia de los nodos V_1 y V_2 no se daría en la imagen. Por consiguiente la ley de visiones genéricas predomina frente a la ley de la colinealidad.

También desde el punto de vista de la geometría es posible establecer una explicación a esta interpretación. Mientras que las aristas 1-2 y 5-6 de la figura 34 pertenecen a una misma cara del modelo, las aristas S_1 y S_2 de la figura 35 pertenecen a

caras distintas del modelo. Por tanto interpretaremos como colineales aquellas aristas del modelo que además de corresponderse con enlaces colineales en la imagen pertenezcan a una misma cara del modelo.

- ***Regularidad de ortogonalidad facial oblicua.***

Se define como cara ortogonal al contorno de aristas coplanarias en las que todas las uniones se realizan según ángulos rectos. Esta regularidad se basa en hecho de que al observar un contorno cerrado de aristas, si en dicho contorno alternan solamente dos direcciones, entonces dichas direcciones se consideran perpendiculares entre sí. Esta regularidad es una aplicación de la regla de la simetría no al objeto completo, sino a alguna o algunas de sus caras.

De acuerdo con esta regularidad, si una determinada cara de la imagen, presenta ortogonalidad oblicua, entonces, dicha cara probablemente sea ortogonal en el espacio. Si tenemos presente que el modelo tridimensional es representado en la imagen desde un punto de vista arbitrario, la ortogonalidad de una cara cualquiera del modelo se mostraría como una ortogonalidad oblicua como las que se muestran en la figura 36.

Las caras en proyección ortogonal oblicua pueden ser fácilmente detectadas en tanto que todas las aristas que conforman un contorno se encuentran alternativamente orientadas según dos direcciones que deben corresponderse con los ejes principales de la imagen (ver figura 36).

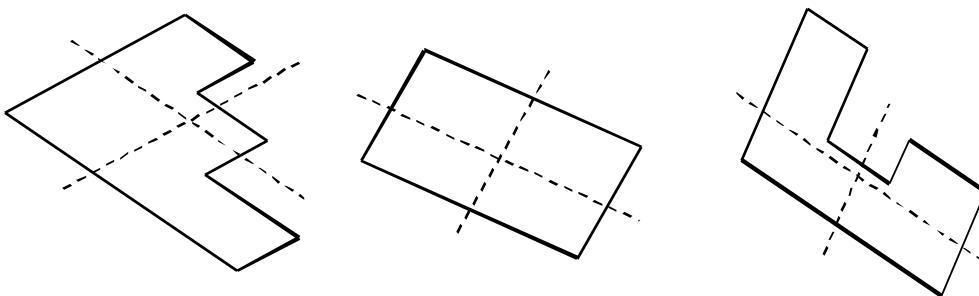


Figura 36. Proyección ortogonal oblicua de tres caras ortogonales

La evaluación de esta regularidad viene dada por la función:

$$R_{\text{ORTOGONALIDAD FACIAL OBLICUA}} = \sum_{i=1}^n (\omega_i \sum_{j=1}^m [\sin^{-1}(u_j \cdot u_{j+1})]^2) \quad (13)$$

siendo:

- u_j el vector unitario de la arista j -ésima en la cara i -ésima del modelo.
- m el número de aristas que definen la cara i -ésima del modelo
- n el número de caras definidas en el modelo.

La expresión dada en (13) viene afectada por un factor de confianza de ortogonalidad oblicua cuyo valor viene expresado por:

$$\omega_i = \mu_{0, 0.2} (\sigma (\beta_{j=1, \dots, m})) \quad (14)$$

siendo:

- μ un factor de confianza.
- β_j un parámetro que nos permite medir la forma alternativa de las orientaciones de aristas en la cara mediante la expresión:

$$\beta_j = (-1)^j [u_j' \cdot u_{j+1}'] [u_j' \times u_{j+1}'] \quad (15)$$

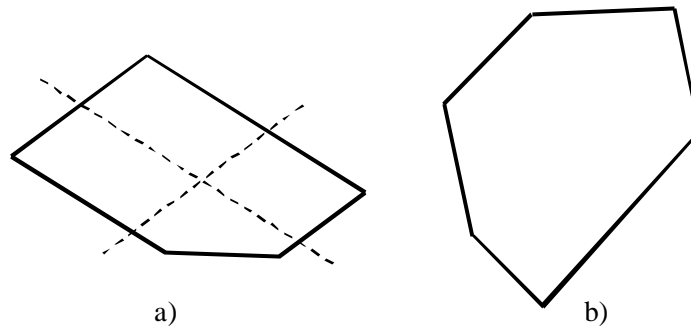
donde: u_j' y u_{j+1}' son los vectores unitarios de cada par de enlaces consecutivos en el contorno de la cara i -ésima.

- σ la desviación estándar de los parámetros β_j dada por la expresión:

$$\sqrt{\frac{\beta_j - \bar{\beta}}{m-1}} \quad \text{siendo } \bar{\beta} \text{ la media de los valores } \beta_j \quad (16)$$

La expresión (13) permite evaluar ortogonalidades parciales que pueden presentarse en algunas caras de la imagen (caras que las que algunas aristas no son

ortogonales), diferenciándolas de aquellas caras que no presentan ningún tipo de ortogonalidad (figura 37).



*Figura 37. Caras con:
a) ortogonalidad parcial b) sin ortogonalidad*

Con el fin de reducir el coste computacional en la ejecución del algoritmo de optimización, debe hacerse notar que la expresión (14) es independiente del modelo obtenido en cada ciclo del proceso de optimización en tanto que sólo depende de las coordenadas (x,y) de la imagen de partida. Por lo que su evaluación debe realizarse exclusivamente en el primer ciclo del proceso de optimización. Además para aquellas caras en las que la expresión (14) tenga valor nulo, deberá omitirse su evaluación en la expresión (13).

Nuevamente podemos encontrar la justificación de esta regularidad en las reglas de la percepción. Los observadores somos sensibles a la simetría, es decir, intentamos interpretar las imágenes como modelos simétricos. La *Ley de la Simetría* dice:

*Interpretamos una imagen de la manera que dé lugar a un modelo
lo más simétrico posible*

Existen innumerables ejemplos que muestran el cumplimiento de dicha ley. Uno de estos ejemplos se muestra en la figura 38, donde la elección de fondo y forma se apoya en la simetría (y no en el color): la simetría existente en las columnas X nos hace

preferir dichas formas frente a las existentes en las columnas Y (que interpretamos como fondo).

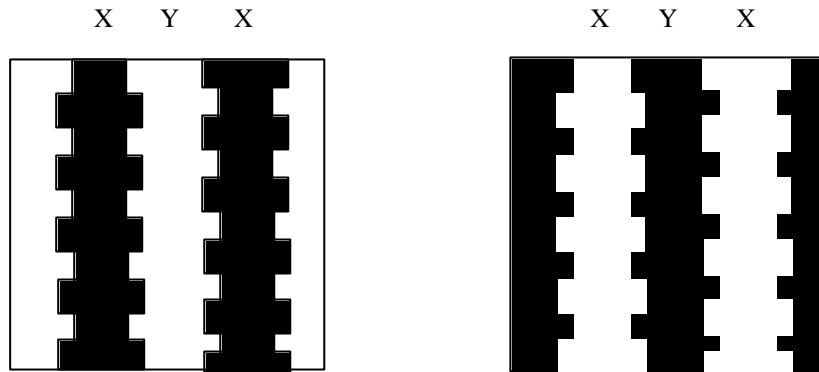


Figura 38. Ley de simetría

- **Regularidad de verticalidad.**

Esta regularidad esta basada en la asunción de que al dibujar seleccionamos puntos de vista tales que la dimensión de la altura de un objeto siempre es representada en la imagen como paralela al borde vertical del papel; lo que equivale a decir, en nuestro caso, que es paralela al eje Y del sistema de coordenadas de referencia. Es decir, la altura suele corresponder con la vertical de la imagen. Podríamos intuir que la visión también posee leyes de construcción basadas en la gravedad. Esta es una idea interesante que según cita Hoffman descubrió el psicólogo Irvin Rock.

Se establece como hipótesis que todo enlace vertical de la imagen (paralelo al eje Y) corresponde con una arista vertical en el espacio (figura 39).

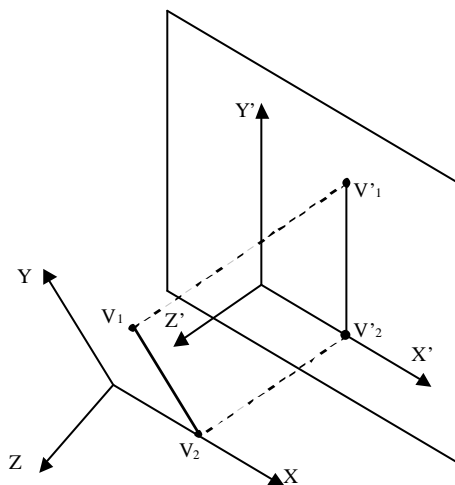


Figura 39. Regularidad de verticalidad

La expresión que evalúa la verticalidad de las aristas es de la forma:

$$R_{\text{VERTICALIDAD}} = \sum_{i=1}^n \omega_i [\cos^{-1}(u_{iY})]^2 \quad (17)$$

siendo:

$$\omega_i = \mu_{0,7^\alpha}(\cos^{-1}(u_{iY}')) \quad (18)$$

donde:

- u_{iY} es la componente “y” del vector unitario de la arista i-ésima del modelo.
- u_{iY}' es la componente “y” del vector unitario del enlace i-ésimo de la imagen.
- n es el número de enlaces en la imagen

Tal como hemos dicho, las implementaciones numéricas de las regularidades están tomadas de Lipson y Shpitalni. Sin embargo, en esta regularidad, hemos detectado cierto error en la valoración de dicha verticalidad. Mientras el ángulo que un determinado enlace de la imagen forma con el eje Y es correctamente determinado por la expresión propuesta:

$$\theta_1 = \arccos (v' \cdot u / |v'|) \quad (19)$$

siendo $v' = (V'_1, V'_2)$ y $u = (0,1)$ el vector unitario en la dirección del eje Y. La obtención de la verticalidad de una arista tridimensional es incorrectamente calculada mediante la expresión propuesta:

$$\theta_1 = \arccos (v \cdot u / |v|) \quad (20)$$

siendo $v = (V_1, V_2, V_3)$ y $u = (0,1,0)$ ya que se considera un vector unitario que no resulta paralelo al eje Y en el espacio tridimensional, dada la consecuente inclinación que el modelo sufre en el momento de la reconstrucción. El vector unitario debería venir expresado en función del ángulo θ_2 que es el ángulo obtenido como consecuencia del cabeceo, pero que resulta imposible de determinar mientras no se realice la reconstrucción (figura 40).

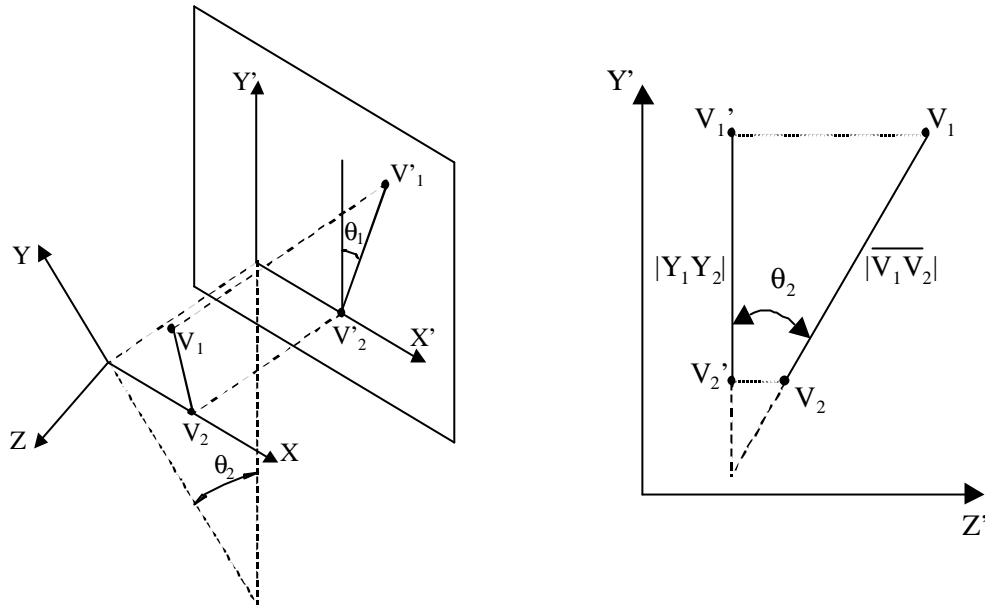


Figura 40. Error en la regularidad de verticalidad

Como consecuencia, dado el error detectado y la imposibilidad de solucionarlo, hemos intentado prescindir de esta regularidad para la reconstrucción de nuestros modelos. En el capítulo 7 de esta tesis se justificará con mayor amplitud las regularidades que han sido seleccionadas para la reconstrucción geométrica de los modelos analizados.

Notar desde el punto de vista de la implementación informática que la expresión dada en (18) es independiente del modelo generado durante el proceso de optimización, por lo que a efectos de reducir el coste computacional de algoritmo los valores ω_i deberán ser calculados exclusivamente al inicio del proceso de optimización. Es más, cuando ω_i sea nulo para un enlace de la imagen, deberá omitirse el cálculo de la expresión dada en (17) para la arista “i” del modelo.

- **Regularidad de isometría.**

Realmente se trata se una aplicación de la regla de las visiones genéricas que ha sido asumida en los dibujos de ingeniería. Esta regularidad esta basada en la asunción de que al dibujar seleccionamos puntos de vista tales que mantengan las proporciones entre las tres direcciones principales del espacio. Es decir, que las longitudes de las

entidades tridimensionales de un modelo resultan uniformemente proporcionales a sus longitudes en la imagen.

Este concepto es formulado matemáticamente mediante la expresión que computa la desviación estándar de la falta de uniformidad de las escalas:

$$R_{\text{ISOMETRIA}} = n \sum_{i=1}^n \sigma^2 (r_i) \quad (21)$$

donde:

- r_i representa la relación entre las distintas entidades espaciales y planas

$$r_i = (\text{longitud arista espacial } i\text{-ésima} / \text{longitud enlace plano } i\text{-ésimo}) \quad (22)$$

- “n” es el número de enlaces de la imagen
- σ la desviación estándar de las distintas relaciones r_i , dadas por la expresión:

$$\sqrt{\frac{\sum_{i=1}^n (r_i - \bar{r})^2}{n - 1}} \quad \text{siendo } \bar{r} \text{ la media de los valores } r_i \quad (23)$$

Si comparamos las dos interpretaciones de la misma imagen plana de la figura 41, se muestra el modo en que la falta de isometría puede conducir a objetos distintos.

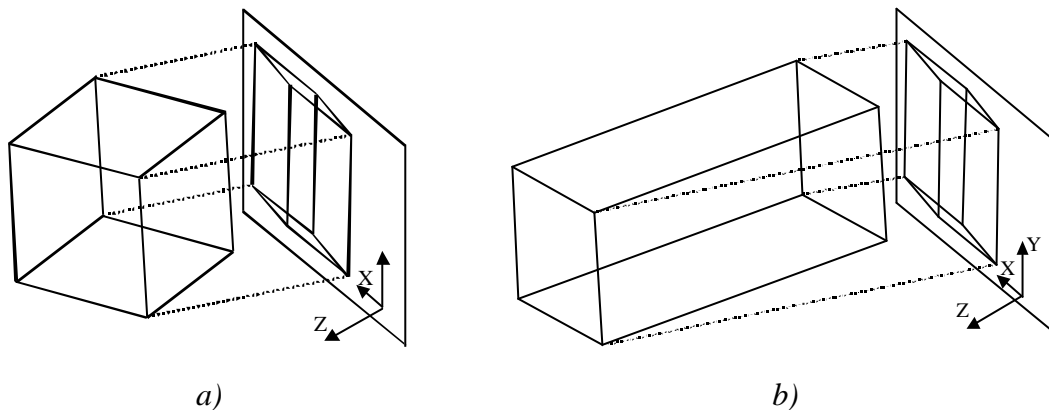


Figura 41. Consenso de proporcionalidad en las tres dimensiones
 a) Modelo cúbico b) Modelo paralelepípedo

De hecho, se puede considerar que la isometría permite escapar de un óptimo local (el prisma) y obtener el óptimo global (el cubo), ya que, perceptualmente, existe consenso al interpretar la imagen como la proyección del modelo cúbico que se representa en la figura 41a (visión estable y genérica) y no como proyección del modelo prismático representado en la figura 41b (proyección accidental del modelo).

Una prueba directa de la asunción de proporcionalidad que realizamos al interpretar las imágenes se muestra en el ejemplo de la figura 42 propuesto por el psicólogo Roger Shepard y citado por Hoffman. Las dos mesas son exactamente la misma, pero sometidas a una rotación, no tienen el mismo aspecto. Una parece larga y estrecha, la otra corta y ancha. Lo que ocurre es que nuestra visión está actuando de acuerdo con el convencionalismo que mantienen la proporcionalidad de las dimensiones de la mesa.

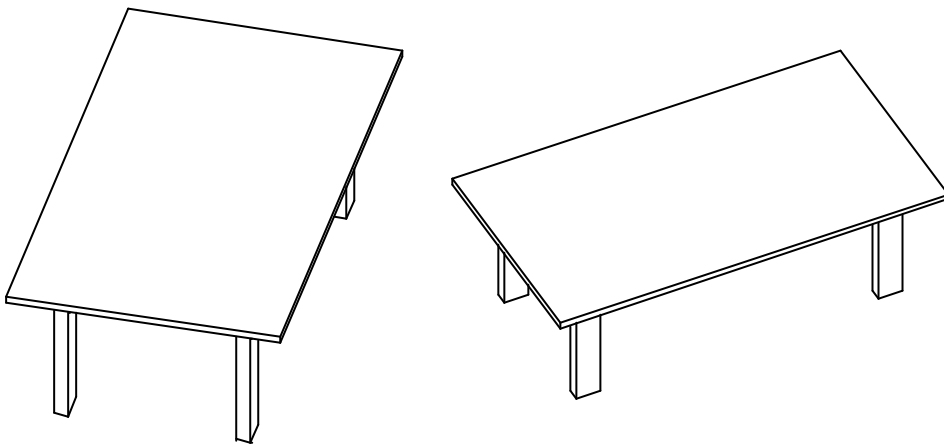


Figura 42. Efectos de la proporcionalidad de las dimensiones de una imagen

- ***Regularidad de mínima desviación estándar de ángulos (MSDA).***

La regularidad de mínima desviación estándar de ángulos, a la que en adelante nos referiremos como MSDA, establece que los ángulos formados por todos los pares de aristas que convergen en un mismo vértice deben ser iguales.

Los términos matemáticos propuestos para evaluar esta regularidad vienen dados por la expresión:

$$R_{MSDA} = \sum_{i=1}^n \sum_{j=1}^m n \sigma^2 (\cos^{-1}(u_1, u_2)) \quad (24)$$

donde:

- u_1 y u_2 representan los vectores unitarios de la primera y segunda arista del j -ésimo par de aristas que concurren en el i -ésimo vértice del modelo.
- m es el número de pares de aristas concurrentes en el vértice i -ésimo.
- n representa el número de nodos de la imagen
- σ es la desviación estándar de los ángulos que forman todos los pares de aristas que concurren en un mismo vértice del modelo:

$$\sqrt{\frac{a_j - \bar{a}}{m-1}} \quad \text{siendo } \bar{a} \text{ la media de los valores } a_j \quad (25)$$

Esta regularidad tiene su fundamento geométrico en la propiedad que verifican los poliedros regulares: todos los ángulos formados por aristas son iguales. Sin embargo, esta propiedad no puede ser incluida de manera indiscriminada a cualquier tipo de modelo. La aplicación de esta regularidad como componente principal de la función objetivo implica de la necesidad de detectar que el modelo representado en la imagen debe verificar esta condición.

Desde el punto de vista de la psicológico, esta regularidad también puede ser justificada mediante una de las reglas mas importantes de la percepción: la simplicidad. Es decir que tendemos a ver interpretar las imágenes de la manera mas simple posible, y dicha simplicidad implica interpretar como iguales los ángulos de un determinado modelo.

- **Regularidad de perpendicularidad de caras.**

En esta regularidad se establece que todas las caras adyacentes en la imagen (caras que compartan alguna arista en común) deben ser perpendiculares entre sí. El término utilizado para la evaluación es:

$$R_{\text{PERPENDICULARIDAD CARAS}} = \sum_{i=1}^n (\text{sen}^{-1}(n_1, n_2))^2 \quad (26)$$

donde:

- n_1 y n_2 son los vectores unitarios normales al par de caras i ésimo adyacentes del modelo
- n representa el número de caras adyacentes definidas en la imagen.

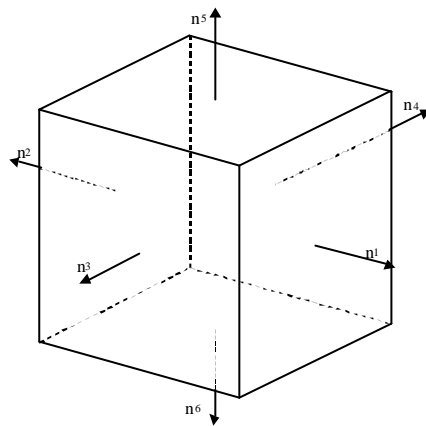


Figura 43. Vectores unitarios de superficie que determinan la perpendicularidad de caras

Esta regularidad es una particularización de la MSDA, puesto que supone considerar que todos los ángulos son de 90° .

Desde el enfoque geométrico, tiene sentido porque responde a una clase de poliedros que son bastante frecuentes en modelos asociados a objetos de uso práctico. De hecho, en la presente tesis se ha adoptado la denominación de normalón, propuesta por Dori [Dori95] para denominar a dicha clase de modelos. En consecuencia, si es una auténtica regularidad cuando se aplica a un objeto de tipología normalón.

- **Regularidad de ortogonalidad de esquinas.**

Esta regularidad establece que todas las ternas de enlaces de la imagen que pueden matemáticamente calificarse como proyecciones de esquinas ortogonales se corresponden con esquinas ortogonales del modelo tridimensional.

Para determinar matemáticamente cuando una terna de enlaces de la imagen puede corresponderse con la proyección de una esquina ortogonal, se establece el hecho de que el ángulo formado entre enlaces consecutivos no debe exceder de 90°. De acuerdo con ello se presentan las ocho combinaciones posibles de enlaces que podrían corresponder a proyecciones de esquinas ortogonales de un modelo.

Para cada uno de los nodos definidos en la imagen, se evalúa si los enlaces concurrentes en el nodo corresponden a alguna de las ocho configuraciones definidas como proyecciones de esquinas ortogonales (figura 44).

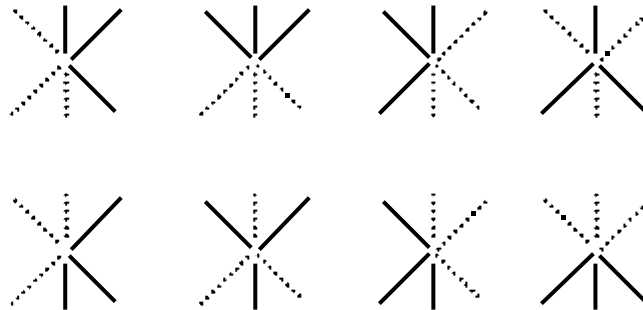


Figura 44. Posibles proyecciones de esquinas ortogonales

Los términos usados para evaluar la condición de ortogonalidad de esquinas son los siguientes:

$$R_{\text{ESQUINAS ORTOGONALES}} = \omega_i \sum_{i=1}^n [\sin^{-1}(u_1 u_2)]^2 \quad (27)$$

$$\omega_i = \begin{cases} 1 & \text{si } \beta \leq 0 \\ \mu_{0,0.1} & \text{si } \beta > 0 \end{cases} \quad (28)$$

$$\beta = \max_{8 \text{ variantes}} \left[\min_{3 \text{ pares}} (u'_1 u'_2) \right] \quad (29)$$

donde:

- u_1 y u_2 representan el primer y segundo vector unitario de todos los posibles pares de aristas que concurren en el i -ésimo vértice del modelo.
- n es el número de nodos de la imagen.

Notar que la expresión dada en (28) es independiente del modelo generado durante el proceso de optimización, por lo que a efectos de reducir el coste computacional de algoritmo los valores ω_i deberán ser calculados exclusivamente al inicio del proceso de optimización. Es más, cuando ω_i sea nulo para un nodo de la imagen, deberá omitirse el cálculo de la expresión dada en (27) para el nodo de la imagen evaluado.

Esta regularidad supone como esquina ortogonal aquella que esta definida por tres caras otogonales, es decir, aplicar esta regularidad implica que necesariamente deben concurrir tres aristas en el vértice evaluado, por lo que el coste de esta regularidad dependerá exclusivamente de los vértices que verifiquen esta condición. En otras palabras, si en algún vértice concurren un número de aristas distinto de tres, la regularidad no es evaluada en dicho vértice.

Esta regularidad también es una particularización de la MSDA, puesto que supone considerar que todos los ángulos son de 90° . Además es otra forma de describir a los objetos de tipo normalón. Frente a la ortogonalidad de caras, tiene la ventaja de que se puede aplicar sin necesidad de detectar caras, por lo que también puede actuar sobre modelos alámbricos.

- **Regularidad de ortogonalidad de líneas (MSDP).**

Esta regularidad, simplificada como MSDP, establece como hipótesis que todos los pares de enlaces que concurren en un mismo nodo deben corresponderse con aristas perpendiculares en el modelo tridimensional, salvo que dichos enlaces resulten colineales.

Los términos matemáticos usados para su evaluación son expresados en la forma:

$$R_{MSDP} = \sum_{i=1}^n \omega_i [\sin^{-1}(u_1, u_2)]^2 \quad (30)$$

siendo:

$$\omega_i = \mu_{0,7}(\cos^{-1}(u_1', u_2')) \quad (31)$$

donde:

- u_1 y u_2 representan el primer y segundo vector unitario de todos los posibles pares de aristas que concurren en el vértice i -ésimo del modelo.
- n representa el número de vértices del modelo.
- u_1' y u_2' representan el primer y segundo vector unitario de todos los posibles pares de enlaces de rectas que concurren en el nodo i -ésimo de la imagen.

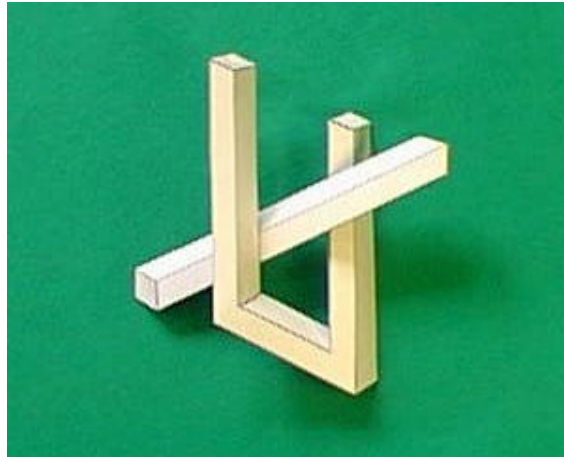
Esta regularidad puede ser considerada como una generalización de la ortogonalidad de esquinas, en tanto que evalúa la perpendicularidad de aristas en todos los vértices donde concurren mas de una arista. Es decir, la regularidad se evalúa en todos los vértices con independencia número de aristas que concurren en él. Además la evaluación es independiente de que la proyección de las aristas pueda o no corresponder con una esquina ortogonal.

Notar desde el punto de vista de la implementación informática que la expresión dada en (31) es independiente del modelo generado durante el proceso de optimización, por lo que a efectos de reducir el coste computacional de algoritmo los valores ω_i deberán ser calculados exclusivamente al inicio del proceso de optimización. Es más, cuando ω_i sea nulo para un enlaces de la imagen, deberá omitirse el cálculo de la expresión dada en (30) para el par de aristas evaluado del modelo.

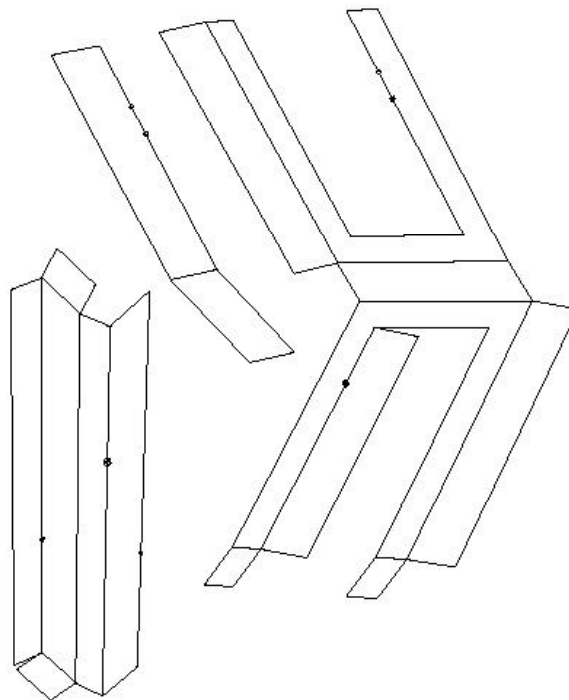
Las tres regularidades anteriormente comentadas, están basadas en el hecho de que siempre que existan tres direcciones diferentes en un dibujo, dichas direcciones se suponen perpendiculares entre sí. Este supuesto es de primordial importancia en la interpretación de dibujos desde el punto de vista psicológico. Esto puede justificarse a partir de la interpretación de figuras que usualmente son consideradas como *figuras imposibles*: figuras que desde el punto de vista de la percepción visual humana serían imposibles de construir materialmente.

Esto es debido a que el efecto de estas regularidades, unido con la regla de simplicidad, nos hace interpretar como imposibles aquellas figuras que supongan el incumplimiento de algunas de estas reglas de la percepción. Es decir, preferimos decir que la figura representada es una figura imposible antes de violar algunas reglas de la

percepción. Sugihara [Sugi] ha analizado algunas de estas figuras imposibles y dado soluciones para construir tales modelos. La figura 45 muestra uno de estos ejemplos.



a)



b)

*Figura 45. Modelo imposible
a) Modelo definido b) Desarrollo de solución posible*

COMPORTAMIENTO DE LOS ALGORITMOS DE OPTIMIZACIÓN EN LA RECONSTRUCCION GEOMÉTRICA

4

En el capítulo anterior se ha planteado la Reconstrucción Geométrica como un problema de optimización y se ha formulado la función objetivo en términos de regularidades.

El objetivo de este capítulo es determinar y ajustar el mejor algoritmo para optimizar la función objetivo. Es decir, elegir el método de optimización más adecuado, y ajustarlo del modo más conveniente para minimizar el incumplimiento de las regularidades.

El ser humano no suele tener problemas para determinar cual es el modelo que se debe percibir en una imagen dada. Esto significa, en términos de problema de optimización, que la optimización realizada por el ser humano *siempre* obtiene el óptimo global. Por tanto, ningún óptimo local es aceptable cuando se reconstruye con el auxilio de un algoritmo numérico de optimización matemática.

En consecuencia, por la propia naturaleza del problema planteado, se precisa determinar la validez de los distintos algoritmos de optimización para el problema específico de la Reconstrucción Geométrica; a fin de intentar garantizar que se empleen algoritmos capaces de alcanzar el óptimo global.

Además, los algoritmos de programación matemática siempre precisan ajustar los parámetros que determinan su comportamiento. Y, dado que se pretende que la reconstrucción sea automática, el ajuste debe ser hecho a priori y de modo totalmente automático.

En este capítulo se analizan los problemas derivados del empleo de la optimización para la Reconstrucción Geométrica de Sólidos; exponiendo los algoritmos utilizados por autores anteriores y los algoritmos implementados en esta tesis. También se detallan los ajustes que se han estudiado para conseguir un funcionamiento automático y eficiente de los algoritmos [Cone00].

4.1 Diferentes aspectos de los algoritmos de optimización.

La elección del algoritmo de optimización más adecuado pasa por considerar aspectos muy dispares y a veces contrapuestos [Press88]. Se han analizado los aspectos que, a nuestro entender, deben ser considerados más críticos: naturaleza de las magnitudes de reconstrucción, complejidad de la función de objetivo, restricciones y método de búsqueda acorde al espacio de soluciones.

4.1.1 Naturaleza de las magnitudes de reconstrucción.

Bajo este epígrafe se analizan las magnitudes que intervienen en el proceso de reconstrucción. Dichas magnitudes se pueden dividir en:

1. Parámetros de la reconstrucción: que representan magnitudes de valor fijo durante todo el proceso de optimización. En nuestro caso, destacan los coeficientes *peso* asignados a cada una de las regularidades del problema (parámetros α de la Función Objetivo descrita en el capítulo 3).
2. Variables de la reconstrucción: que representan magnitudes cuyos valores van cambiando conforme evoluciona el proceso de optimización. En nuestro caso, estas variables representan las coordenadas “Z” asignadas a cada uno de los vértices del modelo.

El conjunto de “n” coordenadas Z que corresponde a una solución válida se denomina un *punto solución* o simplemente un *punto*. La denominación de punto es el resultado de asimilar el conjunto de soluciones posibles a una región de un espacio n dimensional en donde las coordenadas de cada punto se corresponden con un conjunto de coordenadas Z que describen una posible solución (espacio de soluciones).

En función de la naturaleza de las variables, los procesos de optimización se clasifican en:

- a) Programación continua: cuando las variables pueden tomar cualquier valor real.
- b) Programación entera: cuando las variables sólo pueden tomar valores enteros.
- c) Programación discreta: cuando las variables sólo pueden tomar valores de un conjunto predefinido.

En nuestro caso los métodos de programación continua son eficientes y se adaptan bien a la reconstrucción geométrica mediante inflado de vértices.

Otro de los aspectos a considerar, relacionado con las magnitudes de reconstrucción, es el tamaño del problema. Cuando el problema es excesivamente grande se recurre a técnicas de reducción de tamaño, tales como la condensación de variables. En la reconstrucción geométrica no se emplea condensación, porque el tamaño del problema no constituye aún un obstáculo serio; dado que los desarrollos son experimentales y los casos tratados hasta la fecha resultan abordables sin necesidad de emplear ninguna técnica de reducción del tamaño.

4.1.2 Complejidad de la función objetivo.

La función objetivo suele ser una función escalar de las variables que intervienen en el problema a optimizar, y la condición habitual que se le exige a dicha función es que tome un valor mínimo en la solución óptima. Atendiendo a la complejidad de la función objetivo se puede distinguir entre:

- a) Programación lineal.
- b) Programación no lineal.

En la Reconstrucción Geométrica mediante inflado de vértices, la formulación de la función objetivo viene expresada en términos de regularidades; que como se ha mostrado en el capítulo anterior, resultan ser no lineales respecto a las coordenadas “Z” de los vértices. Por tanto el problema debe ser tratado por algoritmos de optimización propios para la programación no lineal.

4.1.3 Método de búsqueda acorde al espacio de soluciones.

Uno de los aspectos más importantes a considerar en la elección de un algoritmo de optimización es seleccionar un método de búsqueda apropiado para el espacio de soluciones. Los métodos de búsqueda pueden clasificarse en:

- a) Métodos de búsqueda exhaustiva. Estos métodos exploran todas las soluciones posibles por lo que sólo suelen ser válidos para espacios de soluciones reducidos. En

la Reconstrucción Geométrica éstos métodos no son viables más que en los casos más elementales.

- b) Métodos de búsqueda iterativa. Estos métodos establecen una estrategia de búsqueda por pasos. La mayoría de los métodos de búsqueda iterativa resuelven el problema buscando una nueva solución que mejore la actual, y repitiendo el proceso hasta llegar a la mejor solución posible; o hasta agotar las opciones de búsqueda. El proceso se puede formular matemáticamente mediante la expresión:

$$Z_{K+1} = Z_K + \alpha_K d_K \quad (32)$$

Es decir, que el vector de variables (punto solución) para la iteración $K+1$ (Z_{K+1}) se obtiene a partir del punto solución anterior (Z_K). La actualización del punto solución, dada por el producto $\alpha_K d_K$, consta de dos partes: d_K que es una *dirección de búsqueda* en el espacio n -dimensional de las variables y α_K que es la *longitud de movimiento* en esa dirección.

Encontrar una dirección de búsqueda consiste en elegir una dirección en el espacio de soluciones de forma que tenga la máxima probabilidad de mejorar el punto actual, conduciendo a un nuevo punto que disminuya el coste de la función objetivo.

La obtención del valor α_K es lo que se llama búsqueda unidimensional, porque coincide con la búsqueda de la solución óptima para un problema con una sola variable. Se trata de obtener el valor de α_K que mejore todo lo posible en la dirección elegida. En otras palabras, se trata de determinar cuanto hay que avanzar en la dirección d_K para obtener la mejor solución.

Los métodos que se emplean para determinar la dirección búsqueda se diferencian entre sí en función del tipo de información que utilizan para lograr su propósito. Se distinguen:

1. Método de búsqueda no guiada (de orden cero): donde se utiliza únicamente la función objetivo como valor de referencia. El más sencillo de estos métodos se denomina de búsqueda aleatoria, y consiste en elegir direcciones al azar en cada desplazamiento, comprobando que el valor de la función objetivo se reduzca con respecto a su valor en el punto de partida. Estos métodos son aceptables cuando

el problema a tratar es prácticamente desconocido. Es decir, cuando no se pueden establecer criterios de búsqueda que tengan una validez suficientemente general. En tales casos, encontrar una solución que mejore las ya conocidas puede ser un objetivo aceptable. Obviamente, estos métodos no son válidos en la Reconstrucción Geométrica, porque, como ya se ha dicho, sólo el óptimo global es aceptable.

Una alternativa, que es más costosa pero garantiza una mejor búsqueda, es la búsqueda exhaustiva de la dirección: se calculan todas las direcciones y se toma la mejor. No es tan costosa como una auténtica búsqueda exhaustiva porque sólo se explora el entorno cercano al punto actual; no se explora todo el espacio de soluciones.

Un método más eficiente derivado del anterior consiste en disponer algún criterio heurístico para determinar la dirección de búsqueda. La forma más elemental para hacer esto consiste en mantener la dirección de búsqueda mientras no nos conduzca a un punto peor (de mayor coste) que el último obtenido.

2. Métodos de búsqueda guiada: que pueden clasificarse en métodos de primer o segundo orden, dependiendo de que se precise la primera o segunda derivada de la función objetivo. Los métodos de búsqueda guiada son muy eficientes cuando se dispone de la información necesaria de *sensibilidades*. En nuestro problema, las sensibilidades serían los parámetros que midieran la influencia relativa de la modificación de cada una de las variables sobre cada una de las regularidades que penalizan la función objetivo. En la actualidad no se ha estudiado ninguna forma eficiente de obtener esta información, por lo que los métodos de búsqueda guiada aun no han sido aplicados a la reconstrucción.

4.1.4 Restricciones.

Las restricciones son condiciones que debe cumplir una solución para que pueda ser considerada válida. Estas condiciones se plantean como funciones de las variables, a las que se les exige tomar algún valor o mantenerse dentro de unos límites. Las restricciones pueden ser clasificadas como:

- a) Explícitas: cuando actúan directamente imponiendo condiciones a una variable, o a un grupo de ellas (optimización con restricciones).
- b) Implícitas: cuando se imponen condiciones, o *penalizaciones*, sobre magnitudes que dependen a su vez de las variables (optimización con penalizaciones o sin restricciones).

Las regularidades, como ya se ha dicho en reiteradas ocasiones, expresan analíticamente distintas reglas de la percepción. Es decir, representan los razonamientos que el ser humano realiza cuando visualiza e interpreta una imagen. Y es nuestra propia intuición la que cuantifica la importancia de cada una de estas reglas para conseguir una interpretación válida. Por consiguiente, y dado que el ser humano valora distintamente durante el proceso de interpretación el cumplimiento de las reglas en función de las características (tipología) de imagen visualizada, ninguna de ellas puede ser considerada como una restricción explícita, sino que deben ser formuladas en términos de probabilidades de aceptación de las reglas que representan. La formulación de estas regularidades como restricciones implícitas constituye el paso fundamental para la aproximación perceptual al problema de la Reconstrucción Geométrica.

Por su propia naturaleza, las regularidades pueden entrar en conflicto entre ellas. Como dice Hoffman, “sólo por accidente” puede dejar de ser cierta una regla. Pero los accidentes se pueden dar. Por tanto, hay que proveer un mecanismo que permita descartar aquellas regularidades que resulten ser un accidente en un caso particular. Ese mecanismo es difícil de introducir, y no ha sido estudiado en profundidad. Nosotros, nos hemos limitado a penalizar el incumplimiento de una regularidad, para no bloquear el proceso de reconstrucción cuando se produzca uno de esos accidentes.

4.2 Problemática de la optimización en la Reconstrucción Geométrica.

La Reconstrucción Geométrica presenta dos problemas fundamentales que le hace diferir de la mayoría de los problemas de optimización aplicados en ingeniería. Para ilustrar tales problemas se muestran algunos *subespacios de soluciones* de la función objetivo.

El concepto de subespacio de soluciones retoma la idea de hacer corresponder el conjunto de soluciones con una región de un espacio n-dimensional. Lo que permite

representar por medio de curvas de nivel los diferentes valores que toma la función objetivo en cada uno de los puntos de dicho espacio. Puesto que el espacio de soluciones tiene normalmente más de dos dimensiones no es posible representarlo completo. Se recurre, por tanto, a representar *cortes*, o subespacios, de dos dimensiones.

En las figuras 46, 47 y 48, que corresponden a un caso de espacio de ocho dimensiones, se han representado los subespacios definidos para dos vértices. Y se muestran, por medio diferentes niveles de gris (más oscuro cuanto mayor es el coste de la función objetivo) los distintos valores que toma la función objetivo en cada uno de los puntos de dicho subespacio.

En primer lugar, el punto de partida (la imagen 2D; o, lo que es lo mismo, el punto que corresponde a $z_1 = z_2 = \dots z_n = 0$) es un mínimo local (porque muchas regularidades son trivialmente nulas), lo que supone un inconveniente para iniciar el proceso de optimización (figura 46).

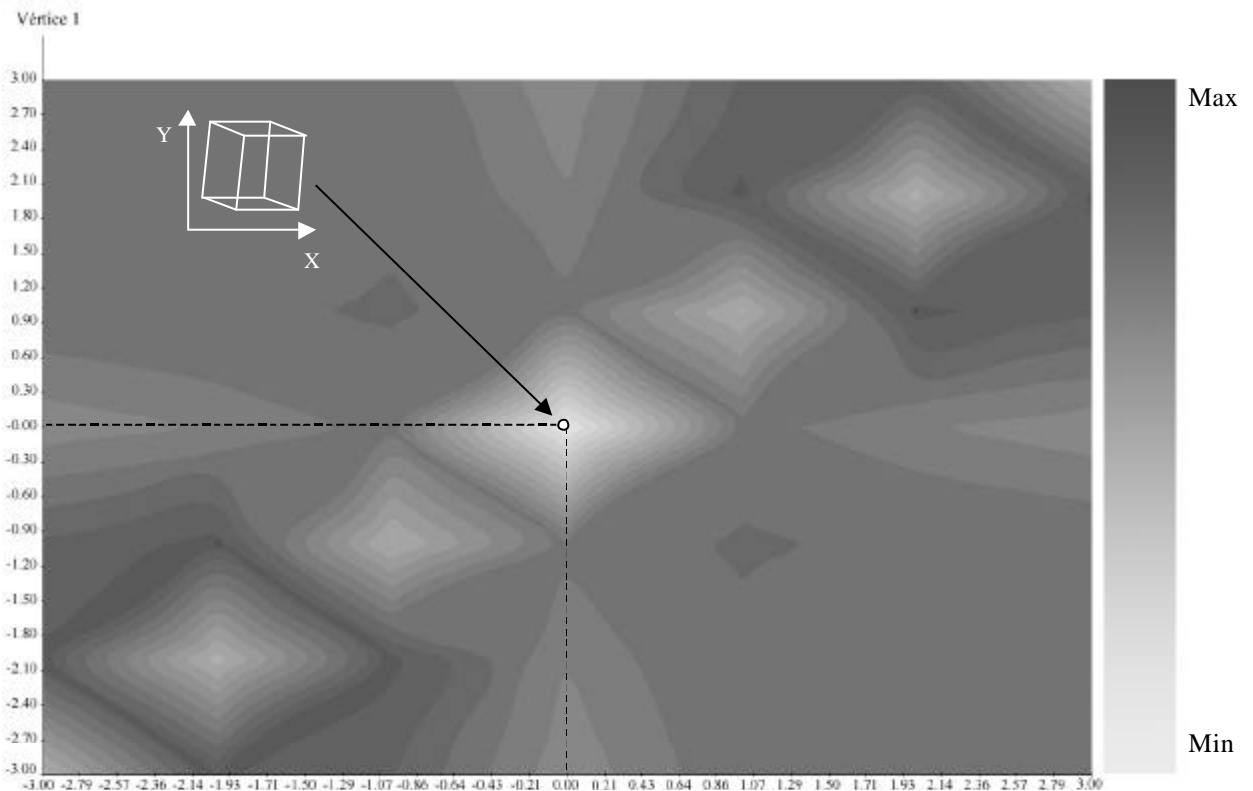


Figura 46. Mínimo local en la imagen

Pongamos dos ejemplos para comprender lo que acabamos de exponer. Por un lado, el paralelismo de líneas se verifica en la imagen, dado que todas las aristas correspondientes con enlaces paralelos son paralelas entre sí en la propia imagen, es decir con coordenada $z = 0$. Por otro lado la planicidad de caras también se verifica para cualquiera de las caras definidas en la imagen, dado que todas ellas están contenidas en el propio plano de la imagen.

En segundo lugar, en la mayoría de los problemas de diseño de ingeniería la consecución de mínimos locales da lugar a soluciones válidas (aunque no sean las mejores), porque ofrecen una reducción de la función objetivo; lo que suele corresponder con una mejora del producto que se está diseñando. Por el contrario, en la Reconstrucción Geométrica, los mínimos locales no son útiles, en tanto que ofrecen modelos tridimensionales no acordes con la percepción visual humana.

Una clara muestra de ello se ilustra en la figura 47 donde se representa el subespacio de soluciones de la función objetivo para la reconstrucción de un cubo. Como puede observarse, en dicho subespacio se distingue la existencia de un mínimo local (figura 47c). En él, el coste de la función objetivo es ligeramente superior al coste que se obtendría en el mínimo global, aunque la figura obtenida tiene una topología claramente diferente al modelo que ser humano interpretaría. Por consiguiente, sólo el modelo generado por el óptimo global resulta válido para la reconstrucción tridimensional y el resto de configuraciones ocasionadas por óptimos locales deben ser rechazadas.

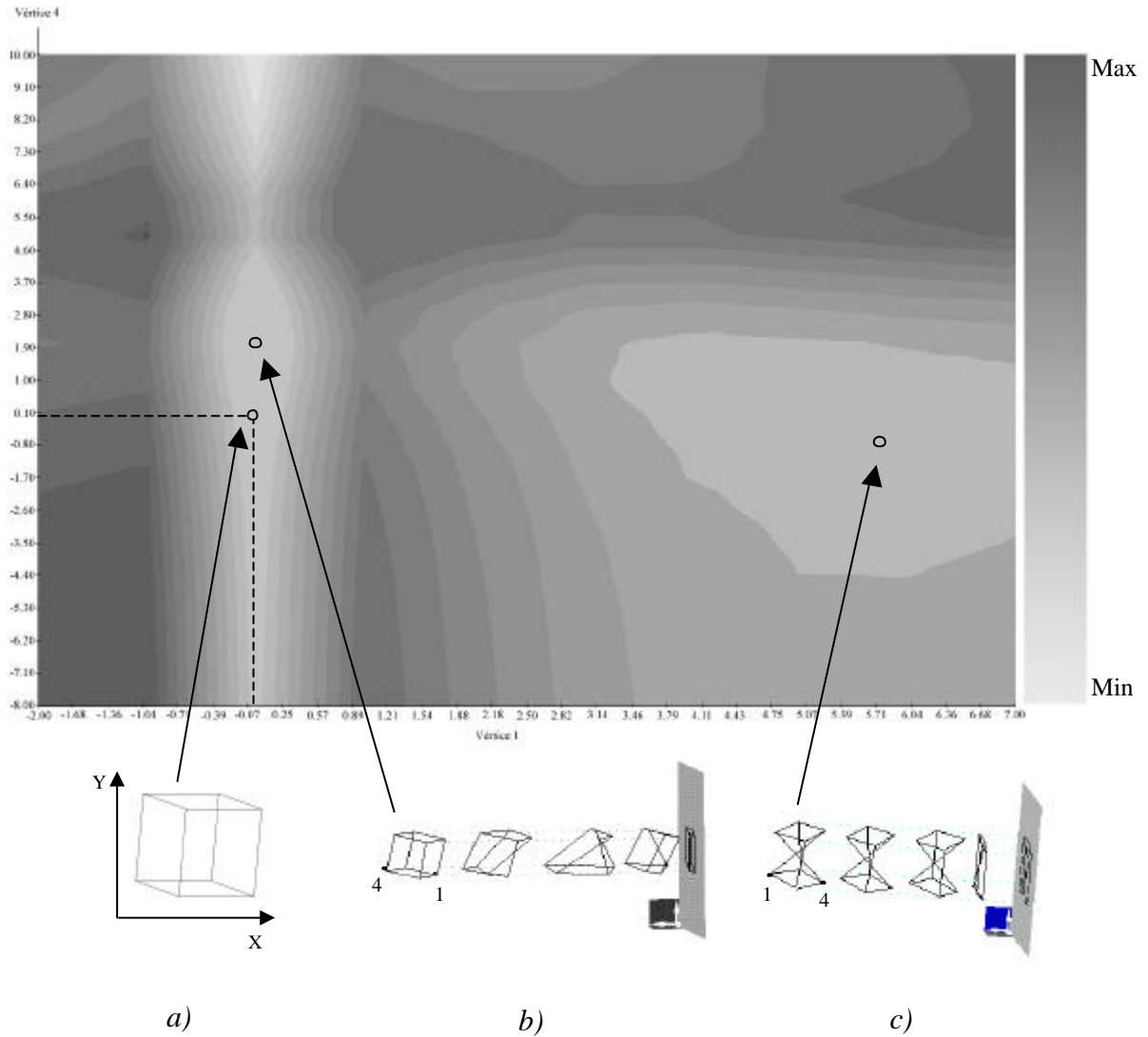


Figura 47. Distintos mínimos en la Reconstrucción Geométrica
 a) Imagen de partida b) Mínimo global c) Mínimo local

Es preciso resaltar otra característica de los procesos de optimización: los resultados obtenidos por los algoritmos de optimización dependen del ajuste de sus parámetros. Todos los algoritmos requieren el ajuste de ciertos parámetros, y el resultado final suele variar sensiblemente al modificarlos. Lo que dificulta considerablemente la tarea de automatización del proceso de reconstrucción, que es uno de los fines que se persigue en esta tesis. La figura 48 presenta los resultados obtenidos como consecuencia de una mala elección de los parámetros que controlan el algoritmo de optimización. Una longitud grande para el salto inicial, provoca que el algoritmo derive hacia una solución “distorsionada”, alargando el cubo hasta convertirlo en un prisma.

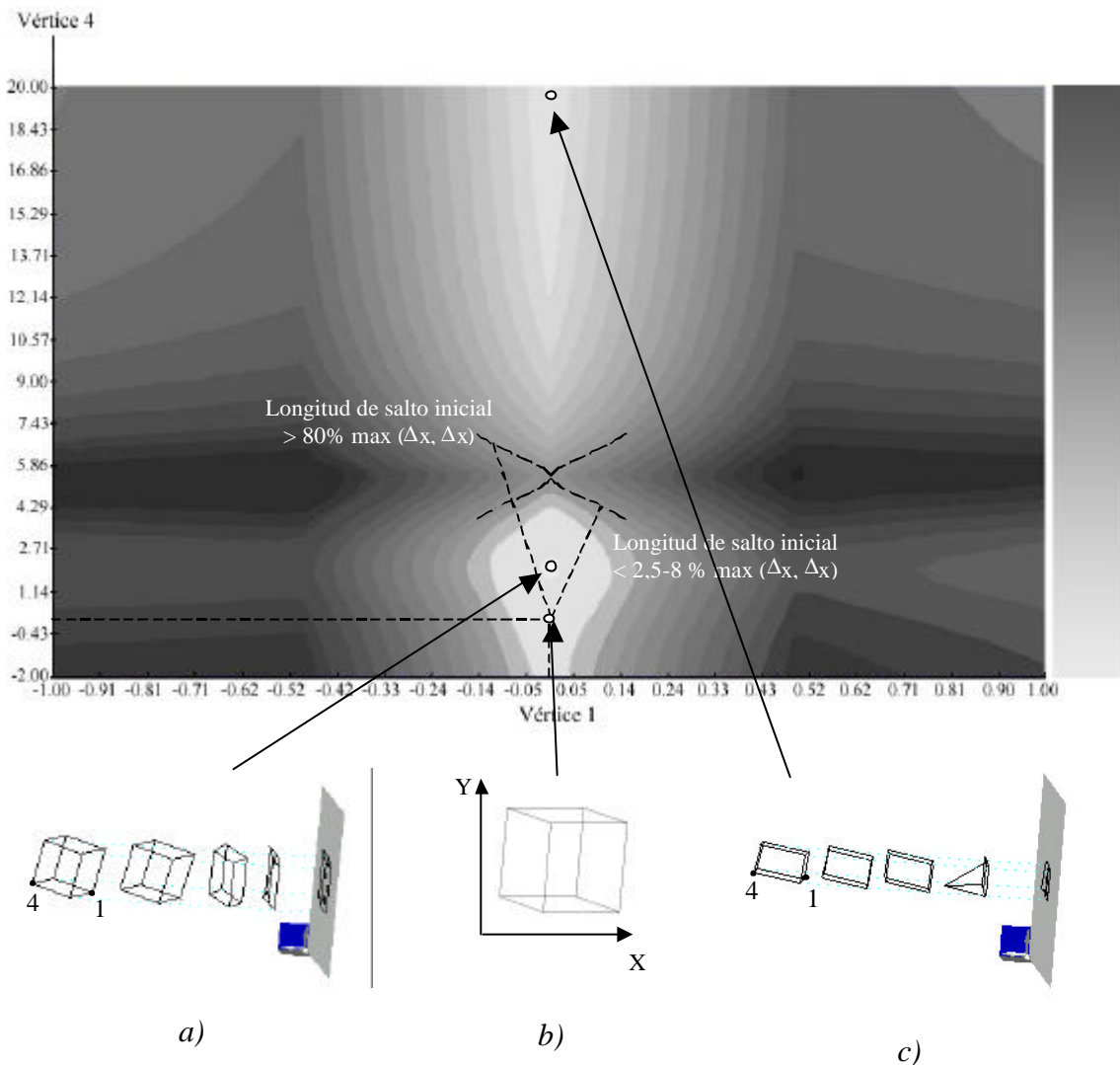


Figura 48. Mínimo local obtenido como consecuencia de una mala elección de los parámetros que dirigen el algoritmo de optimización
 a) Mínimo global b) Imagen de partida c) Mínimo local

4.3 Antecedentes de la reconstrucción mediante optimización.

La optimización de funciones que, como en el caso de la Reconstrucción Geométrica resultan ser no lineales y multidimensionales, puede realizarse con muchos algoritmos. De hecho, los distintos autores que han trabajado en el campo de la reconstrucción han propuesto distintos métodos.

El más apropiado en cada caso depende de las particularidades del problema; si bien, resulta imposible el seleccionar a priori el algoritmo perfecto para la obtención de

mínimos globales. Todos los métodos aplicados hasta la fecha en Reconstrucción Geométrica han sido iterativos o exhaustivos, continuos, no lineales y sin restricciones.

En la Figura 49 se muestra un cuadro resumen de los algoritmos de optimización utilizados hasta la fecha. Las características particulares de cada uno de estos algoritmos de optimización pueden ser consultadas en el anexo 2 de esta tesis.

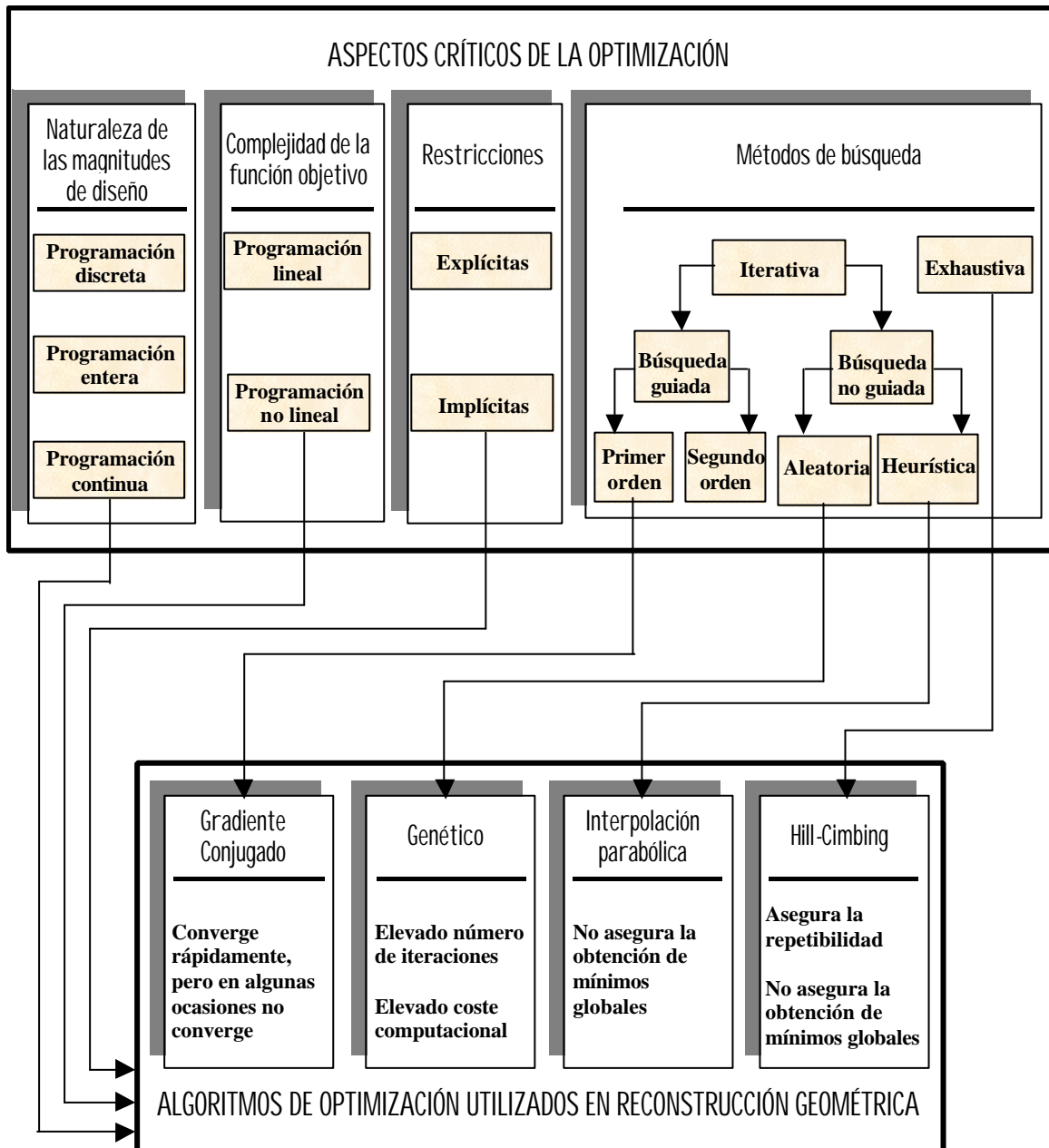


Figura 49. Clasificación de los algoritmos de optimización utilizados por autores anteriores

4.4 Algoritmos implementados.

En esta tesis se han explorado dos algoritmos de optimización. Por un lado se ha implementado el algoritmo Hill Climbing, para estudiar y contrastar los ejemplos propuestos por Marill y por Leclerc y Fischler. Por otro lado se ha implementado un algoritmo tipo Simulated Annealing, con el propósito de escapar de mínimos locales y encontrar la solución del modelo psicológicamente válida. A continuación se exponen los ensayos y resultados obtenidos a partir de dichos algoritmos.

4.4.1 Algoritmo Hill-Climbing.

Hill-Climbing fue el primer algoritmo utilizado para la reconstrucción mediante procesos de optimización. Se trata de un algoritmo iterativo “de descenso” donde la elección de la dirección de búsqueda se hace de forma exhaustiva, es decir, se calculan todas las posibles direcciones y se elige aquella que consigue un mayor descenso. El algoritmo presenta como principales ventajas su sencilla implementación, robustez y una “repetibilidad” en el proceso (siempre realiza el mismo camino de búsqueda) que permite analizar y contrastar los ejemplos propuestos por los distintos autores.

- *Estructura del algoritmo Hill-Climbing implementado.*

Dada una función $F = F(z)$ donde z es un vector de “n” variables (en nuestro caso las coordenadas “z” de los vértices del modelo), se parte de un valor inicial del vector: $z = (z_0, z_1, \dots, z_{n-1})$ evaluando la función para los $2n$ puntos que se obtienen por desplazamientos (o saltos) de longitud S de cada una de las variables:

$$\begin{aligned}
 z_1 &= (z_0 + S, z_1, \dots, z_{n-1}) \\
 z_2 &= (z_0 - S, z_1, \dots, z_{n-1}) \\
 z_3 &= (z_0, z_1 + S, \dots, z_{n-1}) \\
 z_4 &= (z_0, z_1 - S, \dots, z_{n-1}) \\
 &\dots\dots\dots \\
 z_{2n-1} &= (z_0, z_1, \dots, z_{n-1} + S) \\
 z_{2n} &= (z_0, z_1, \dots, z_{n-1} - S)
 \end{aligned}
 \tag{33}$$

seleccionando el desplazamiento z_i que hace mínima la función objetivo:

$$z_i / F(z_i) = \text{mínimo} (F(z_1), F(z_2), \dots, F(z_n)) \quad (34)$$

El proceso se repite de manera iterativa para el nuevo vector seleccionado, mientras no se alcance alguno de los criterios de parada establecidos:

- La función objetivo no disminuye su coste en un cierto valor mínimo.
- El número de iteraciones realizado sobrepasa al máximo permitido.

Cuando se verifica alguno de los criterios de parada, el ciclo se inicia de nuevo, modificando el valor de “S”. El valor del “salto” S se reduce en cada ciclo, para obtener soluciones cada vez más refinadas: $S_1 > S_2 > \dots > S_m$

El número de ciclos “m” se debe elegir para alcanzar un compromiso entre precisión final obtenida y tiempo de cálculo utilizado. La optimización finaliza cuando concluye el m-ésimo ciclo.

A cada uno de los valores asignados al parámetro S_i , que representa la longitud del movimiento (incrementos o decrementos de las variables), se le ha denominado “longitud de paso”. Y cada uno de los “m” ciclos realizados por el algoritmo de optimización se ha denominado “escalón”.

El diagrama de flujo del algoritmo puede verse en la figura 50 donde:

- “ n_e ” representa el escalón evaluado.
- “ N_e ” representa el número total de escalones.

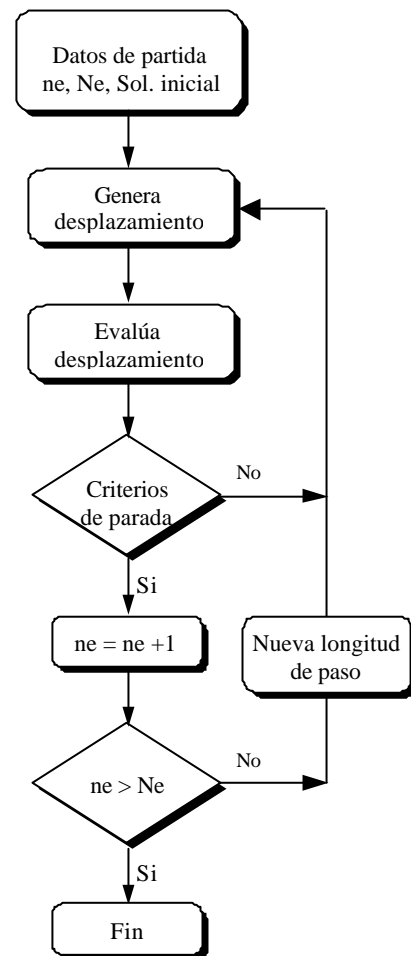


Figura 50. Diagrama de flujo del algoritmo de optimización Hill Climbing

En la figura 51 se muestran las gráficas de la evolución del coste para distintos escalones del algoritmo de Hill Climbing. En la figura 51a se muestran los escalones

propuestos por Marill (1, 0.5 y 0.1). En la figura 51b se muestran los escalones definidos por Leclerc y Fischler (0.125, 0.0625, 0.03125, 0.015 y 0.007).

La función analizada, corresponde a un caso de ocho variables (reconstrucción de un cubo), y optimizada fijando un número máximo de 800 iteraciones por escalón.

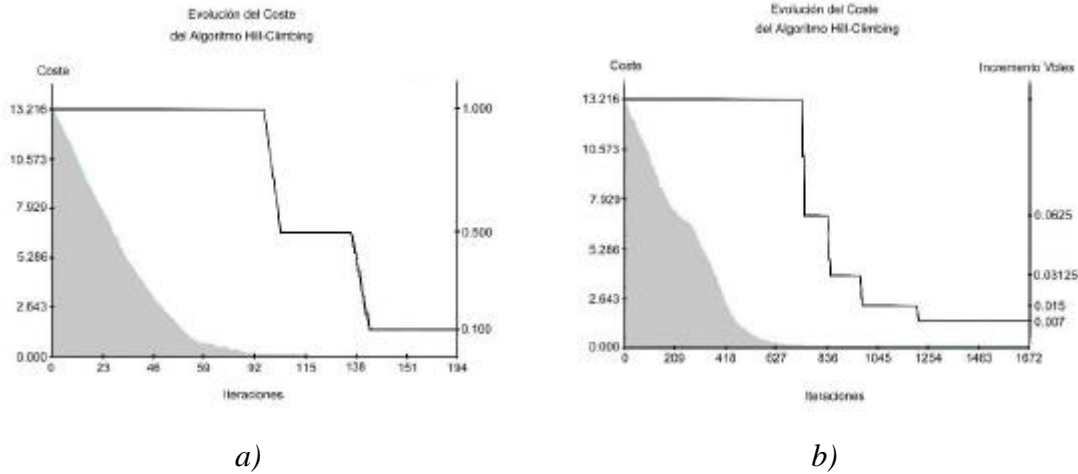


Figura 51. Diagrama de evolución del coste para escalones definidos por
a) Marill b) Leclerc y Fischler.

En las representaciones puede observarse el carácter siempre descendente de la función coste definido por los criterios de aceptación del algoritmo, el cual puede ocasionar la obtención de mínimos locales como soluciones al problema de optimización. También puede observarse como la función decrece de manera más brusca durante el primer escalón (escalón con mayor longitud de paso), suavizándose su monotonía para escalones posteriores donde la longitud del paso se ve reducida. Se puede hablar de un primer escalón de optimización, seguido de otros de refinado.

- **Ensayos y conclusiones.**

En primer lugar se ha optado por elegir un conjunto de parámetros reducido y coherente que permita configurar el algoritmo de forma que se mantenga una cierta analogía entre el problema de optimización y el de reconstrucción. Para justificar esta decisión hay que destacar que se ha valorado la “amigabilidad” de la interacción con el usuario. Por tanto, se ha buscado un conjunto de parámetros que, más que permitir un

control directo del algoritmo, permitan que el usuario intuya los efectos que la modificación de cualquier parámetro deberán tener sobre el proceso de búsqueda.

Los parámetros que hemos definido para controlar el funcionamiento del algoritmo son:

- Longitud de paso inicial (S_1).
- Reducción de paso, expresada en tanto por ciento del paso actual: $S_{i+1} = \% S_i$
- Longitud de paso final (S_m).

Al definir estos parámetros se crea una dependencia entre ellos tres y un cuarto parámetro que el usuario no controla directamente: el número de pasos. Con el agravante de que la dependencia puede dar lugar a soluciones inconsistentes debido a que el número de escalones debe ser entero. En tal caso, se ha optado por redondear por exceso. Se ha constatado que la “calidad” de la solución final es más fácil de controlar si se introduce el tamaño de paso final que si se introduce el número de escalones. Para justificarlo, basta hacer notar que la longitud del paso final es, a *grosso modo*, el error que pueden llegar a tener las coordenadas z del modelo final. En otras palabras, creemos que un usuario está más interesado en controlar la precisión final de las medidas del modelo, que en controlar el número de iteraciones necesarias para conseguirlo.

Los parámetros que definen el equilibrio del algoritmo son:

- Número máximo de iteraciones por escalón.
- Disminución del coste de la función en al menos un tanto por ciento fijado.

Las dos diferencias entre las propuestas de Marill y Leclerc y Fischler en lo referente al algoritmo de optimización, radican en el número de escalones, así como en el valor cuantitativo de los mismos. Por un lado Marill proponía tres escalones en el algoritmo dados por los valores 1, 0.5, 0.1. Por otro lado, Leclerc y Fischler proponían cinco escalones dados por los valores 0.125, 0.0625, 0.03125, 0.015, 0.007. Proponían un mayor número de escalones, argumentando que grandes diferencias entre escalones consecutivos conducían al algoritmo de optimización hacia mínimos locales.

Al analizar los trabajos de Marill y Leclerc y Fischler observamos que en ambos casos los escalones tenían valores absolutos. Es decir, que los incrementos iniciales de coordenadas eran 1, o 1,125 *unidades de longitud*. Por lo que no se tenía en cuenta el

tamaño absoluto de las imágenes de partida. En ningún caso se indicaba ningún proceso previo de normalización del tamaño de las imágenes. Sin embargo, era evidente que algún tipo de normalización es necesario. Y, de hecho, se pudo contrastar que todos los ejemplos ensayados por Marill tenían sus dimensiones definidas en un rango máximo $\Delta x = 7.79$ y $\Delta y = 8.16$. Es decir, que la longitud máxima del paso oscilaba en un 12% de las dimensiones máximas de la imagen. Los ejemplos ensayados por Leclerc y Fischler se encontraban dentro de un rango máximo $\Delta x = 4.78$ y $\Delta y = 3.50$, es decir, que la máxima longitud de paso oscilaba con un máximo del 2% las dimensiones máximas de la imagen.

La propuesta obvia de mejora y automatización del algoritmo fue normalizar el valor absoluto del paso por medio de un parámetro interpuesto: se propuso definir el valor del paso como un porcentaje del máximo rango de la imagen de partida. De forma que el paso inicial se defina como un porcentaje “p” de las dimensiones máximas de la imagen:

$$S_1 = p * \max(\text{RangoX}, \text{RangoY}) \quad (35)$$

Para introducir un conjunto más amplio de mejoras en la automatización del proceso de optimización, se ha realizado un total de 90 ensayos variando los distintos parámetros que configuran el algoritmo de optimización para intentar definir un conjunto de valores que permita una optimización que equilibre calidad y tiempo de computo. La función analizada, corresponde a un caso de ocho variables (reconstrucción de un cubo) para la que se ha definido una longitud de paso final (precisión de la solución) del 0,15% de la máxima dimensión de la imagen. En las figuras 52, 53 y 54, se muestra el coste de la función objetivo (Coste), y el tiempo computacional necesario para la ejecución del algoritmo (Tiempo), en función del número de pasos definidos en cada escalón, el número de escalones definidos (escalones), la longitud del paso inicial y la reducción de la longitud del paso entre escalones consecutivos.

De la observación de dichas gráficas se han extraído las siguientes conclusiones:

1. Se ha comprobado que es una buena estrategia normalizar el tamaño de los escalones. El valor fijado para la longitud de paso el primer escalón, es dependiente de las dimensiones de la imagen a reconstruir. Por tanto se evita establecer una lista de escalones, haciendo esta dependiente de las dimensiones de la imagen. Se asume

que el máximo valor de la longitud de paso inicial debe estar muy por debajo del 100% del max (RangoX, RangoY). En concreto y a partir de una visión genérica de todas las gráficas que se muestran en las figuras 52, 53 y 54, entre el 2% y el 3% de la máxima dimensión de la imagen parece ser un buen valor para la optimización en un tiempo de computo aceptable. Como puede observarse para longitudes de paso superiores al 3%, aumentan la velocidad de ejecución del algoritmo pero se pierde calidad en el modelo y se aumenta el peligro de escapar hacia un óptimo local.

2. La ley de decrecimiento de las longitudes de paso no tiene gran influencia. Se ha ensayado una ley del tipo $S_{i+1} = R * S_i$, es decir una ley donde la longitud de paso para cada escalón (S_{i+1}) es determinada a partir de la longitud de paso del escalón inmediato anterior (S_i). Sin embargo, a partir de las figuras 52, 53 y 54 puede observarse que una reducción drástica de la longitud del paso, en concreto superior al 50%, puede provocar la pérdida de la convergencia del proceso de optimización. Por otro lado, una ley de decrecimiento excesivamente lenta (inferior al 25%) provoca un aumento importante en el coste computacional, dado que aumenta el número de escalones y en consecuencia el número de iteraciones a realizar por el algoritmo. Como conclusión, queda definido un intervalo (25%-50%) donde el proceso de optimización es acorde con la convergencia y el tiempo de computo.
3. Además comparando las gráficas dadas para igual longitud de paso inicial e igual reducción del paso (es decir para cada final de gráficas mostrada en las figuras), puede observarse como el aumento del número máximo de iteraciones por escalón aumenta considerablemente el tiempo de computo sin producir mejoras importantes en la solución final. Por ello hemos definido como suficiente un máximo de 100 iteraciones por variable y escalón.

Por consiguiente y como resumen de las conclusiones que se han expuesto creemos que definir una longitud de paso inicial entorno al 2.5% max (RangoX, RangoY) podría servir para definir un algoritmo automático independiente de las dimensiones de la imagen de partida. De igual forma, la definición de una longitud de paso final del 0,15% es suficiente para conseguir un refinado acorde a las dimensiones del modelo sin excesivo tiempo de computo. Además entre cinco y diez escalones, puede considerarse un buen rango para obtener una buena solución acorde con el tiempo preciso para su computo, lo que establece una reducción de la longitud de paso entorno

al 50%. Un número máximo de 100 iteraciones por variables y escalón ha sido considerado suficiente.

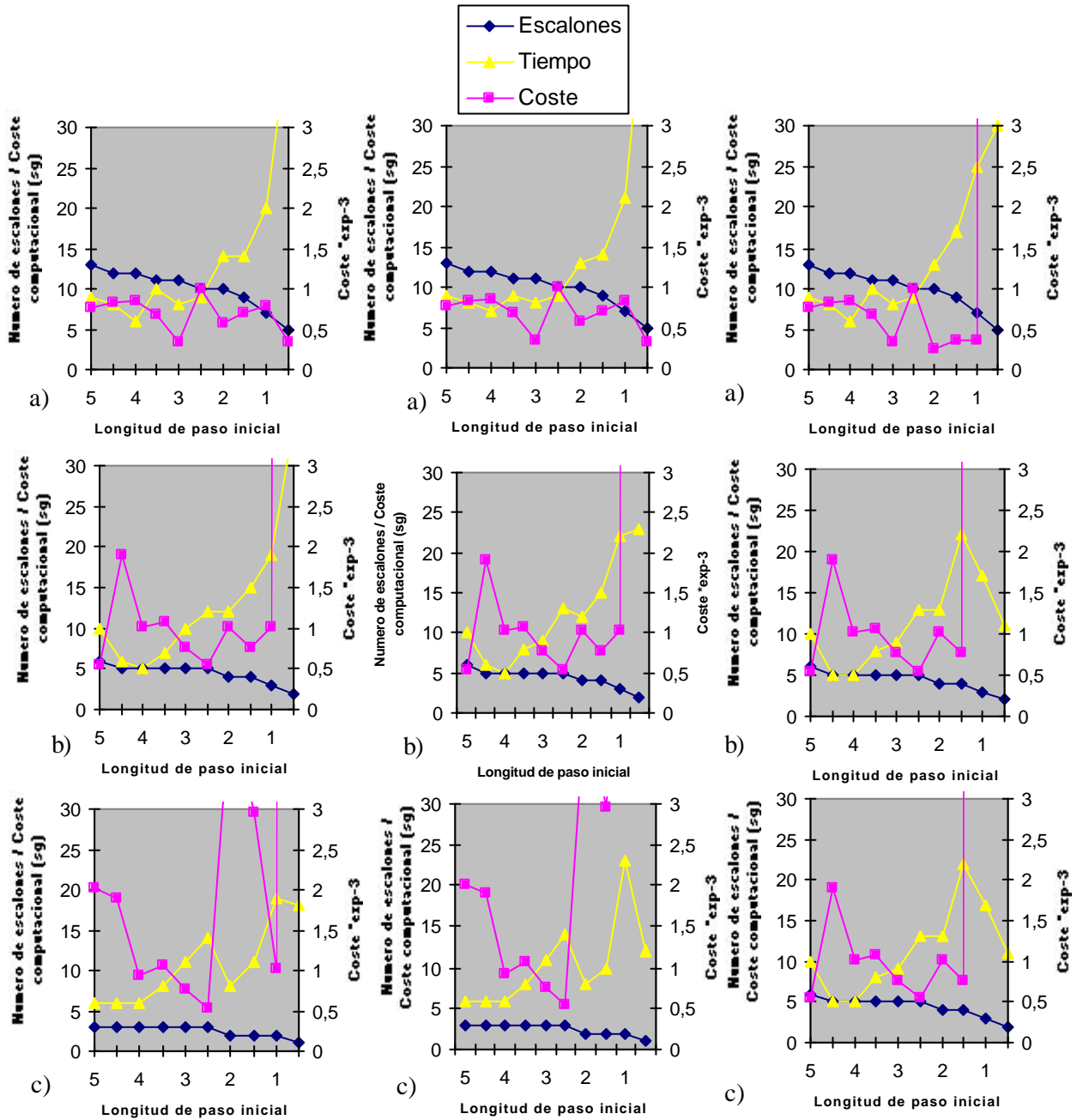


Figura 52. Número máximo de pasos por escalón 300, reducciones de paso: a) 75% del actual b) 50% del actual y c) 25% del actual

Figura 53. Número máximo de pasos por escalón 200, reducciones de paso: a) 75% del actual b) 50% del actual y c) 25% del actual

Figura 54. Número máximo de pasos por escalón 100, reducciones de paso: a) 75% del actual b) 50% del actual y c) 25% del actual

4.4.2 Algoritmo Simulated Annealing.

La técnica del Simulated Annealing, a la que a partir de ahora nos referiremos como SA, es un método iterativo de búsqueda aleatoria dirigida basada en métodos enumerativos y que utiliza información adicional para guiar la búsqueda. Esta técnica surge en 1983 con la publicación del artículo de Kirkpatrick y otros [Kirk83].

El término de “Simulated Annealing”, que se podría traducir al castellano por “Recocido simulado”, hace referencia a un tratamiento térmico como es el “recocido” que según la acepción recogida en el Diccionario de la Lengua Española significa “caldear los metales para que adquieran de nuevo la ductilidad o el temple que suelen perder al trabajarlos”. En el desarrollo de este epígrafe se justificará la razón de utilizar este símil cuando realmente se está tratando de resolver un problema de optimización.

- ***Estructura del algoritmo SA.***

Todo algoritmo SA se puede caracterizar por una serie de parámetros, unos específicos del problema en cuestión y otros genéricos, independientes de la naturaleza del problema.

Los parámetros específicos del problema son:

- Un espacio de soluciones S , que es el conjunto de todas las soluciones del problema.
- Una función objetivo que evalúa la bondad de cada solución dada.
- Un mecanismo de generación de soluciones, que permite el desplazamiento desde una solución a otra de su entorno. Entendiendo como entorno de una solución el conjunto de soluciones a las que podemos llegar desde ella, aplicando el mecanismo de generación.

Los parámetros genéricos que son independientes del tipo de problema que se esté resolviendo y que controlan el funcionamiento del algoritmo son:

- Temperatura inicial.
- Ley de evolución de la temperatura.
- Criterios de parada.

Al conjunto de parámetros genéricos se les denomina “esquema de enfriamiento”.

Las principales diferencias de este algoritmo con un algoritmo tipo Hill-Climbing son el método de generación de nuevas soluciones y el criterio establecido para la aceptación de los desplazamientos.

En la fase de generación de nuevas soluciones, Hill Climbing evalúa los desplazamientos de todas las variables (algoritmo exhaustivo) mientras que el algoritmo SA simplemente selecciona una de dichas variables al azar (algoritmo aleatorio).

En la fase de evaluación del desplazamiento, Hill Climbing sólo busca el máximo descenso de la función coste, mientras que en el algoritmo SA se usa el criterio metrópolis [Metro53] (que será analizado posteriormente), introduciendo la probabilidad de aceptar un desplazamiento desfavorable, es decir, aunque este suponga un incremento de la función coste.

Dada una función $F = F(z)$ donde z es un vector de “n” variables (en nuestro caso las coordenadas “z” de los vértices del modelo), se parte de un valor inicial del vector: $z = (z_0, z_1, \dots, z_{n-1})$ evaluando la función para un nuevo vector z_1 obtenido mediante el mecanismo generador de soluciones que se tratará posteriormente, de acuerdo con un valor de desplazamiento (o paso) de longitud L_T (longitud de paso para una temperatura T).

El proceso se repite de manera iterativa para el nuevo vector seleccionado, mientras no se alcance alguno de los criterios de parada establecidos denominados criterios de equilibrio:

- El número de aceptaciones de desplazamientos supera un cierto valor.
- El número de iteraciones realizado sobrepasa al máximo permitido.

Cuando se verifica alguno de los criterios de parada, el ciclo se inicia de nuevo, modificando el valor de “ L_T ” y el de la temperatura T de acuerdo con una ley de enfriamiento que como posteriormente se justificará influye en la aceptación de una solución dada. El valor del “salto” L_T se reduce en cada ciclo, para obtener soluciones cada vez más refinadas.

Por similitud con el algoritmo de Hill Climbing, a cada uno de los valores asignados a L_T , y que representa la longitud del movimiento (incrementos o decrementos de las variables) se le ha denominado “longitud de paso” y a cada uno de

los ciclos realizados por el algoritmo de optimización se le ha denominado “escalón”. El diagrama de flujo del algoritmo puede verse en la figura 55.

De esta manera el sistema va pasando por una serie de escalones hasta que se verifica alguno de los criterios de parada (denominado criterio de congelación):

- El número de aceptaciones de desplazamientos supera un cierto valor.
- El número de iteraciones realizado sobrepasa al máximo permitido para el último escalón.
- **Mecanismo de generación de soluciones.**

Todo algoritmo de optimización precisa de un mecanismo para modificar las variables, buscando aquellas que determinarán la solución final. El proceso por el cual dichas variables son modificadas se conoce con el nombre de mecanismo de generación de soluciones.

En nuestro caso se ha establecido la estrategia de selección aleatoria de una variable, que será modificada también de manera aleatoria mediante adición o sustracción de la longitud de paso. Es decir, cada nueva solución ensayada es obtenida a partir de la solución actual de la forma:

$$Z_{K+1} = Z_K + \alpha_K L_T \quad (36)$$

donde la dirección de desplazamiento es seleccionada mediante los valores de:

- “ Z_K ” elemento del espacio de soluciones elegido al azar.
- “ α_K ” con valor aleatorio 1 o -1 , que modifica aleatoriamente mediante sustracción o adición el valor asignado a “ Z_K ”.

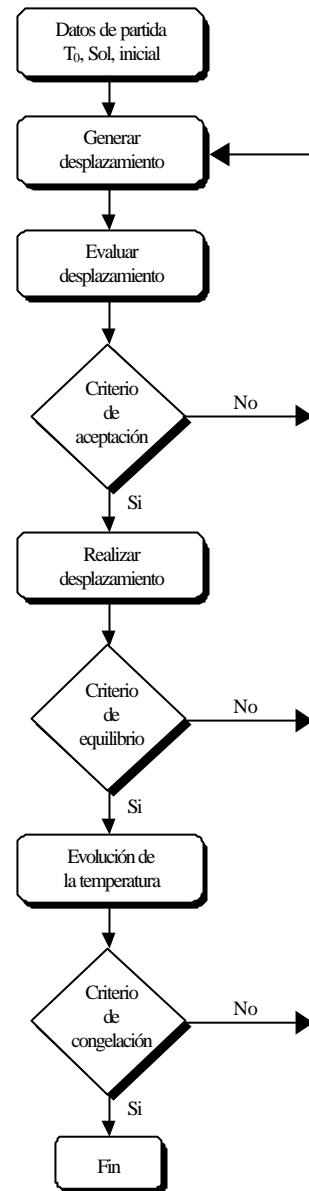


Figura 55. Diagrama de flujo del algoritmo S.A.

y la longitud de desplazamiento viene dada por:

- L_T longitud de paso en el escalón de temperatura T.
- ***Aceptación de una solución dada.***

En la mayoría de los algoritmos de optimización el cálculo del coste para unas determinadas variables supone el único criterio para la aceptación o no de las soluciones planteadas. La técnica del SA difiere de éstas técnicas tradicionales de mejora iterativa en su capacidad para escapar de mínimos locales gracias al empleo del criterio Metrópolis para la aceptación de los desplazamientos.

Metrópolis [Metro53] presentó uno de los primeros métodos de cálculo que aprovecha las capacidades computacionales de los ordenadores. En su algoritmo se proponía el cálculo del incremento del coste ΔC provocado por el desplazamiento realizado en el espacio de soluciones, de manera que:

- Si $\Delta C < 0$ entonces, ya que el desplazamiento conduce al sistema a un estado de menor coste, dicho movimiento es permitido.
- Si $\Delta C > 0$ entonces el desplazamiento es permitido con una probabilidad:

$$\exp(-\Delta C/T) \quad (37)$$

siendo T la temperatura del sistema para ese escalón.

Para ello se generaba, utilizando una distribución uniforme, un número aleatorio ξ comprendido entre 0 y 1. Si $\xi < \exp(-\Delta C/T)$ se produce el desplazamiento hasta la nueva posición. En caso contrario el desplazamiento es rechazado.

La adopción de tal criterio de aceptación permite a los algoritmos SA escapar de posibles mínimos locales que serían insalvables en el caso de algoritmos descenso (figura 56).

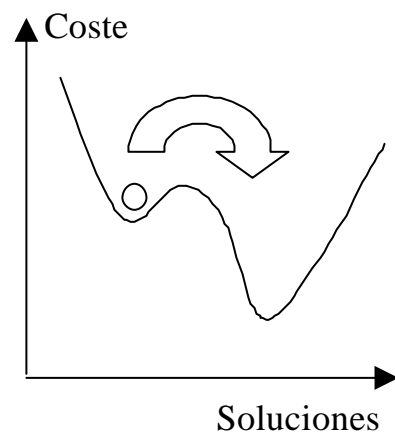


Figura 56. Criterio Metrópolis

- **Temperatura inicial.**

Son muy diferentes las estrategias propuestas para determinar la temperatura inicial T_0 . Por ejemplo escoger T_0 de tal forma que prácticamente sean aceptadas todas las transiciones, es decir que $\exp(-\Delta C/T_0) \approx 1$ donde ΔC hace referencia al incremento del coste de la función en dos transiciones consecutivas.

En nuestro caso se ofrecen dos alternativas de cálculo:

- En primer lugar la expresión propuesta por Kouvelis [Kouv92]:

$$T_0 = \frac{\Delta C^{(+)}}{\ln(\chi)^{-1}} \quad (38)$$

donde $\Delta C^{(+)}$ es el incremento de coste medio de los desplazamientos que provocan incremento de coste y χ es el coeficiente de aceptación, definido como el cociente entre el número de transiciones aceptadas y el número de transiciones intentadas.

Para calcular el incremento de coste medio se realiza una exploración para asegurar que se dan tales incrementos positivos y evitar que la temperatura inicial sea cero.

- Un segundo método de cálculo es el propuesto por Kirkpatrick y otros [Kirk83] quienes, basándose en una regla empírica, proponen escoger un valor grande de T_0 y realizar cierto número de transiciones, de tal forma que, si el valor de χ obtenido es menor de una cierta cantidad, por ejemplo 0,8, entonces se dobla el valor de T_0 , procediendo así sucesivamente hasta verificar la condición impuesta. En nuestro caso se ha aceptado como valor inicial T_0 para el bucle de transiciones, la décima parte del coste de la solución de partida supuesta ésta no nula. En caso de serlo se comienza el proceso de generación de soluciones hasta conseguir cumplir tal condición.

A partir de los ensayos que hemos realizado, hemos llegado a la conclusión de que el método de Kouvelis permite obtener una temperatura inicial que oscila en un intervalo que depende directamente del coeficiente de aceptación adoptado, que tiende a la temperatura inicial determinada por el método de Kirkpatrick conforme el coeficiente de aceptación tiende al 100%. Para valores inferiores del coeficiente de aceptación, la temperatura inicial determinada por Kouvelis suele oscilar entorno al 20% de la determinada por Kirkpatrick.

- **Ley de enfriamiento de la temperatura.**

Existen diferentes leyes de evolución de temperatura que pueden ser clasificadas en función de su complejidad (ver [Pres88]). Un primer grupo agruparía a los esquemas sencillos de enfriamiento. Dentro de este grupo, aparecen expresiones de la forma:

$$T_M = c T_{M-1} \quad (39)$$

oscilando el valor “c” entre 0.50 y 0.99. Dicha expresión puede ser también definida en función de la temperatura inicial como:

$$T_M = c^M T_0 \quad (40)$$

con lo que se obtiene una curva de enfriamiento de tipo potencial, produciéndose un enfriamiento más rápido a temperaturas elevadas, que posteriormente se hace más lento a bajas temperaturas.

Otro tipo de ley de enfriamiento consiste en establecer de antemano un número determinado de escalones, por ejemplo K, de tal forma que se mantiene el salto de temperatura constante en la forma:

$$T_M = \frac{K - M}{K} T_0 \quad (41)$$

Otras leyes de enfriamiento utilizadas por diversos autores son:

$$T_{M+1} = \frac{T_M}{1 + c T_M} \quad (42)$$

$$T_{M+1} = \frac{T_0}{1 + c M} \quad (43)$$

En nuestro caso se ha optado por seleccionar la ley de enfriamiento dada en (39). Su justificación se debe a que, como se demostrará más tarde, la ley de enfriamiento debe asemejarse en lo posible a la evolución de coste de la función objetivo. Y como se analizará a partir de la figura 57, el comportamiento de la función objetivo indica claras oscilaciones en las etapas iniciales de la optimización, para suavizarse conforme avanza el proceso.

- **Criterio de equilibrio.**

Se denomina criterio de equilibrio, al criterio de parada que establece el final del proceso de evaluación de soluciones para cada uno de los escalones de temperatura definidos. La elección más sencilla para determinar el criterio de equilibrio consiste en elegir un valor K independiente del número de iteraciones a realizar en cada escalón de temperaturas. Otras propuestas más elaboradas establecen que para cada valor T_K se deberían realizar un número mínimo de transiciones η_{\min} fijo y prefijado de antemano. Sin embargo al aproximarse T al valor 0, las transiciones son aceptadas cada vez con una probabilidad menor de tal forma que el número de transiciones tendería al infinito conforme la temperatura tendiese a cero, por lo que el número de transiciones debe estar acotado superiormente.

En esta línea Kirkpatrick propone un número máximo de iteraciones igual al múltiplo del número de variables a resolver. Otro sistema para definir el número de iteraciones por escalón de temperaturas es establecer el concepto de *época*. Se define época como un número de transiciones con un número fijo de transiciones aceptadas, siendo el coste de una época el coste de la última configuración de la misma. En el momento que el coste de una época está a una cierta distancia de las épocas precedentes, el número de transiciones acaba. De esta manera la condición de término de las transiciones para un determinado escalón de temperatura queda ligada a las fluctuaciones de la función de coste observadas en la misma.

El criterio de equilibrio tiene relación con la complejidad del problema. Es decir, para resolver un problema complejo será necesario realizar un número elevado de transiciones por escalón de temperatura. Como índice de la complejidad del problema se emplea el tamaño de la *clase de vecindad* (K). Definiendo clase de vecindad como el conjunto de soluciones a las que se puede acceder desde una solución dada aplicando el mecanismo de generación.

En nuestro caso se han definido dos condiciones límites:

- El número total de intentos realizados a una temperatura dada ($e1 * K$).
- El número máximo de intercambios aceptados a una temperatura dada ($e2 * K$).
- ***Criterio de congelación.***

El criterio de congelación es definido como la condición de parada del algoritmo. En nuestro caso se ha fijado un porcentaje máximo de rechazos al intentar los desplazamientos, de tal manera que si el número de rechazados superase este valor límite el algoritmo se detendría.

Dada la característica del criterio de aceptación en la fase inicial del enfriamiento, cuando la temperatura es elevada se produce un elevado número de transiciones que provoca un incremento del coste. Progresivamente, al avanzar el proceso de enfriamiento, el sistema va gradualmente evolucionando hacia soluciones de menor coste; hasta que prácticamente se estabiliza en torno a la solución final (figura 57).

En la figura 57 también puede observarse otro fenómeno que con frecuencia aparece en la utilización del algoritmo SA: la solución final obtenida no siempre es la mejor de las que se han explorado. La causa es que la implementación real de los algoritmos SA no satisface todos los requisitos teóricos de convergencia de los mismos. Esto es debido al criterio de aceptación Metrópolis que, si bien corresponde a un mecanismo muy útil para escapar de mínimos locales, en ocasiones puede hacer que el proceso de optimización se dirija hacia soluciones de mayor coste. Como consecuencia, no siempre la solución final obtenida es la mejor solución por la que ha pasado el sistema.

Existen varias propuestas que producen ciertas mejoras prácticas. En nuestro caso, se ha almacenado la mejor solución obtenida por el sistema a lo largo del proceso de búsqueda; de tal forma que si la solución final no es la mejor, el sistema recupera de la memoria la mejor solución por la que se ha pasado.

Además, y dada la naturaleza aleatoria de este algoritmo, resulta fácil deducir que la solución obtenida en el problema de optimización planteado no tiene por qué ser siempre la misma. Es decir, si un mismo problema es analizado reiteradamente, el punto

solución obtenido mediante optimización variará entre un conjunto de soluciones mas o menos amplio, en función de la complejidad de la función objetivo. Es por ello que en este algoritmo hemos implementado la posibilidad de ejecutar reiteradamente el proceso de optimización. La solución obtenida en cada uno de estos procesos es almacenada. Y cuando todos los procesos de optimización definidos han finalizado, se selecciona de todas las soluciones almacenadas, aquella que resulte más válida (de menor coste).

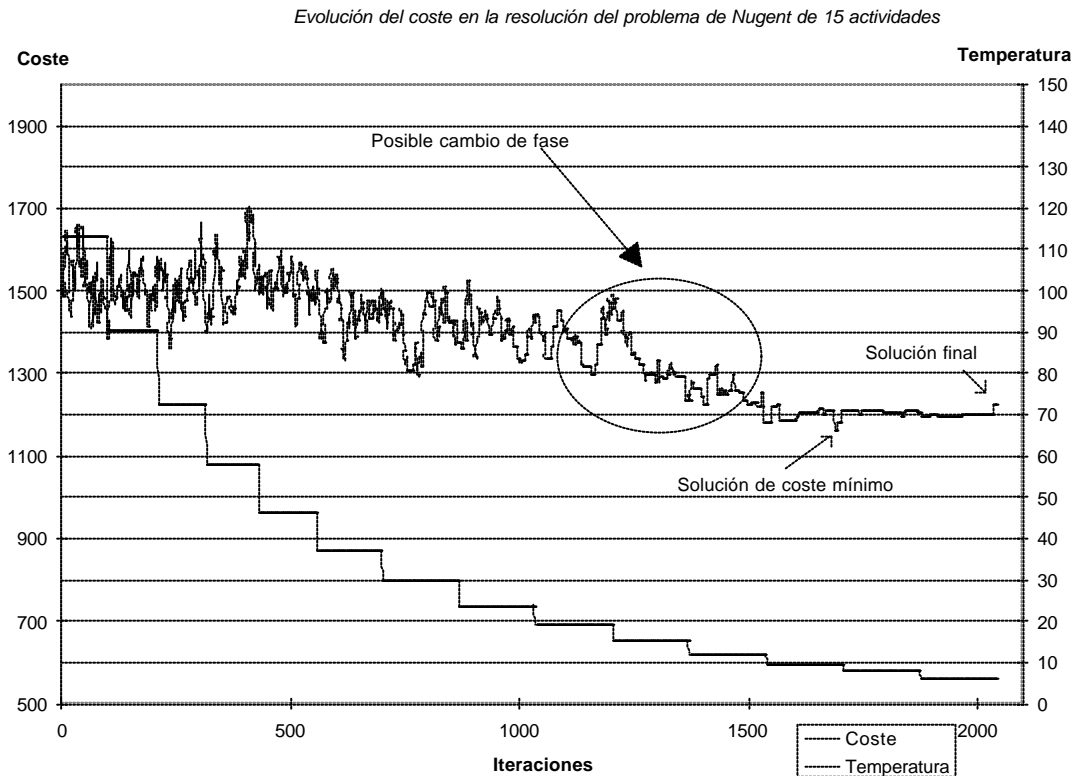


Figura 57. Evolución del coste y la temperatura en el algoritmo Simulated Annealing

- ***Elección del punto inicial de reconstrucción***

Hasta el momento, el punto inicial viene definido por la propia imagen de partida. Sin embargo, en algoritmos aleatorios como Simulated Annealing parece una contradicción el partir de puntos deterministas como es, en este caso, la imagen de partida. En REFER se ha implementado la posibilidad de partir de punto iniciales aleatorios. La razón de esta posibilidad no es otra que la de fijar el criterio aleatorio del algoritmo desde el inicio de su aplicación, permitiéndose así explorar distintos caminos en el proceso de optimización.

• **Ensayos y conclusiones.**

Como resultado de los ensayos realizados pueden extraerse las siguientes conclusiones:

1. Resulta evidente que un aumento del número de iteraciones a realizar en cada escalón de temperatura provoca un aumento del coste computacional, si bien la solución obtenida es más precisa. No obstante un excesivo aumento de este parámetro provoca un mayor número de rechazos y por tanto la condición de congelación del algoritmo puede llegar a verificarse para incrementos de variables grandes, dando como resultado costos elevados de la función objetivo y por consiguiente modelos poco próximos al modelo requerido. Deberá fijarse un número de iteraciones por escalón entre 2 y 4 (figura 58).

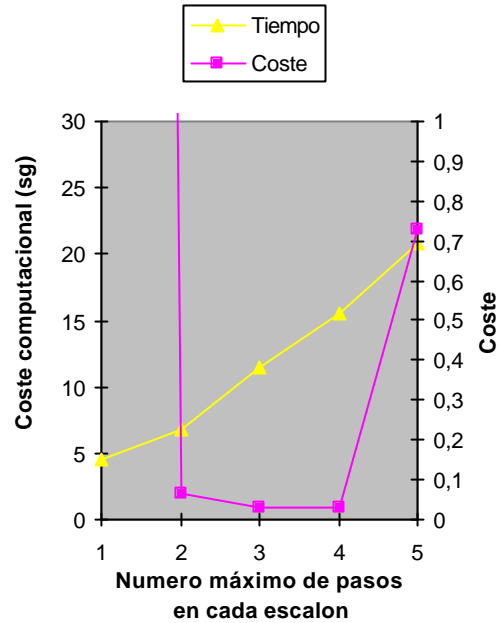


Figura 58. Efectos del número de pasos por escalón en el algoritmo SA

2. La capacidad de los algoritmos SA para escapar de mínimos locales radica en la aplicación del criterio metrópolis que como se ha comentado con anterioridad depende directamente de la temperatura. Como consecuencia, la definición de la temperatura inicial y su ley de enfriamiento son claves fundamentales para capacitar al algoritmo para escapar de mínimos locales. Se ha presentado el método implementado para la determinación de la temperatura inicial, pero hemos

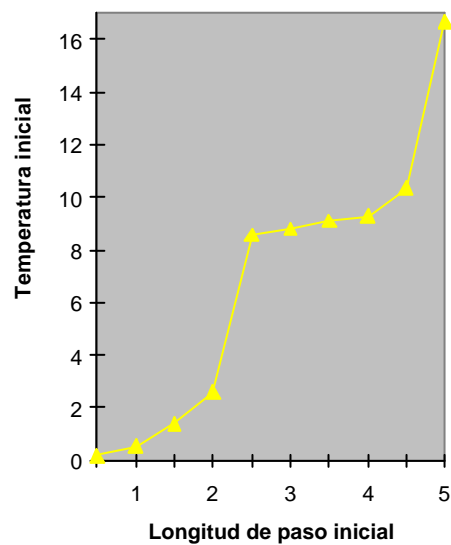


Figura 59. Efectos de la longitud del paso inicial en la temperatura inicial

de tener presente que dicho método se ve influenciado por el valor asignado para la longitud de paso inicial dado que, en el análisis del problema que nos ocupa en el que partimos de un mínimo local, elevados incrementos de variables iniciales favorecen el incremento de costos positivos en las transiciones iniciales, lo que favorece la obtención de una temperatura inicial elevada (figura 59).

3. Las temperaturas iniciales elevadas aumentan el número de transiciones aceptadas y por consiguiente el número de iteraciones a realizar por el algoritmo, aumentando el coste computacional. Pero además, dado que el número de iteraciones aceptadas es elevado, se verifica rápidamente el criterio de equilibrio y en consecuencia se suceden los saltos de escalón provocando un elevado número de transiciones en las fases finales, sin disminuciones importantes del coste. Por otro lado y dada la gran aceptación de desplazamientos permitidos, el algoritmo pierde capacidad para escapar de mínimos locales (ver figura 60).

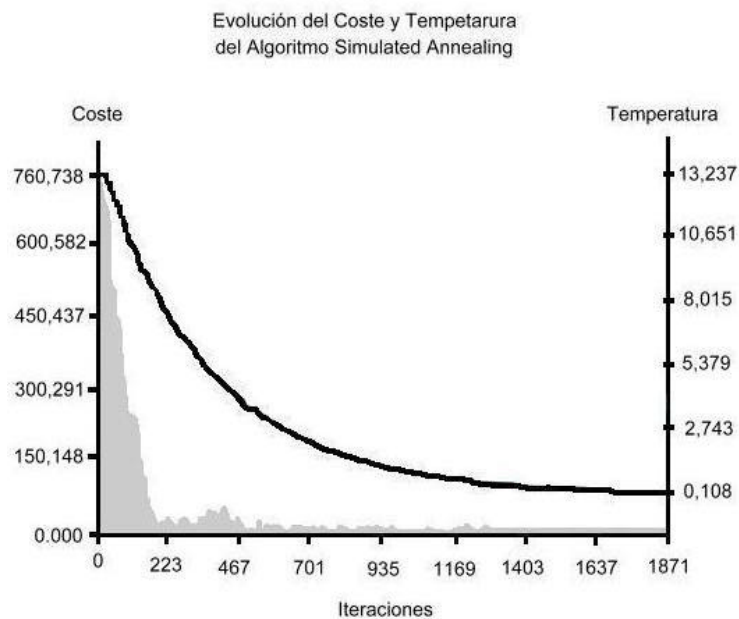


Figura 60. Representación del coste y la temperatura para un paso inicial igual al 4% min (RangoX, RangoY)

4. La longitud del paso inicial que determina el incremento inicial de las variables, al igual que ocurría para el algoritmo de Hill Climbing, es dependiente de las dimensiones de la imagen a reconstruir, por lo que no puede establecerse una lista

de escalones general sino que deberá normalizarse de forma automática en función de las dimensiones de la imagen.

5. La ley de enfriamiento debe asemejarse en lo posible a la evolución de la función coste. Cualquier desfase entre ambas curvas provocaría una disminución de las capacidades del algoritmo para escapar de mínimos locales. Por un lado, una ley de enfriamiento excesivamente drástica provoca temperaturas muy bajas en las últimas transiciones del algoritmo, por lo que el criterio metrópolis desciende en probabilidades y el comportamiento del algoritmo tiende hacia el de un algoritmo simple descendente. Por otro lado, una ley de enfriamiento excesivamente lenta mantiene temperaturas elevadas en los últimos pasos del algoritmo por lo que las transiciones son aceptadas en una gran mayoría produciendo grandes oscilaciones en el algoritmo de optimización.

6. El aumento de la longitud de paso inicial para la consecución de algoritmos rápidos, presenta los problemas de elevadas temperaturas iniciales puestos de manifiesto en 2 y como conclusión los problemas expuestos en 3. Dicho problema podría paliarse reduciendo el coeficiente de aceptación para la determinación de la temperatura inicial (ver figura 61), pero no se evitaría el problema de ejecución de excesivas iteraciones con efectos mínimos sobre la función coste.

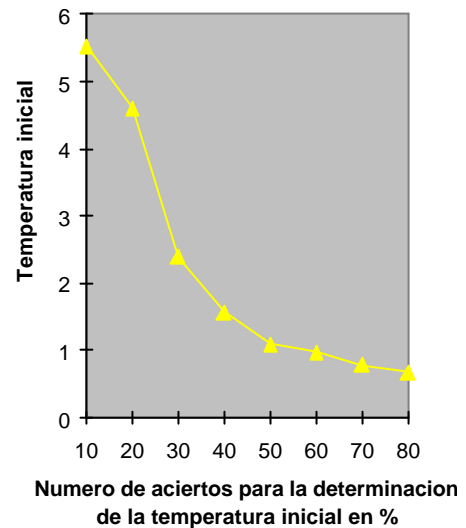


Figura 61. Efectos del coeficiente de aceptación en la determinación de la temperatura inicial

Por ello se aconseja una ley de enfriamiento más drástica. Fijando dicha ley de enfriamiento en la forma $T_{i+1} = 87\% T_i$ puede conseguirse un algoritmo rápido y fiable para un paso inicial del 4% min (RangoX, RangoY). La bondad de la propuesta se justifica porque, como se muestra en la figura 62, la curva de temperaturas se adapta al decremento del coste.

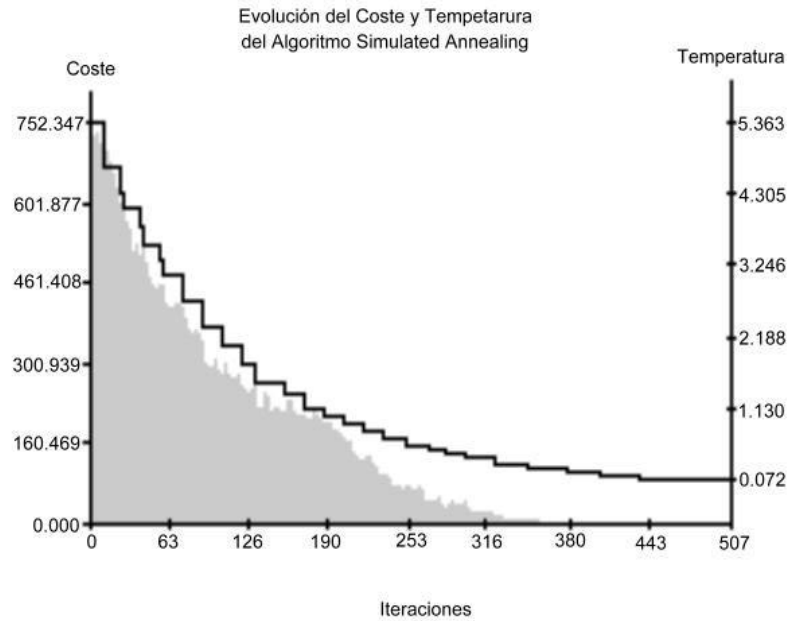


Figura 62. Representación del coste y la temperatura para un paso inicial igual al 4% min (RangoX, RangoY) y ley de enfriamiento del 87%

7. La reducción de incrementos de variables debe estar ligada a la ley de disminución de la temperatura inicial. Es decir grandes variaciones de los incrementos de variables pueden provocar incrementos muy pequeños para temperaturas elevadas, lo que, a su vez, provocaría gran número de aceptaciones y por consiguiente grandes oscilaciones del algoritmo de optimización. Dichas oscilaciones implican un gran coste computacional y una disminución de la capacidad del algoritmo para escapar de mínimos globales. Por otro parte, pequeñas variaciones de los incrementos de variables pueden provocar grandes incrementos de variables para temperaturas bajas. Lo que aumentaría el número de rechazos en el algoritmo y como consecuencia favorece la llegada prematura al criterio de congelación, finalizando el proceso en modelos muy alejados del mínimo global. Como consecuencia se aconseja reducciones del paso entorno al 98% del paso actual.
8. Resulta obvio que el decremento de paso final asegura un mayor refinado del modelo obtenido a costa de un mayor coste computacional.
9. La posibilidad de seleccionar puntos iniciales aleatorios, así como de ejecutar en reiteradas ocasiones el proceso de optimización, no asegura la obtención del mínimo global dada la complejidad de la función objetivo, y en cualquier caso, supone un alto coste computacional.

Según se ha mostrado hasta el momento, la optimización, por su analogía con la percepción visual humana, es el método más prometedor para reconstruir modelos psicológicamente válidos. Sin embargo, aunque sólo el óptimo global es válido para asegurar un modelo “psicológicamente verosímil”, los procesos de optimización no lo pueden garantizar. En otras palabras: aunque se consiga formular el problema de modo que el óptimo global coincida con el modelo psicológicamente plausible, los algoritmos de optimización no siempre podrán obtener dicho modelo de forma automática.

Uno de los problemas a los que se enfrenta el algoritmo de optimización es el punto solución inicial. Hasta la fecha, se han propuesto pocos métodos para determinar un buen punto solución inicial. Lo habitual es tomar la propia imagen (dibujo 2D). Esto supone un problema, dado que muchas regularidades se cumplen en dicha imagen; por lo que en algunos casos el algoritmo de optimización ni siquiera arranca por encontrarse en un mínimo local (*solución trivial*). Por consiguiente, se precisa algún método de *inflado previo*, que genere *modelos tentativos* que ayuden al algoritmo de optimización a escapar de dicho óptimo trivial.

Pero además, y dada la dificultad que los algoritmos de optimización tienen para escapar de mínimos locales en problemas que presentan complejidad del espacio de soluciones, se precisa que tales métodos de inflado previo conlleven la obtención de puntos solución que se aproximen bastante al óptimo global.

Es decir, se trata de elaborar una metodología para obtener un modelo 3D tentativo cuya geometría sea tan cercana como fuera posible a la solución final. A dicho proceso le hemos denominado *inflado previo*.

En este capítulo se presentan los pocos métodos conocidos hasta el momento, y se desarrollan nuevas aportaciones propias.

5.1 Estrategias de inflado conocidas.

Las estrategias conocidas de inflado previo en los procesos de optimización están basadas en *reglas heurísticas* y pueden agruparse en dos categorías:

- inflado iterativo
- generación directa de un modelo.

Las estrategias de inflado iterativo fueron las primeras empleadas. Aunque la intención de sus autores era obtener modelos finales, antes que modelos previos.

Marill [Mari91] presentó la primera propuesta de estrategia de inflado en la cual se definía la función objetivo con un único componente: el MSDA (Mínima desviación estándar de ángulos). El MSDA, aplicado sin que haya ninguna señal visual de la imagen que justifique que el modelo a reconstruir debe cumplir tal condición de *equiangularidad*, no es una regularidad real. Se trata pues, de una regla heurística basada en el hecho de que los poliedros convexos regulares cumplen esta condición. Por ello resulta fácil adivinar que tal estrategia sólo funciona bien para poliedros “bastante” regulares. En definitiva, el método de Marill es una estrategia de inflado previo iterativo. No es una verdadera optimización por regularidades, dado que no identifica señales visuales que justifiquen el empleo de la única regularidad que define.

Por otro lado, en la aproximación de Leclerc y Fischler [Lecl92] la función Objetivo se obtiene sumando tres tipos de regularidades, que son balanceadas con un parámetro variable λ de la siguiente manera:

$$F = F_a + \lambda F_1 + (1-\lambda) F_0 \quad (44)$$

donde:

λ : Es un parámetro que varía durante el proceso de optimización desde 1 a 0.

F_1 : (la parte de “inflado previo”) incluye un conjunto de condiciones que no son cumplidas en la solución inicial ($z=0$), pero no son regularidades reales y no tienen que cumplirse en el óptimo.

F_0 : (la parte “optimización”) contiene las regularidades reales que son verificadas en la solución inicial (y que, por tanto, dan lugar a la solución trivial).

F_a: incluye las regularidades que pueden siempre ser aplicadas, porque no son verificadas en la solución inicial.

La estrategia pretende resolver el problema de escapar del óptimo trivial *al mismo tiempo* que se optimiza para encontrar el óptimo global. Utiliza inicialmente aquellas regularidades que no se ven afectadas por el óptimo trivial, junto con otras regularidades que ayudan a escapar de dicho óptimo trivial. Después desactiva estas regularidades y activa el resto de las regularidades, para asegurar que se alcance el óptimo global. No obstante, presenta dos inconvenientes: el primero es que es difícil ajustar el parámetro λ para asegurar que siempre se escapa del óptimo trivial sin pérdida sustancial de eficiencia. El segundo inconveniente es que es complejo ajustar el funcionamiento del algoritmo de optimización para evitar las oscilaciones provocadas por el cambio de la función objetivo durante el propio proceso. Por tanto, la técnica impide el uso de los algoritmos más sensibles a las oscilaciones.

La primera estrategia de inflado previo directo fue propuesta por Lipson y Sphitalni [Lips96]. Son los primeros que proponen un proceso formal de *reconstrucción preliminar* para evitar mínimos locales y acelerar la convergencia. El objetivo es que el modelo previo se centre en la forma global ignorando los pequeños detalles. Intentado imitar el proceso de percepción humana buscan los rasgos generales de las dimensiones principales del cuerpo. En consecuencia, su propuesta está basada en el análisis de la distribución angular gráfica de líneas con el fin de obtener ángulos prevalentes y direcciones principales (“trends”). Cuando se obtienen tres ángulos principales, la regla perceptual ortogonal (“aquellos ángulos deben corresponder a tres direcciones ortogonales”) es aplicada para obtener un modelo tentativo. Tal decisión se justifica porque “es la tendencia más prevalente en los dibujos de ingeniería, y también es la más fácil de identificar”.

Según la descripción de Lipson y Sphitalni [Lips96], el proceso comienza identificando las direcciones principales. Luego se construye un modelo asignando una coordenada $z=0$ a un nodo arbitrario (lo que equivale a fijar la altura del modelo respecto al plano de la imagen, por lo que no supone pérdida de generalidad). El proceso avanza asignando coordenadas Z a los nodos conectados al ya conocido. Si el modelo tiene una distribución angular con exactamente tres direcciones principales, la

asignación de coordenadas puede propagarse a todos los nodos, dando lugar a un modelo previo bueno.

No obstante, no hemos podido encontrar una descripción detallada del modo en que el algoritmo calcula las coordenadas Z , ni como resuelve los casos singulares, ni ningún resultado de su funcionamiento que permita evaluar la bondad de los resultados que obtiene. Nuestra conclusión es que Lipson y Sphitalni debieron aplicar el método propuesto por Lamb y Bandopadhyay [Lamb90], aportando la mejora de la detección automática de las tres direcciones prevalentes.

El método de Lamb y Bandopadhyay [Lamb90] no es propiamente un inflado previo al proceso de optimización. Su intención era reconstruir modelos. No obstante, creemos que es el antecedente directo de la propuesta de Lipson y Sphitalni, y nosotros lo hemos utilizado como guía para una de nuestras estrategias.

5.2 Estrategias propias para generación de modelos tentativos.

Una conclusión que empieza a ser ampliamente aceptada es que no existe una única estrategia de reconstrucción que sea siempre válida. Por el contrario, la estrategia más adecuada depende de la tipología del objeto a reconstruir. En el trabajo de De Bonet [DeBo95] por ejemplo, se sugiere que los objetos sean reconstruidos según su “gramática”. En nuestro caso, hemos aplicado tal conclusión a la obtención del modelo tentativo: buscando reglas distintas para las diferentes tipologías.

En otras palabras, que, dada la gran diversidad de modelos a tratar, muchas estrategias podrán dar resultados más o menos viables siempre que se apliquen a ciertas tipologías de modelos. Por tanto, hay que determinar previamente la tipología del modelo, para posteriormente elegir la estrategia más apropiada.

Tres son las estrategia desarrolladas en esta tesis: el inflado axonométrico, el inflado por niveles y el inflado por plano oblicuo. Todas se exponen tras analizar la estrategia de determinación de tipologías.

5.2.1 Determinación de la tipología del modelo.

Nuestro trabajo en reconstrucción se centra de momento en los objetos poliédricos. Sin embargo, no hemos utilizado ninguna de las clasificaciones clásicas de

los poliedros porque no son útiles para el propósito de decidir qué criterio de inflado se debe utilizar para reconstruir cada tipo de poliedro.

Por ello, hemos elaborado una clasificación orientada únicamente al propósito de controlar automáticamente la selección del criterio de reconstrucción más apropiado. La clasificación está estrechamente vinculada con las estrategias para generar un modelo previo y pueden consultarse en [Cone99].

En nuestra clasificación se contemplan:

- Poliedros normalones: el nombre es una generalización al mundo de los poliedros, del concepto de *polígonos normalones*, definidos por Dori [Dori92] como aquellos que tienen la propiedad de que todos los ángulos entre dos aristas concurrentes son de 90° .
- Poliedros prismáticos. Aquellos que están limitados por una superficie prismática y dos planos, generalmente paralelos, que cortan a todas las aristas de la superficie prismática generando paralelogramos (por lo que tienen directriz poligonal y generatrices paralela a una dirección dada).
- Poliedros piramidales. Aquellos en los que todas sus caras menos una concurren en un punto (por lo que tienen directriz poligonal y generatrices que pasan por un vértice dado).

Una vez clasificados los tipos de poliedros que se pueden reconstruir con diferentes criterios y configuraciones de parámetros, se precisa alguna herramienta de determinación automática de tipologías. Porque es obvio que la reconstrucción no puede ser automática si el método de inflado previo tiene que ser elegido cada vez por el usuario, o es preciso introducir la tipología del modelo.

La dificultad radica en que dicha detección debe realizarse cuando el objeto aun no está reconstruido. Por lo tanto, la única fuente de información es la figura a partir de la cual se va a efectuar la reconstrucción. Entonces, las características de dicha figura deben servir para clasificar al poliedro que representa.

No hemos encontrado ningún estudio sobre detección automática de las propiedades de poliedros a partir de una proyección de los mismos, por lo que hemos abordado el problema desde el principio. Hemos buscado criterios heurísticos simples, basados en propiedades fáciles de detectar en la imagen de partida.

En la tabla 63 se resumen las características que hemos tomado como criterios para diferenciar los distintos modelos a partir de propiedades que puedan ser detectadas.

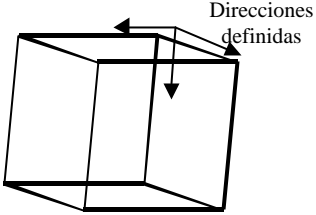
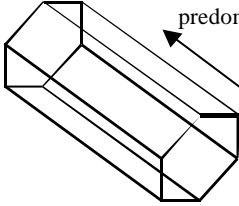
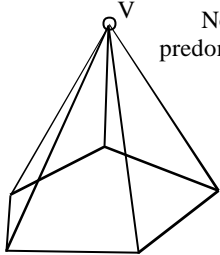
TIPOLOGÍAS DE POLIEDROS	EJEMPLOS
<p style="text-align: center;">Normalones</p> <p>Se considera que una figura representa un normalón cuando hay tres <i>direcciones principales</i>. Es decir, cuando los enlaces de la figura están orientados según tres y sólo tres direcciones.</p> <p>Se aceptan desviaciones de hasta 3° para considerar una recta paralela a una dirección dominante.</p>	 <p style="text-align: right;">Direcciones definidas</p>
<p style="text-align: center;">Prismáticos</p> <p>Se considera que una figura representa un prisma cuando entre los enlaces de la figura predomina una dirección. Es decir, cuando el número de enlaces paralelos a una dirección es suficiente respecto al número de nodos que componen la imagen.</p> <p>Un número de enlaces paralelos a la dirección predominante de un 50% del número de nodos de la imagen es suficiente para considerar un modelo prismático.</p>	 <p style="text-align: right;">Dirección predominante</p>
<p style="text-align: center;">Piramidales</p> <p>Se considera que una figura representa una pirámide cuando predomina uno de los nodos de la figura. Es decir, cuando el número de enlaces que concurren en un nodo es superior al de enlaces que concurren en los demás nodos.</p> <p>Un nodo se considera predominante cuando en él concurren un 15% o más enlaces que en el resto de nodos.</p>	 <p style="text-align: right;">V Nodo predominante</p>

Tabla 63. Criterios de clasificación de los modelos según las propiedades de la figura

Para implementar la detección de tipologías basada en los criterios de la tabla 63 se han precisado ciertos algoritmos que calculen los datos necesarios. En concreto se deben determinar las direcciones dominantes y los nodos dominantes. Al implementar la detección de tipologías, también se ha llegado a la conclusión de que era necesario establecer unos parámetros de tolerancia que permitiesen clasificar las imágenes. Dichos parámetros, cuyos valores se reflejan en la tabla 63, han sido fijados experimentalmente a partir del conjunto de modelos ensayados. Y el objetivo de los mismos es establecer una clasificación acorde a la interpretación de las tipologías definidas.

5.2.2 Inflado axonometrico.

El antecedente más claro al inflado previo axonométrico podemos encontrarlo en el trabajo de Lamb y Bandopadhyay [Lamb90]. Su trabajo precisaba de una entrada formada por una imagen en la que debían venir representados un conjunto de ejes (eje x, eje y, eje z) que eran considerados como ejes principales. Cada uno de los pares de ejes definían un plano coordenado (X, Y, Z). Cualquier plano paralelo a un plano coordenado se considera principal. Cualquier otro plano, que no puede ser identificado como plano principal, se denomina plano oblicuo.

Partiendo del etiquetado de Waltz, se identificaban las caras definidas en la imagen y se clasificaban como planos principales o planos oblicuos. La asignación de coordenadas para cada uno de los nodos de la imagen se realizaba siguiendo los siguientes pasos:

1. Selección de la “mejor” cara para asignar las coordenadas. El término de mejor cara hace referencia a la orientación de la misma y permite establecer un orden de evaluación de las caras definidas en la imagen. Así, las caras que son planos principales son evaluadas en primer lugar y las caras que son planos oblicuos son evaluadas en último lugar. De todas las caras que son planos principales se selecciona aquella de mayor longitud paralela a un eje principal. A modo de ejemplo, en la figura 64, el plano Y es evaluado en primer lugar porque representa la mayor longitud paralela a un eje principal (eje x).
2. Selección del nodo de referencia. Todas las coordenadas de los vértices de una cara del modelo son obtenidas por referencia a un único vértice denominado vértice de referencia. Si la cara evaluada tiene ya un vértice con coordenadas definidas, dicho vértice es tomado como vértice de referencia. Si la cara no posee ningún vértice al que le hayan sido asignadas las coordenadas (es la primera cara en evaluarse), entonces se toma como vértice de referencia aquel vértice que se encuentre en la posición más baja de la imagen (vértice 1 en la figura 64). En el caso de tratarse de la primera cara evaluada, al vértice de referencia se le asignan las coordenadas (0, 0, 0).
3. Asignación de coordenadas a todas los vértices de una cara. Las coordenadas de cada uno de los vértices que definen la cara son evaluadas a partir de las coordenadas del vértice de referencia, definiendo la ecuación del plano que

contiene a cada cara. La ecuación del plano es definida atendiendo a las siguientes situaciones:

- Si la cara evaluada es paralela a uno de los planos principales, la coordenada correspondiente al eje principal perpendicular a la cara de todos los vértices, se iguala a dicha coordenada en el vértice de referencia. Las otras dos coordenadas se obtienen directamente midiendo las longitudes de los enlaces según los ejes principales que definen el plano principal paralelo a la cara evaluada. En el ejemplo de la figura 64, los vértices 2, 3 y 4 tienen igual coordenada “y” que el vértice de referencia (vértice 1). Las coordenadas “x” y “z” son asignadas proporcionalmente a las longitudes de los enlaces representados en la imagen.

Si la cara esta formada por algún nodo no accesible a partir de enlaces paralelos a los ejes principales, se obtiene la distancia de dicho vértice al vértice de referencia y se descompone según vectores paralelos a los ejes principales (este es el caso de los vértices 8, 9, 10 y 11 de la figura 64).

- Si la cara es paralela a un plano oblicuo del que se conocen al menos las coordenadas de tres de sus vértices, se establece la ecuación del plano definido por estos tres vértices. Las coordenadas del resto de los vértices que definen la cara son determinadas de idéntica manera a como se expuso en el punto anterior, pero considerando ahora la ecuación determinada para el plano oblicuo.

En la figura 64, la coordenada del vértice 7, sería determinada a partir de las coordenadas de los vértices 2, 3 y 6, evaluadas anteriormente por pertenecer a caras paralelas a planos principales. Es decir, el orden de evaluación previsto para la figura 64 sería el siguiente:

1. En primer lugar se evalúan las coordenadas de los vértices 2, 3 y 4, a partir del vértice de referencia 1, por encontrarse en un plano principal Y.
2. Posteriormente se evalúan las coordenadas de los vértices 5 y 6 por pertenecer a otro plano principal (plano X).
3. Por último se evalúan el vértice 7 que pertenece a un plano oblicuo, a partir de la ecuación del plano determinada por las coordenadas de los vértices 2, 3 y 6.

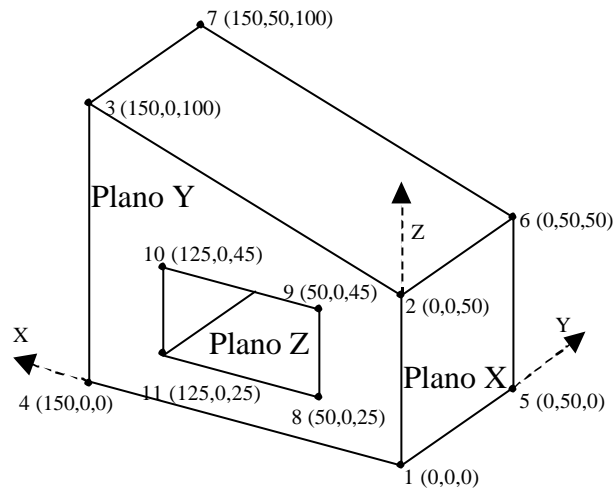


Figura 64. Asignación de coordenadas conforme al algoritmo de Lamb y Bandopadhyay

- Si la cara es un plano oblicuo, pero no existe suficiente información para determinar la ecuación del plano, se intenta establecer relaciones de simetría en la propia cara (simetría facial oblicua). En la figura 65 se muestra un ejemplo donde puede adivinarse la simetría de la cara definida por los vértices 1, 2, 3 y 4. La región definida por dichos vértices es un paralelogramo y se supone que dicho paralelogramo es interpretado por el observador como un rectángulo. Esta asunción es adoptada y las coordenadas de los vértices pueden ser fácilmente designadas.

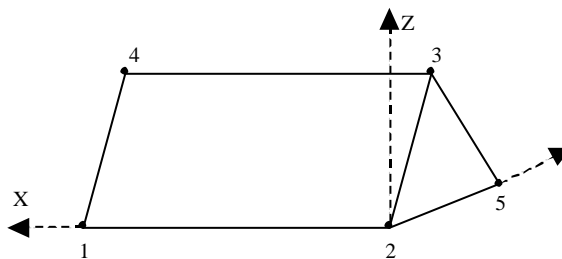


Figura 65. Plano oblicuo en un modelo simétrico.

- Si la cara es paralela a un plano oblicuo del que no existe suficiente información para establecer la ecuación del plano, y las relaciones de simetría no pueden ser

definidas, entonces se precisa de interactividad con el usuario para que responda a las cuestiones necesarias para establecer las coordenadas precisas para definir el plano.

En la figura 66 se muestra un ejemplo donde el usuario debe introducir las coordenadas para definir la posición tridimensional del vértice 1.

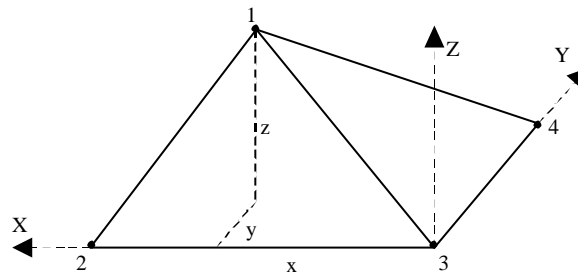


Figura 66. Plano oblicuo en un modelo asimétrico

4. Proceso iterativo. Los pasos descritos en los puntos 1, 2 y 3 se repiten de manera iterativa hasta que se le asignan coordenadas a todos nodos de la imagen.

El método que nosotros proponemos, actúa de manera similar al expuesto, en tanto que calcula de manera iterativa las coordenadas de cada uno de los vértices a partir de un vértice de referencia al que le son asignadas las coordenadas (0, 0, 0).

Sin embargo, el método de reconstrucción propuesto por Lamb y Bandopadhyay precisaba que el usuario determinase las proyecciones de los ejes de proyección impidiendo la automatización del proceso. También se requiere interactuar con el usuario cuando aparecen vértices de difícil determinación (vértices pertenecientes a planos oblicuos en los que no se conocen las coordenadas tridimensionales de ninguno de sus vértices). Además el dibujo sólo debía contener aristas visibles. La justificación era la ambigüedad que las aristas ocultas ocasionan en la interpretación de la imagen; aunque también aseguran ser capaces de obtener la parte oculta del modelo a partir de su geometría frontal mediante el método de etiquetado. Pero entre todas sus limitaciones, quizás la más importante es la necesidad de detectar caras para aplicar el método de reconstrucción propuesto.

El método que proponemos presenta diferencias sustanciales con el de Lamb y Bandophaday. En primer lugar, incorpora la detección automática de direcciones principales propuesta por Lipson y Shpitalni. En segundo lugar, está basado en el modelo alámbrico, por lo que no necesita detección de caras. Por último, se pueden determinar las coordenadas de más vértices; dado que a partir de la coordenada arbitraria de un vértice inicial se pueden determinar todos vértices conectados a él por enlaces paralelos a las direcciones principales, aunque pertenezcan a caras oblicuas.

El método se basa en aplicar la relación axonométrica que existe entre los ángulos y las longitudes de tres aristas ortogonales que concurren en un vértice y sus correspondientes proyecciones ortogonales. En concreto, la relación se basa en el cálculo del ángulo definido por una arista y su proyección en la imagen dada, que, como se justifica detalladamente en el anexo 3, viene dado por la expresión:

$$\phi = \arcsen \sqrt{\cotg \alpha' \cotg \beta'} \quad (45)$$

siendo α' y β' los ángulos que la proyección ortogonal de la arista dada (CD) forma con las proyecciones ortogonales de las otras dos aristas que convergen en su mismo vértice formando un triedro trirectángulo, (ver figura 67).

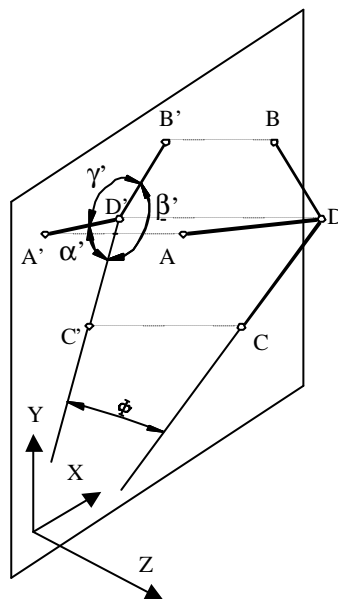


Figura 67. Dependencia de los ángulos en una proyección

Determinando los ángulos que cada una de las tres aristas (CD, AD y BD) forman con sus correspondientes proyecciones, nuestro método calcula las coordenadas “Z” relativas de todos los vértices conectados al vértice conocido.

$$ABS(Z_C - Z_D) = C'D' \text{ atan } \phi \quad (46)$$

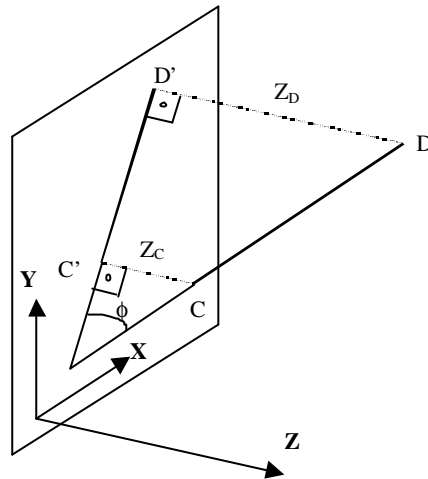


Figura 68. Relación entre el ángulo y las coordenadas “Z”

Sin embargo, la ecuación dada en (45) sólo es válida para proyecciones ortogonales de triedros trirectángulos. Por tanto, el método es válido siempre que en el vértice de partida concurren tres y sólo tres aristas, y que éstas formen un triedro trirectángulo. Por ello el método resulta válido para los poliedros que anteriormente hemos definido como normalones.

El método se puede propagar de forma sencilla tomando cualquiera de los tres vértices ya conocidos (A, B, C) como vértice central de un nuevo conjunto de tres aristas concurrentes y ortogonales. De esta manera, recorriendo los nodos de la imagen ordenadamente, pueden ser obtenidas las distintas coordenadas Z. Así, para la imagen en la figura 69, y tomada una coordenada arbitraria “Z” en el vértice V₀, podrán evaluarse los distintos vértices según se refleja:

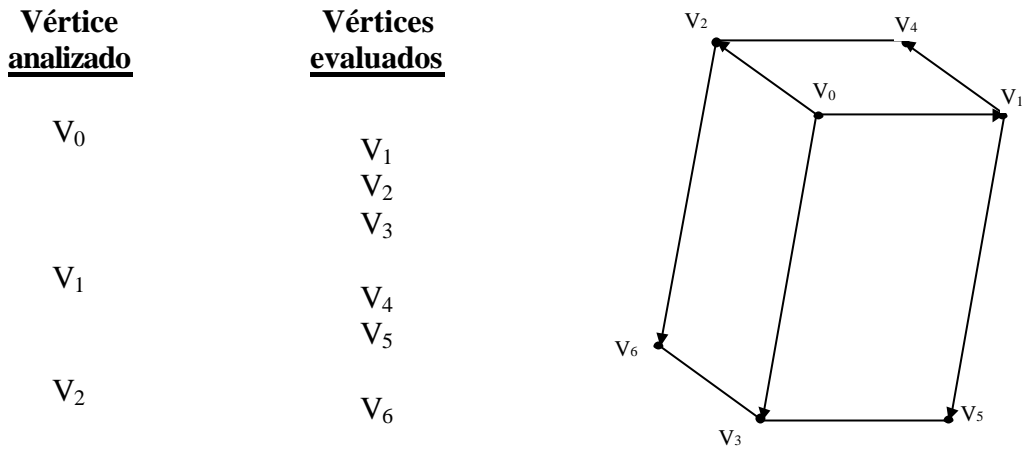


Figura 69. Determinación de coordenadas “Z” de una imagen mediante inflado axonométrico

La coordenada “Z” arbitraria inicialmente asignada al vértice V₀ no influye en el resultado del inflado obtenido ni en la topología del modelo, en tanto que dicho valor simplemente reflejaría un desplazamiento perpendicular al plano sobre el que es proyectada (figura 70).

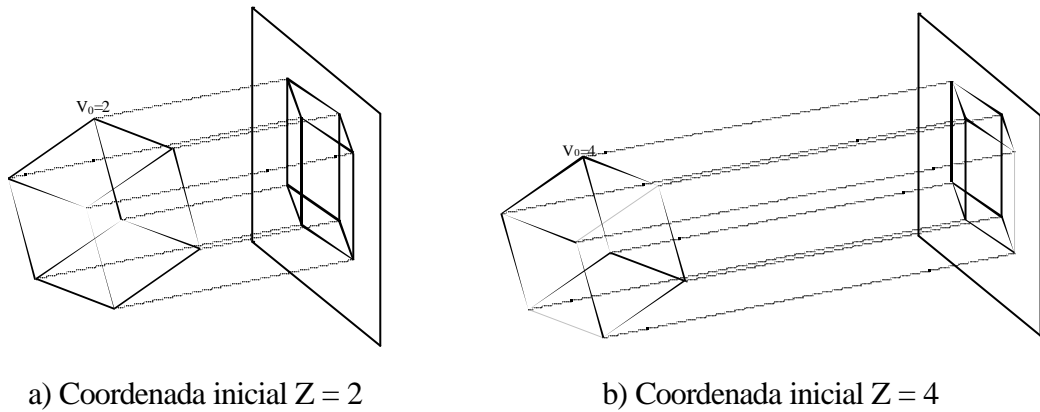


Figura 70. Efectos de la asignación de coordenada inicial arbitraria en el inflado axonométrico

En REFER, la coordenada “Z” inicial de un vértice aleatorio es puesto a cero para el inflado previo axonométrico. Pero, una vez realizado el inflado, se produce un desplazamiento perpendicular al plano de proyección, hasta que la menor de las

coordenadas “Z” obtenidas se iguala a cero (el vértice queda contenido en el plano de la imagen). Se consigue así que el plano de la imagen no corte al modelo resultante.

No obstante, el problema no queda resuelto, puesto que, una vez determinado el ángulo ϕ correspondiente, nos resta por seleccionar la orientación correcta de la arista; o, lo que es lo mismo, el valor final de la coordenada “Z” del vértice buscado.

De la figura 71 pueden deducirse dos posibles coordenadas “Z” en el análisis del vértice “C” en función de la orientación tomada para el ángulo axonométrico, y que vienen dadas por las expresiones (47) y (48).

$$Z_{C1} = Z_D - C'D' \tan(\phi) \quad (47)$$

$$Z_{C2} = Z_D + C'D' \tan(\phi) \quad (48)$$

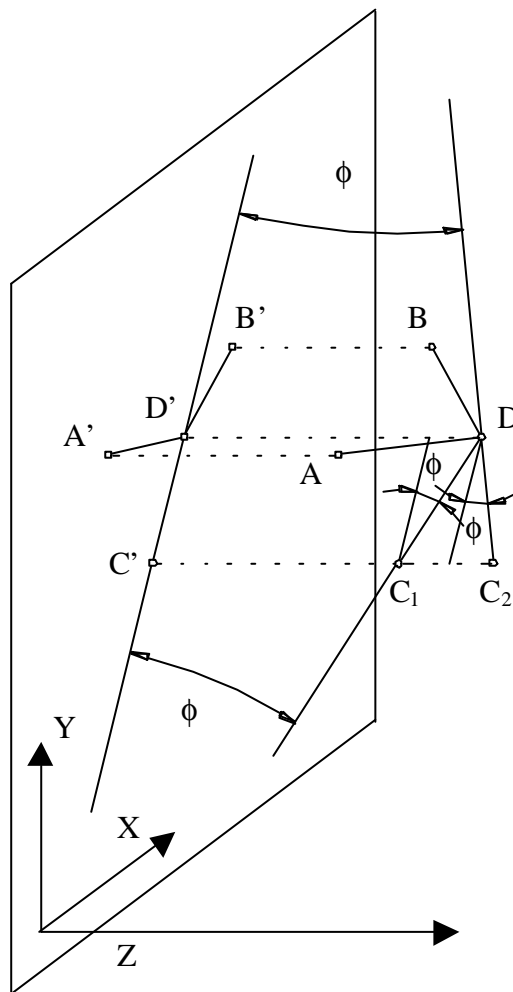


Figura 71. Selección de la orientación del ángulo axonométrico

Parece imposible definir, con carácter general, la orientación adecuada para la obtención del modelo válido a partir de la imagen bidimensional. No obstante, los normalones cumplen una condición que nos ayuda a descartar la orientación no deseable. En efecto, la característica particular de estos poliedros, por la que todas sus esquinas resultan ser ortogonales, puede utilizarse para seleccionar la orientación del ángulo apropiada. Basándonos en dicha propiedad, de las coordenadas Z_{C1} y Z_{C2} obtenidas mediante las ecuaciones (47) y (48), seleccionaremos como válida aquella que consiga el ángulo tridimensional más ortogonal posible con respecto a una arista de referencia inicial que incida en el mismo vértice. En la configuración de la figura 71, supuesta como arista de referencia la arista AD, queda claro que la coordenada Z_{e1} será la adecuada para definir un modelo normalón válido.

Por último, debemos definir una *arista de referencia*, que es aquella arista cuyos dos vértices son los primeros de cada triedro a los que se asigna coordenadas “Z”.

Evidentemente la arista de referencia inicial será una de las tres que concurren en el vértice al que se le asigna una coordenada Z arbitraria en el inicio del proceso. La elección de una de las tres aristas que concurren en él, así como la orientación del ángulo tridimensional para la evaluación de la coordenada Z del segundo de los vértices realmente puede dar lugar a dos modelos distintos. En otras palabras, la elección de esta arista de referencia tiene efectos en la consecución de un modelo o su respectiva inversión de Necker como se pone de manifiesto en la figura 72.

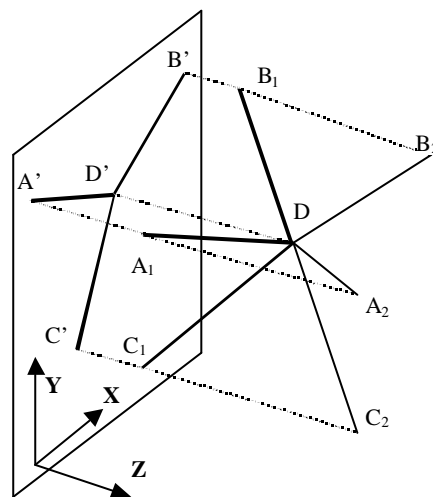


Figura 72. Inversión de Necker

Cuando el vértice “D” de la figura 72 ha sido el seleccionado para fijar su coordenada “Z” aleatoria, y la coordenada del vértice “C” ha sido obtenida mediante la aplicación de un criterio cualquiera (y se ha seleccionado C_1 ó C_2), la arista CD (C_1D ó C_2D) se convierte en arista de referencia para determinar la orientación de las aristas AD y BD, permitiendo obtener las coordenadas “Z” de los vértices A y B. Posteriormente, las aristas AD y BD serán consideradas como aristas de referencia para determinar la orientación adecuada de todas aquellas otras aristas que converjan respectivamente en los vértices A y B.

En REFER se han establecido los siguientes criterios para determinar la orientación de la primera arista de referencia (ver figura 73):

1. Si $Y_{D'} < Y_{C'}$ entonces: $Z_C = Z_D - C'D' \text{ tang}(\phi)$
2. Si $Y_{D'} > Y_{C'}$ entonces: $Z_C = Z_D + C'D' \text{ tang}(\phi)$
3. Si $Y_{D'} = Y_{C'}$ (proyección horizontal), entonces:
 - 3.1 Si $X_{D'} < X_{C'}$ entonces $Z_C = Z_D - C'D' \text{ tang}(\phi)$
 - 3.2 Si $X_{D'} > X_{C'}$ entonces $Z_C = Z_D + C'D' \text{ tang}(\phi)$

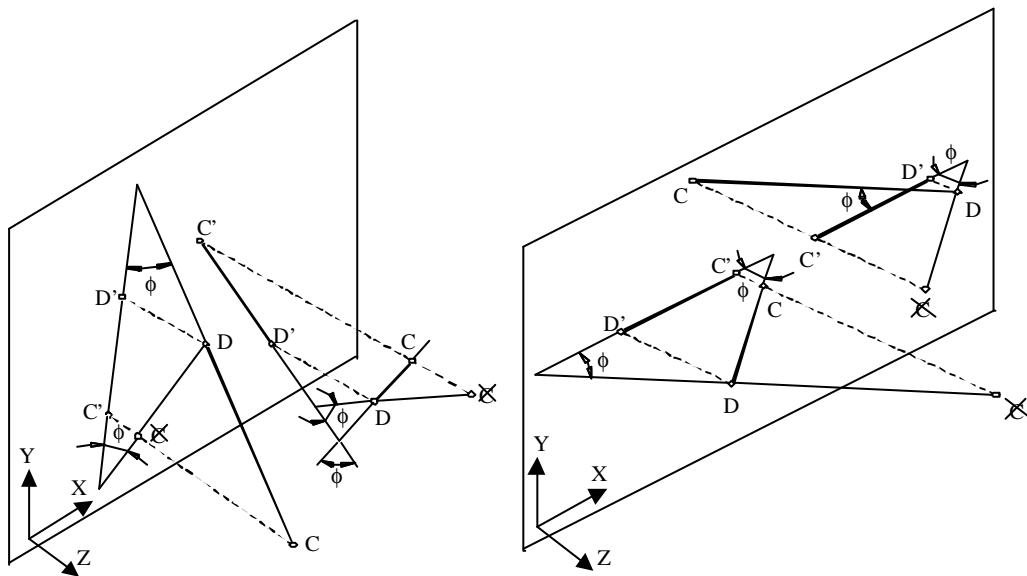


Figura 73. Criterios para la selección en la orientación del ángulo axonometrico

En la figura 74 pueden verse los resultados obtenidos en REFER mediante inflado axonómico sobre un modelo de tipología normalón. Los resultados obtenidos

dejan patente la posibilidad de reconstrucción automática y directa de tales modelos sin precisar de procesos de optimización.

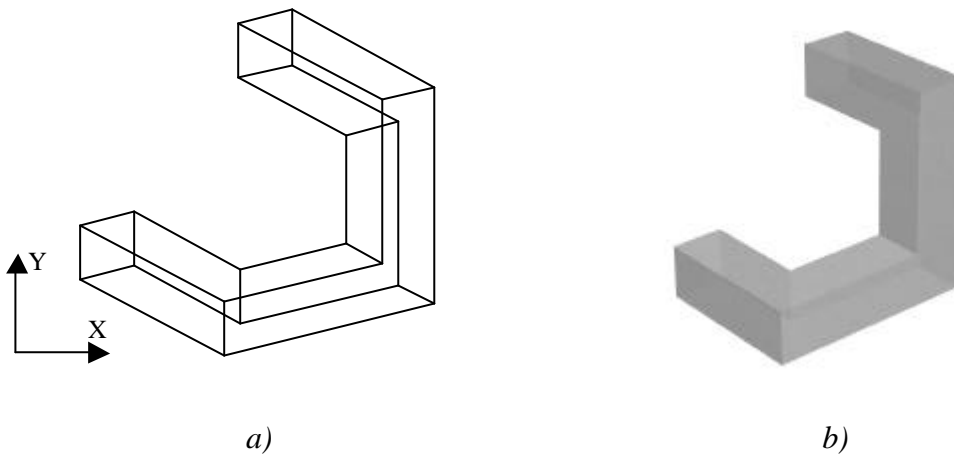


Figura 74. Reconstrucción obtenida por inflado axonométrico
a) Imagen original b) Objeto reconstruido

- **Ampliaciones del método a otras tipologías.**

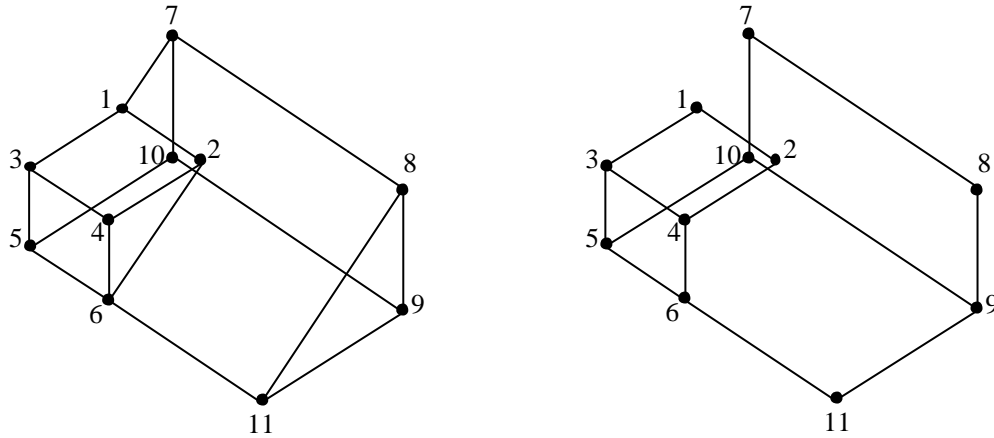
Como ya se ha comentado anteriormente el método propuesto sólo obtiene modelos finales para aquellas imágenes que representen proyecciones ortogonales de objetos normalones.

Su validez para otras tipologías se limita inicialmente a la obtención de modelos tentativos más o menos próximos al modelo final.

No obstante, aprovechando la ventaja de que el método no utiliza información de caras, es posible aplicarlo a ciertos poliedros que a pesar de no ser normalones permitan la evaluación de todos los vértices, utilizando para ello exclusivamente aristas que resulten paralelas a tres direcciones principales. Es decir, que se trata de modelos que cumplen la condición de que la eliminación de todas las aristas no paralelas a tres direcciones principales no supone la pérdida de vértices. Hay que destacar que, en este contexto, se consideran direcciones principales aquellas que se repiten con mayor frecuencia en una imagen.

A los modelos que cumplen la condición descrita arriba los hemos denominado de tipología *cuasi-normalón*. En la figura 75 se muestra un claro ejemplo de este tipo de imágenes, donde como puede observarse, la eliminación de los enlaces 1-7, 2-6 y 8-11 no supone pérdida de información sobre los nodos que componen la imagen. Como

consecuencia de esta primera transformación de un objeto cuasi-normalón, la imagen resultante queda convertida en normalón, lo que permitiría la reconstrucción del modelo mediante inflado axonométrico.



a) Imagen de tipología cuasi-normalon

b) Imagen de tipología normalon

Figura 75. Primera transformación a normalón evaluable

No obstante debemos tener presente que en los vértices que componen el camino recorrido para la evaluación del modelo, se precisa de la concurrencia de tres aristas para permitir la evaluación del ángulo que cada arista forma con su proyección de acuerdo con la ecuación (45). Sólo el último vértice de cada camino puede quedar exento de tal condición (puesto que a él se llega, pero de él no se sale). De ahí que el análisis realizado en el modelo de la figura 75b, suponiendo que se parte del nodo 3 como vértice de referencia inicial, no pueda ser otro que el que se refleja en la figura 76.

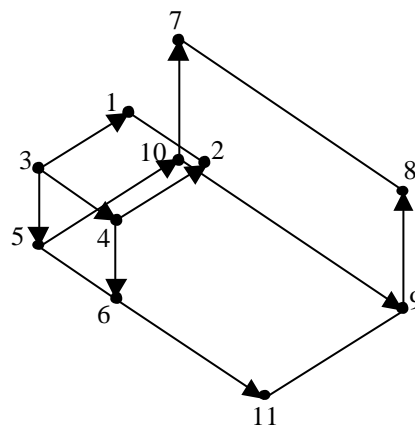
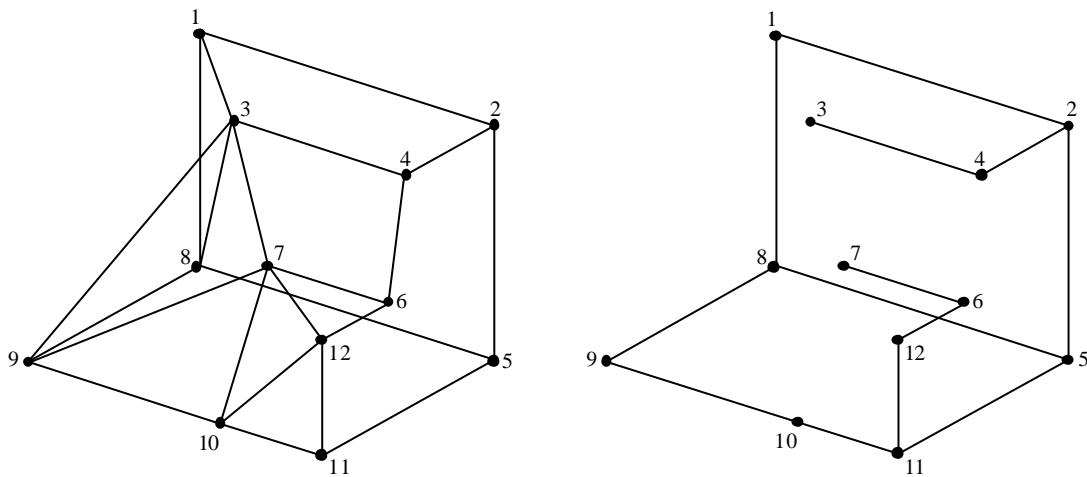


Figura 76. Recorrido en análisis de inflado axonométrico

Es decir, el vértice de partida debe ser algún vértice en el que concurran tres aristas. En el ejemplo se ha tomado el vértice 3. A partir de dicho vértice pueden determinarse las coordenadas de los vértices 1, 4 y 5. Sin embargo, al continuar el proceso, el vértice 1 no permite evaluación alguna en tanto que en él sólo concurren dos aristas, por lo que el camino a través de dicho vértice queda cerrado.

La evaluación continua por el vértice 4 que permite obtener coordenadas en 2 y 6. El vértice 5 permite obtener coordenadas en 10, el vértice 6 coordenadas en 11, y el vértice 10 coordenadas en 2 y 9. La evaluación del vértice 7 nuevamente presenta idénticos problemas que el vértice 1 por lo que la evaluación del vértice 8 se realiza a través del vértice 9, finalizando la definición del modelo.

Por consiguiente puede ocurrir que, a pesar de conseguir una tipología normalón mediante la transformación de un objeto cuasi-normalon, la imagen no pueda ser modelizada si ciertos vértices no son accesibles a través de algún camino válido. El ejemplo de la figura 77 es uno de esos casos, ya que al vértice 3 solamente puede llegarse a partir del vértice 4 en el cual sólo concurren dos aristas, y al vértice 7 y 6 solamente puede llegarse a partir del vértice 12 al cual sólo llegan también dos aristas.



a) Imagen de tipología cuasi-normalon

b) Imagen de tipología normalon

Figura 77. Primera transformación a normalón no evaluable

Para estos casos se utiliza una segunda transformación, añadiendo aristas ficticias de longitud unidad y orientadas de acuerdo con aquellas direcciones principales no presentes en el vértice.

En la figura 78 pueden verse los resultados de esta segunda transformación.

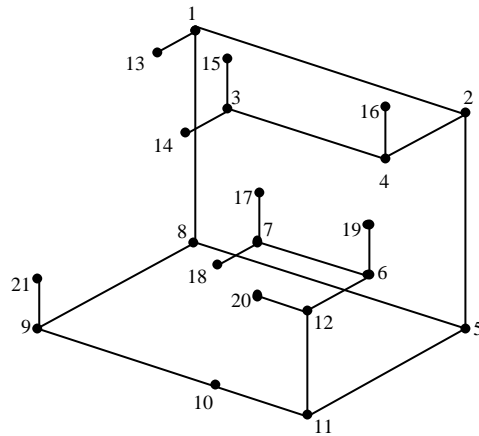


Figura 78. Segunda transformación a normalón

La imagen dada en la figura 78 sí corresponde ahora con una imagen de tipología normalón que permite la evaluación de todos sus nodos mediante inflado axonométrico.

Otra tipología de objetos no normalón que se pueden reconstruir, pero requieren un tratamiento previo, son aquellos que tienen vértices de valencia superior a tres. Vértices en los que concurren más de tres aristas.

Quizás el tema se puede zanjar diciendo que, de momento, nos hemos conformado con obtener uno de los posibles modelos tentativos eligiendo, al azar, tres de las aristas que concurren en dichos vértices: las tres primeras que aparecen en la base de datos de la imagen.

En particular, se debe considerar el caso de aquellos vértices en los que concurren aristas colineales, dado que se presenta un inconveniente en la determinación de la orientación del ángulo tridimensional.

En la figura 79 se muestra un ejemplo de aristas colineales para ilustrar mejor este problema. Supongamos conocidas las coordenadas Z de los vértices 1, 2 y 3 del modelo y que se desea determinar la coordenada Z del vértice 4. Siguiendo con el procedimiento expuesto, se determina el valor del ángulo ϕ , a partir de los ángulos que forman el enlace 2'-4' con los enlaces 1'-2' y 2'-3'.

Las posibles coordenadas del vértice 4, vienen determinadas por las expresiones (47) y (48) que representan los vértices 4_1 y 4_2 de la figura 79. Sin embargo, como puede observarse, tanto la arista $2-4_1$ como la arista $2-4_2$ resulta ortogonal a las aristas $1-2$ y $2-3$, lo que imposibilita seguir el criterio de perpendicularidad para seleccionar la configuración más apropiada.

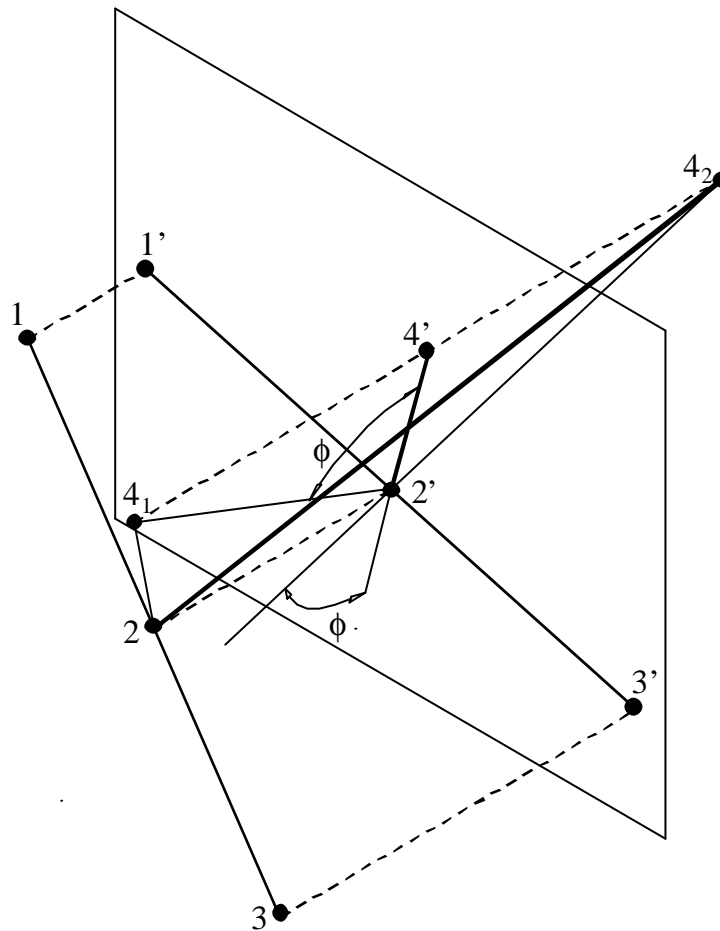
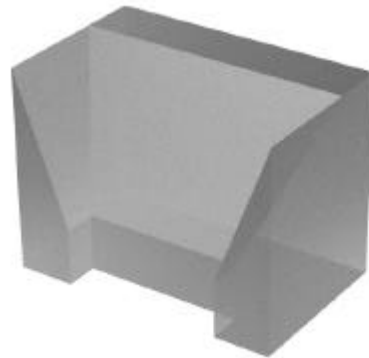
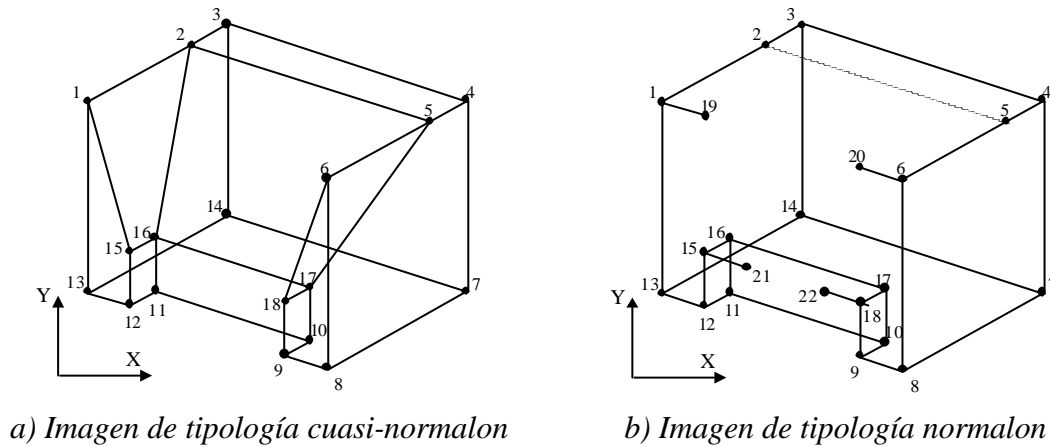


Figura 79. Indeterminación del inflado axonométrico ante aristas colineales

Por tanto, en la aplicación de esta estrategia deberá asegurarse buscar caminos de evaluación que no determinen vértices a partir de aristas colineales dado que su evaluación podría provocar errores que se arrastrarían al resto de los vértices en el inflado previo obteniendo modelos inverosímiles.

Un ejemplo de lo expuesto se muestra en la figura 80 donde se representa la imagen de partida (figura 80a) y su correspondiente transformación a cuasi-normalón (figura 80b). Como se observa en la evaluación de esta imagen las aristas $1-2$ y $2-3$ y las

aristas 4-5 y 5-6 resultan ser colineales entre sí, por lo que no deberán ser consideradas como aristas de referencia para evaluar cualquier otra aristas que concurra en los vértices 2 y 5. En otras palabras, el vértice 5 deberá ser evaluado a partir del vértice 4 o del vértice 6, y el vértice 2 deberá ser evaluado de manera independiente a partir del vértice 1 o del vértice 3. Nunca la arista 2-5 debe ser considerada arista perteneciente al camino de evaluación.



c) *Reconstrucción del modelo en REFER*

Figura 80. Inflado axonométrico en imagen con aristas colineales

El algoritmo hasta aquí presentado resulta válido para todo tipo de normalones y casi todo tipo de cuasi-normalones. No obstante, puntos de vista accidentales pueden ocasionar interpretaciones erróneas de no-normalones como falsos normalones o cuasi-normalones. Tal es el caso en el que dos aristas proyectadas paralelamente en la imagen no corresponden a dos aristas paralelas tridimensionalmente, sino a dos aristas que se cruzan en el espacio (cuestión ya debatida en el estudio de la regularidad de paralelismo). La figura 81 muestra un ejemplo donde la arista 5-6, resulta paralela a las

aristas 1-4, 2-3 y 7-8, si bien realmente debe interpretarse como una arista que se cruza con estas.

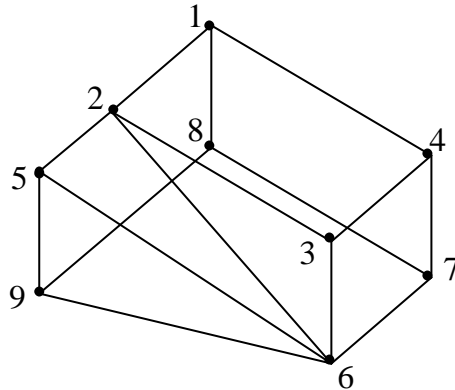


Figura 81. Inconsistencias en el paralelismo de aristas

La imagen dada en la figura 81 puede ser transformada a imagen de un modelo normalón, quedando de la manera que se muestra en la figura 82. En dicha figura se observa que la aristas 5-6 no se elimina por ser paralela a una de las direcciones principales.

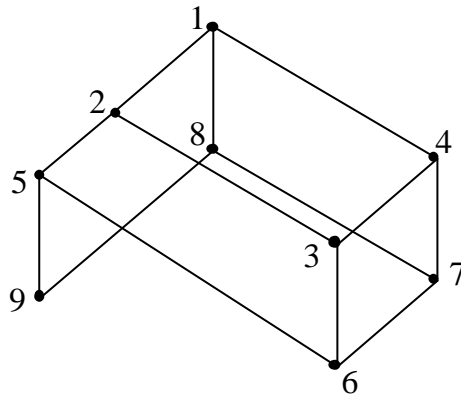


Figura 82. Primera transformación a normalón de una imagen con inconsistencias de paralelismo

Aplicando el inflado axonométrico y teniendo presente que éste se basa en la ortogonalidad de sus aristas, si la arista 5-6 es evaluada a partir de vértice 6, considerando como arista de referencia la arista 3-6, y dado que la arista 5-6 es paralela en la imagen a la arista 2-3, la orientación dada sería tal que en el vértice 6 todos los

ángulos tridimensionales formasen 90 grados , es decir, los vértices 5, 6 y 7 resultarían coplanarios (figura 83).

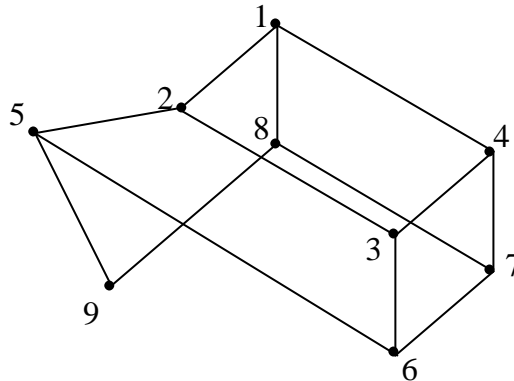


Figura 83. Errónea reconstrucción de un modelo con inconsistencias de paralelismo

Se precisa por tanto introducir un nuevo factor de comprobación. Para ello se ha optado por realizar una comprobación de la colinealidad de aristas prioritaria respecto del paralelismo de arista. De este modo, si una arista de la imagen es colineal con cualquier otra arista con al que comparte algún vértice, ambas deberán ser colineales tridimensionalmente. En el ejemplo que nos ocupa, se fuerza de este modo el análisis del vértice 5 a partir de la arista de referencia 1-2. Lo que conlleva seleccionar una orientación para el ángulo ϕ tal que las aristas 1-2 y 2-5 resulten colineales en el espacio. Mediante la aplicación de este nuevo criterio se consigue la reconstrucción que se muestra en la figura 84.

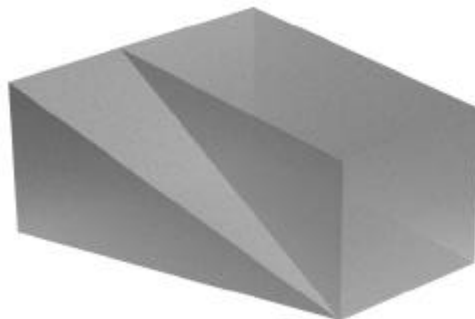
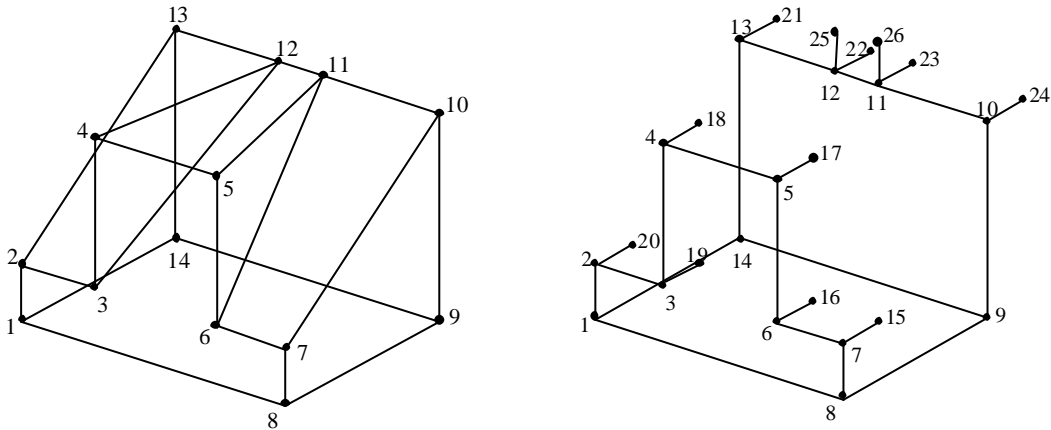


Figura 84. Reconstrucción de un modelo con inconsistencias de paralelismo

No obstante, para prever posibles malfuncionamientos debidos a una mala selección de caminos, se hace una comprobación a posteriori del modelo final. En caso de que el modelo no sea normalón se invalidan los caminos afectados por los vértices no trirrectangulares y se intenta una reconstrucción por caminos alternativos. Caso de que tampoco se consiga un modelo válido, se debe informar al usuario de los defectos del modelo tentativo obtenido.

• **Ejemplo práctico de la estrategia de inflado axonométrico.**

Para entender mejor el método de inflado axonométrico se detalla el procedimiento para la imagen de la figura 85. Inicialmente la imagen de tipología cuasi-normalón de la figura 85a, es transformada a normalón, resultando el conjunto de aristas y vértices que se muestra en la figura 85b. En una lista inicialmente vacía se irán almacenando los vértices cuya coordenada “Z” esté ya evaluada y que denominaremos LV. El proceso se inicia seleccionando al azar uno de los vértices de la imagen en el que concurren tres aristas reales. Supongamos que sea éste el vértice 1, al que se le aplica una coordenada “Z” determinada, en nuestro caso $Z=0$ ($LV=1$).



a) Imagen de tipología cuasi-normalón b) Imagen tras la segunda transformación

Figura 85. Aplicación del inflado axonométrico

A continuación se determina la orientación de una arista arbitraria que concurra en el vértice 1, por ejemplo la arista 1-2, de acuerdo con los criterios establecidos en la figura 73 (LV = 1, 2). Una vez establecida esta primera arista se evalúan las coordenadas Z del resto de las aristas que concurren en los vértices ya definidos utilizando como arista de referencia la 1-2. De esta manera se obtienen las coordenadas Z de los vértices 8 y 14 (LV=1,2,8,14).

El proceso se repite extrayendo los vértices evaluados en “LV” tal y como se muestra en la tabla 86.

Vértice evaluado	Arista de referencia	Vértices obtenidos	LV
1	-	1, 2	1,2
2	1-2	3, 20	1, 2, 8, 14, 3, 20
8	1-8	7, 9	1, 2, 8, 14, 3, 20, 7, 9
14	1-14	13	1, 2, 8, 14, 3, 20, 7, 9, 13
3	2-3	4, 19	1, 2, 8, 14, 3, 20, 7, 9, 13, 4, 19
7	7-8	6, 15	1, 2, 8, 14, 3, 20, 7, 9, 13, 4, 19, 6, 15
9	8-9	10	1, 2, 8, 14, 3, 20, 7, 9, 13, 4, 19, 6, 15, 10
13	13-14	12, 21	1, 2, 8, 14, 3, 20, 7, 9, 13, 4, 19, 6, 15, 10, 12, 21
4	3-4	5, 18	1, 2, 8, 14, 3, 20, 7, 9, 13, 4, 19, 6, 15, 10, 12, 21, 5, 18
10	9-10	11, 24	1, 2, 8, 14, 3, 20, 7, 9, 13, 4, 19, 6, 15, 10, 12, 21, 5, 18, 11, 24

Tabla 86. Vértices analizados por inflado axonométrico en un modelo cuasi-normalón

Notar que el proceso finaliza cuando todos los nodos de la imagen de partida han sido evaluados. Y que los nodos ficticios (nodos 15 al 26) añadidos a la imagen para permitir la evaluación en aquellos nodos en los sólo concurren dos aristas, no son evaluados durante la aplicación del método.

5.2.3 Inflado por niveles.

El inflado por niveles, está inspirado en el etiquetado de vértices. Su característica fundamental es la de tratarse de un método que, aunque no obtiene un modelo final, puede ser aplicado a cualquier modelo sea cual sea su tipología.

La reconstrucción de modelos mediante inflado por niveles es un método heurístico simple que consiste en el inflado directo, no iterativo, de la figura para obtener el modelo.

Para conseguir que el modelo que se obtenga al inflar la figura sea el modelo psicológicamente aceptable se debe adoptar un criterio de asignación de las coordenadas Z. El criterio adoptado en el método propuesto de inflado por niveles es asignar, a las coordenadas Z de los vértices, valores prefijados que dependan de sus tipologías.

Las tipologías de los vértices ya han sido utilizadas con anterioridad en los métodos de reconstrucción. En concreto, se han utilizado para validar las figuras a reconstruir en los denominados métodos de etiquetado [Wang93]. En el método propuesto las tipologías definidas son básicamente la “Y” y la “W”, aunque se consideran las diferentes orientaciones (por lo que la “Y” y la “Y invertida” se consideran tipologías distintas).

Sin embargo, la clave del método no se centra en las tipologías definidas, sino que consiste en aceptar que hay un alto grado de correspondencia entre la tipología de cada vértice de la figura y la coordenada Z que debe corresponderle al vértice del modelo a reconstruir. Por tanto se supone que las tipologías de los vértices se pueden ordenar secuencialmente en “niveles”, haciendo corresponder cada nivel con un valor de coordenada Z creciente. De forma que los vértices de la figura cuya tipología corresponda a la de mayor nivel tendrán la mayor coordenada Z en el modelo y viceversa.

- ***Clasificación por niveles de las tipologías***

La correspondencia entre las tipologías y los consiguientes niveles de vértices que hemos considerado se pueden justificar intuitivamente a partir de la observación directa de la figura 87, que muestra la imagen de partida (que es la imagen contenida en el plano XY del sistema de coordenadas usado) y el modelo psicológicamente aceptable de un objeto poliédrico simple. En la figura se indica como se han etiquetado los seis niveles en que se reparten los vértices del objeto (desde el 0 hasta el 5).

Es obvio que la asignación de los niveles puede cambiar totalmente en función del cabeceo que tenga el modelo tridimensional respecto a la figura plana de partida. El cabeceo es patente al mirar la vista lateral YZ de la figura 87, y comprobar la

inclinación de aquellas aristas que en la figura de partida son “verticales” (paralelas al eje vertical del plano XY) y que en el modelo están inclinadas respecto al mismo eje Y.

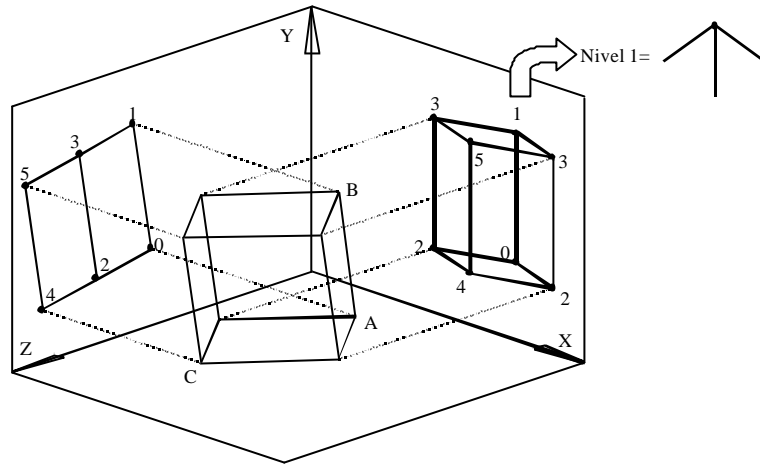


Figura 87. Relación entre niveles y tipologías de vértices

Este efecto no es casual. Recordemos que el inflado es la operación inversa de la proyección, y que antes de proyectar un objeto sobre un plano, se gira intencionadamente el objeto (evitando que sus caras y aristas queden paralelas al plano de proyección) para evitar que la proyección sea de tipo ortográfico, característico de los sistemas multivistas diédricos. Sin embargo el cabeceo que se aplica es libre, por lo que cualquier variación del mismo dará lugar a diferentes proyecciones del mismo objeto. De hecho, la variación del *cabeceo* y la *guiñada* (por seguir con la analogía de los términos náuticos) determinan todas las posibles representaciones axonométricas. Por ejemplo, podemos medir el cabeceo con el ángulo que forma la arista AB con el eje Y. Entonces observamos que inclinando un poco más el objeto de la figura 87 (aumentando su cabeceo), los vértices que se han etiquetado de nivel 3 pasarían a tener mayor coordenada Z que el que se ha etiquetado de nivel 4, aunque sus tipologías no cambiarían hasta que el cambio fuera muy grande. Análogamente, si la guiñada la medimos como el ángulo que forman la línea AC con el eje Z, observamos que al aumentar dicho ángulo los dos vértices de la izquierda dejarán de tener los mismos niveles que los dos de la derecha.

El ángulo de cabeceo queda implícitamente contenido en la figura resultante de la proyección, pero el sentido del cabeceo (hacia delante o hacia detrás) no queda

patente. Por tanto, al hacer la operación inversa, al inflar, debemos elegir uno de los dos posibles sentidos de cabeceo. Si el cabeceo se hace hacia delante, la parte superior quedará más adelantada que la inferior, por lo que sus vértices deben tener mayor coordenada z que los vértices similares de la parte inferior. Si el cabeceo se hace hacia atrás, se invierte el criterio. En la definición de niveles se ha optado por suponer que el cabeceo siempre se hace hacia delante. Esta suposición no limita el resultado, dado que el cabeceo hacia atrás da lugar al mismo modelo, aunque afectado de una inversión de Necker (tal como se muestra en la figura 88). En cualquier caso, esta suposición no hace sino asumir la tendencia psicológica para objetos pequeños, en relación al tamaño del hombre, de observar estos de arriba abajo.

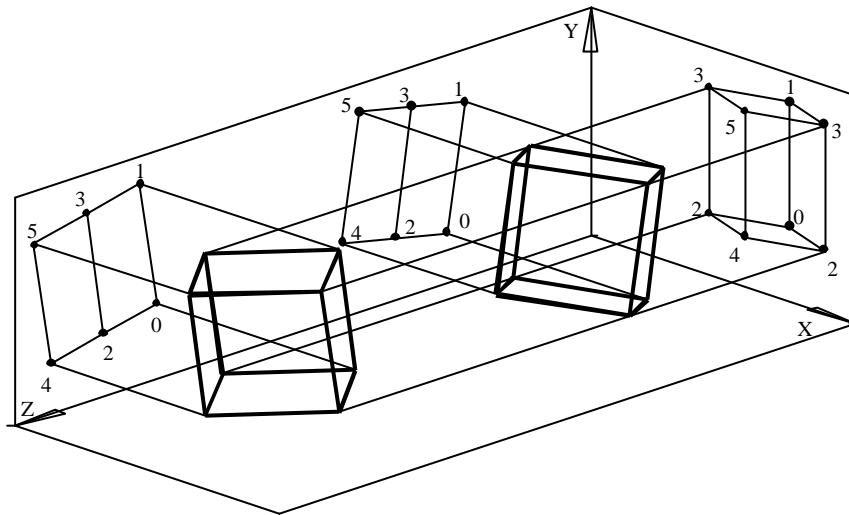


Figura 88. Inflado de un modelo y su inversión de Necker

En cuanto a la guiñada, el criterio adoptado supone aceptar que la tendencia psicológica es minimizar dicho ángulo para mantener la mayor simetría posible.

En definitiva, las seis tipologías definidas se muestran en la tabla 89.

Nivel 5	Nivel 4	Nivel 3	Nivel 2	Nivel 1	Nivel 0

Tabla 89. Clasificación por niveles de tipologías de vértices.

La clasificación mostrada en la tabla 89 se ha obtenido suponiendo vertical la *arista predominante* de cada tipología. El concepto de arista predominante se puede entender comparando las tipologías de dos niveles como 5 y el 0, que corresponden ambos a formas de “Y”. Se observa que en un caso la arista central, que es la que en este caso denominamos predominante, está orientada hacia abajo y en el otro hacia arriba. La existencia de una arista predominante es consecuencia de que es habitual que los objetos se representen buscando intencionadamente la apariencia psicológica de que el objeto parezca orientado verticalmente.

Por tanto, el método requiere contemplar aquellos casos en los que no se respeta ese criterio, como en el ejemplo que se muestra en la figura 90a. Por ello, el algoritmo desarrollado detecta la dirección predominante y aplica la rotación necesaria para que dicha dirección quede vertical. Si analizamos la imagen que se muestra en la figura 90, claramente puede observarse que como resultado del análisis de la tipología del vértice A en su orientación inicial (figura 90b), resultaría nivel 4, si bien aplicando una rotación para hacer coincidir la dirección predominante de la figura 90a con la vertical, debería clasificarse como nivel 2, como se muestra en la figura 90c.

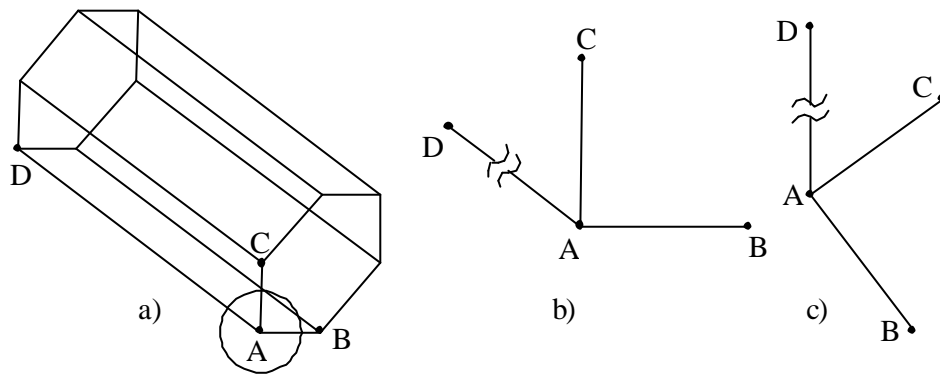


Figura 90. Aristas predominantes y giro de los vértices.

- **Detección de aristas predominantes.**

Como se ha comentado con antelación, la clasificación de niveles mostrada en la tabla 89 está sujeta al criterio supuesto de que la representación de un objeto se realiza de manera tal que este aparezca orientado verticalmente. Para ello hemos recurrido a un

caso particular de transformación. Las transformaciones son habituales en geometría descriptiva, porque sirven para simplificar problemas reduciéndolos a un conjunto mínimo de casos generales y presentándolos del modo más conveniente. Pero, hay que recordar que las reglas de transformación descritas persiguen que se cambie la representación *sin* cambiar el objeto representado.

En concreto hemos implementado un algoritmo capaz de realizar la rotación precisa para chequear los niveles de vértices para aquellos casos en los que la orientación del objeto no sea la adecuada. El algoritmo que se ha implementado para realizar una rotación previa de la imagen, para orientar ésta de manera que verifique el criterio de verticalidad, esta basado en la detección previa de la tipología del objeto y actúa en función de la clasificación mostrada en la tabla 63. Los criterios adoptados para la determinación de dicha arista predominante son los siguientes:

1. Objetos de tipología normalón y cuasi-normalón.

Los objetos de tipología normalón será girados de manera que aquel enlace cuya orientación sea la más próxima a 90° o 270° actúe como arista predominante. En la figura 91 se muestra un ejemplo del giro provocado a una imagen de tipología normalón donde se verifica que $\theta_1 < \theta_2$ y $\theta_1 < \theta_3$.

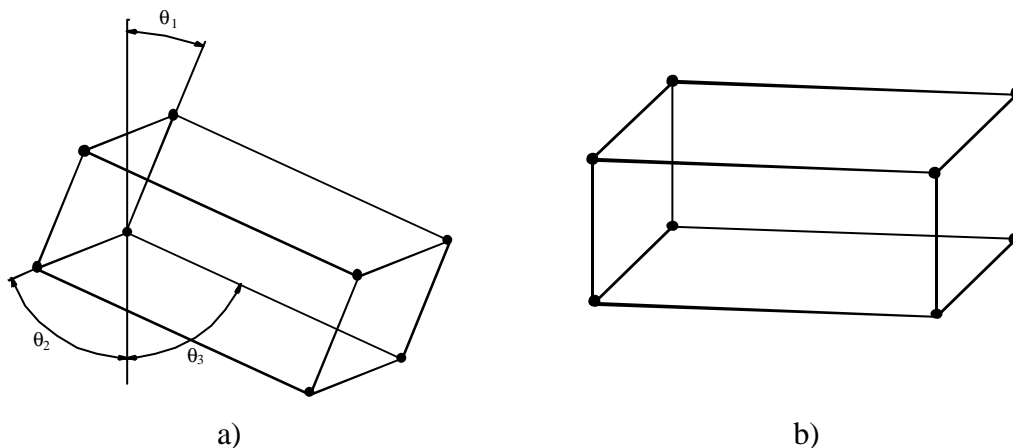


Figura 91. Rotación en un objeto normalón para orientar la arista predominante
a) Imagen con orientación original b) Imagen girada para detectar niveles en vértices

2. Objetos de tipología prismática.

Para conseguir la orientación vertical de un objeto de tipología prismática, se girará la imagen de partida de manera que la dirección predominante quede orientada verticalmente según se muestra en la figura 92.

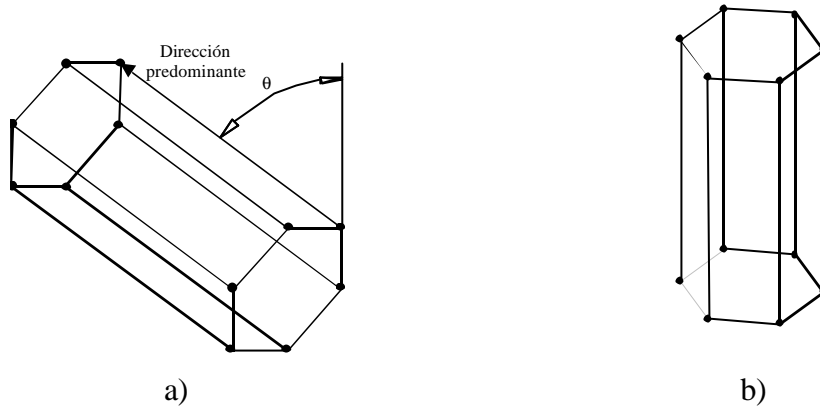


Figura 92. Rotación en un objeto prismático para orientar la arista predominante
 a) Imagen con orientación original b) Imagen girada para detectar niveles en vértices

3. Objetos de tipología piramidal.

En el caso de objetos de tipología piramidal, se realizan giros independientes en cada una de los enlaces que unen el vértice de la pirámide con su directriz, de manera que esta arista considerada como predominante quede en la orientación vertical. Los giros se realizan de manera tal que los arcos descritos posean la menor longitud posible para alinear la arista predominante con 90° o 270° . En la figura 93 se muestran los giros realizados para cada uno de las seis aristas predominantes de la imagen.

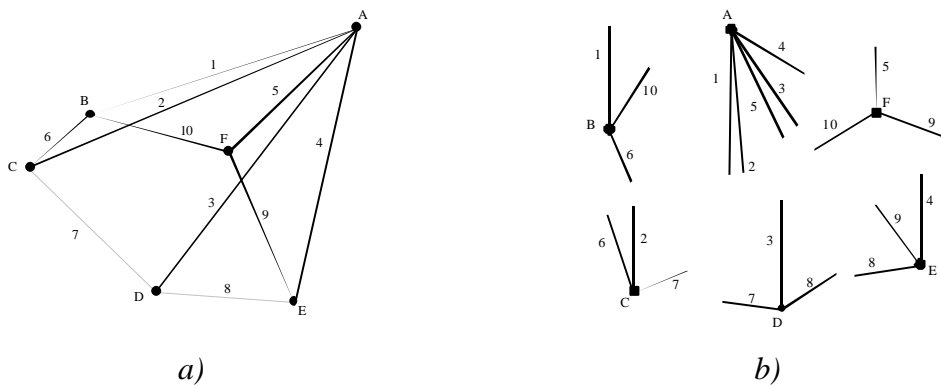


Figura 93. Rotación en un objeto piramidal para orientar la arista predominante
 a) Imagen con orientación original b) Imagen girada para detectar niveles en vértices

4. Objetos de tipología indefinida.

Se consideran como objetos de tipología indefinida todos aquellos que no pueden ser considerados como objetos de alguna de las tipologías anteriores. Para este tipo de objetos se selecciona como arista predominante aquella arista cuya orientación aparezca con mayor frecuencia en la imagen. En el caso de existir orientaciones con igual frecuencia de aparición se selecciona aquella cuya orientación presente una menor desviación con respecto a 90° o a 270° .

Como ejemplo, en la imagen de la figura 94a existen definidas ocho direcciones de las cuales, d_A y d_B se repiten con una mayor frecuencia (tres enlaces presentan orientación de acuerdo con d_A y otros 3 enlaces presentan orientación de acuerdo con d_B). De ambas orientaciones, d_B se aproxima más a la verticalidad y por tanto será la seleccionada como arista predominante. Dicha arista provoca un giro para situar la imagen verticalmente como se muestra en la figura 94b y determinar los niveles de sus vértices. El modelo es girado para orientar dichas aristas hacia 270° dado que esta nueva orientación implica recorrer un menor ángulo que si se orientasen las aristas hacia 90° .

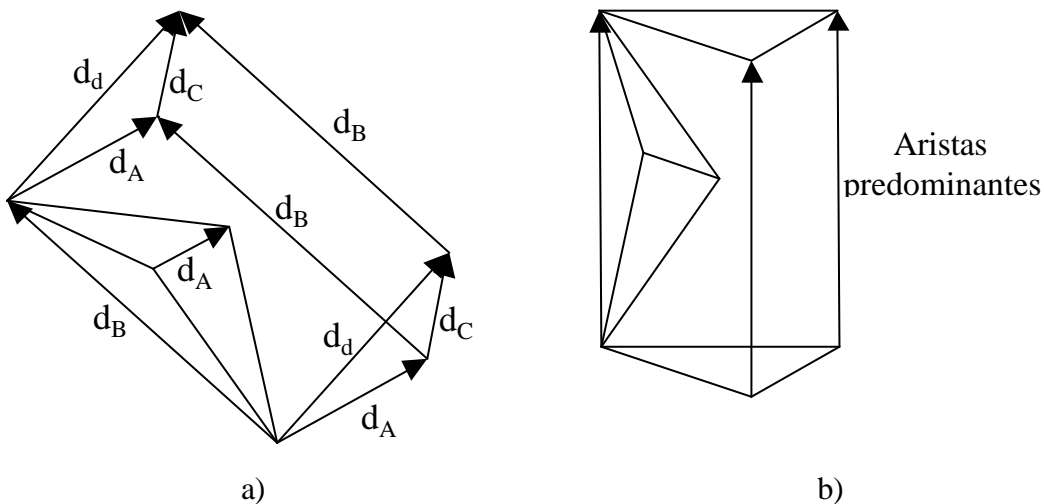


Figura 94. Rotación en un objeto indefinido para orientar la arista predominante
 a) Imagen con orientación original b) Imagen girada para detectar niveles en vértices

- **Determinaciones analíticas de tipologías.**

Las distintas tipologías representadas en la tabla 89 pueden ser agrupadas en función de la orientación de la arista predominante y de los ángulos que forman cada una de las tres aristas que convergen en el vértice. Por ello el primer paso preceptivo es

la determinación de los ángulos de cada una de las aristas, definidas por un vértice inicial (V_i) y un vértice final (V_f). Debe considerarse como vértice inicial, el vértice común a todas las aristas (figura 95).

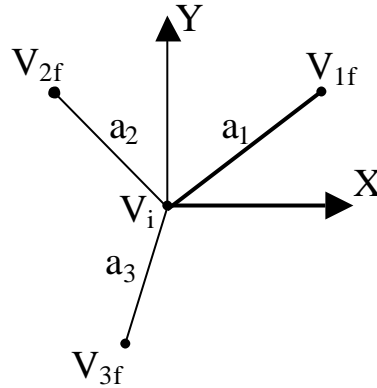


Figura 95. Vértices iniciales y finales de aristas en la determinación de ángulos

Las distintas tipologías pueden agruparse en dos grandes bloques:

1. Si la arista predominante forma un ángulo de 90° serán posibles las tipologías 4, 2 y 0. Las reglas aplicadas para la determinación del nivel al que pertenece el vértice se muestra en la tabla 96.
2. Si la arista predominante forma un ángulo de 270° serán posibles las tipologías 5, 3 y 1. Las reglas aplicadas para la determinación del nivel al que pertenece el vértice se muestra en la tabla 97.

En las tablas 96 y 97 se ha utilizado la siguiente terminología:

- θ_m representa el menor de los ángulos de entre los definidos por las tres aristas concurrentes en el vértice.
- θ_{med} representa el ángulo medio de entre los definidos por las tres aristas concurrentes en el vértice.
- θ_M representa el mayor de los ángulos de entre los definidos por las tres aristas concurrentes en el vértice.

Lo dicho permite elaborar una tabla con todas las reglas para determinar tipologías:

Condición	Tipologías	Imagen
a) Si la arista predominante es la de ángulo θ_M	2	
b) Si la arista predominante es la de ángulo θ_m	4, 2 o 0	
b1) Si $\theta_{med} < 180^\circ$		
b1.1) Si $\theta_M < 270^\circ$	2	
b1.2) Si $\theta_M > 270^\circ$ y $\theta_M < \theta_{med} + 180^\circ$	0	
b1.3) Si $\theta_M > 270^\circ$ y $\theta_M > \theta_{med} + 180^\circ$	4	
b2) $\theta_{med} > 270^\circ$	2	
b3) Si $180^\circ < \theta_{med} < 270^\circ$	0	
b3.1) Si $\theta_M < 270^\circ$	2	
b3.2) Si $\theta_M > 270^\circ$	0	
c) Si la arista predominante es la de ángulo θ_{med}	4, 2 o 0	
c1) Si $\theta_M < \theta_m + 180^\circ$	4	
c2) Si $\theta_M > \theta_m + 180^\circ$ y $\theta_M < 270^\circ$	0	
c3) Si $\theta_M > \theta_m + 180^\circ$ y $\theta_M > 270^\circ$	2	

Tabla 96. Reglas analíticas para la determinación de tipologías para aristas predominantes con ángulo de 90 grados.


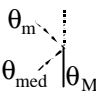
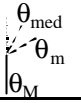
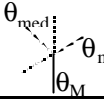
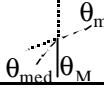
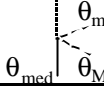
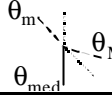

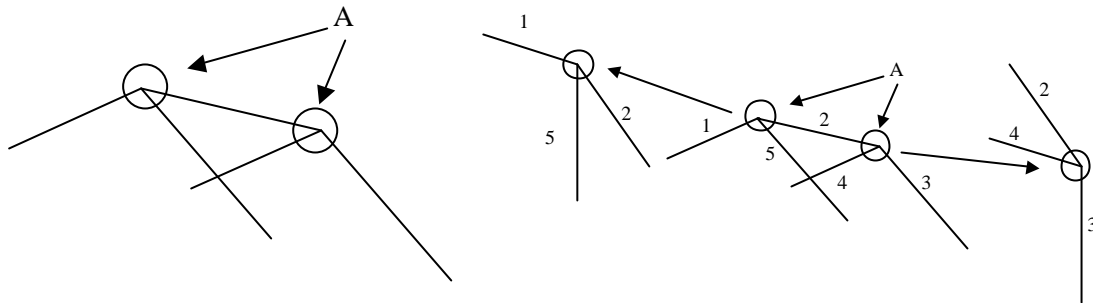
Condición	Tipologías	Imagen
a) Si la arista predominante es la de ángulo θ_m	3	
b) Si la arista predominante es la de ángulo θ_M	5, 3 o 1	
B1) Si $\theta_m > 90^\circ$	3	
B2) Si $\theta_m < 90^\circ$	3, 5 o 1	
B2.1) Si $\theta_{med} < 90^\circ$	3	
B2.2) Si $\theta_{med} > 90^\circ$ y $\theta_{med} < \theta_m + 180^\circ$	5	
B2.3) Si $\theta_{med} > 90^\circ$ y $\theta_{med} > \theta_m + 180^\circ$	1	
c) Si la arista predominante es la de ángulo θ_{med}	5, 3 o 1	
C1) Si $\theta_m < 90^\circ$	3	
C2) Si $\theta_m > 90^\circ$	5 o 1	
C2.1) Si $\theta_M > \theta_m + 180^\circ$	5	
C2.2) Si $\theta_M < \theta_m + 180^\circ$	1	

Tabla 97. Reglas analíticas para la determinación de tipologías para aristas predominantes con ángulo de 270 grados.

No obstante podría ocurrir que en determinados vértices ninguna de las aristas presente la orientación deseada de 90° o 270° como ocurre en los vértices de la figura 98a.

Para esos casos, se prevé la necesidad de girar tales aristas de modo que alguna de ellas quede en la orientación adecuada para seleccionar el nivel al que pertenece. El giro de las aristas se efectuará de tal forma que el ángulo descrito sea el menor posible,

es decir, se giraran todas las aristas de modo que aquella que se encuentre más próxima a los 90° o 270° sea la que se sitúe verticalmente (figura 98b).

a) *Imagen original*b) *Aristas en vértice A giradas para conseguir arista predominante*Figura 98. *Vértices sin aristas predominantes*

- **Correspondencia entre coordenadas Z y niveles.**

Tanto o más importante que definir la tipología de cada vértice es asignar valores a las coordenadas z de los vértices de cada nivel.

Llegados a este punto, hay que hacer algunas consideraciones:

1. El método, tal como se ha descrito, sólo es válido cuando los vértices son triedros (siempre concurren tres aristas en un vértice).
2. El método se ha justificado con modelos simples, que cumplen dos condiciones particulares que hacen que no se puedan generalizar:
 - Las dimensiones en las direcciones de los tres ejes son semejantes.
 - El poliedro es convexo.

Para generalizar la aplicación del método cuando no se cumplen estas condiciones hemos elaborado dos estrategias más sofisticadas que se describen a continuación.

Niveles Fijos

La estrategia más sencilla, que denominamos de niveles fijos, parte del criterio perceptivo de que la figura tiende a mantener la proporción entre las tres dimensiones principales del objeto (ancho, alto y profundo). Es decir, se asume que el dibujante realiza intencionadamente el acto de respetar las proporciones. En otras palabras, el punto de vista se elige a propósito para poner de manifiesto la proporción o desproporción que pueda tener el objeto. Por tanto, podemos suponer que la profundidad (que corresponde con el valor de $\Delta z = Z_{\max} - Z_{\min}$ en el sistema de referencia adoptado) debe tener el mismo orden de magnitud que la anchura ($\Delta x = X_{\max} - X_{\min}$) y la altura ($\Delta y = Y_{\max} - Y_{\min}$).

En realidad, si el objeto está representado en perspectiva, la “caja” que lo encierra (de caras paralelas a los planos coordenados y dimensiones Δx , Δy , Δz) no tiene la misma orientación que el objeto, porque el objeto, debido al cabeceo, estará inclinado respecto a la caja. Pero definir una caja proporcional favorece que el objeto encerrado en ella también sea proporcional. Por ello, adoptamos el criterio de que $\Delta z = \max(\Delta x, \Delta y)$.

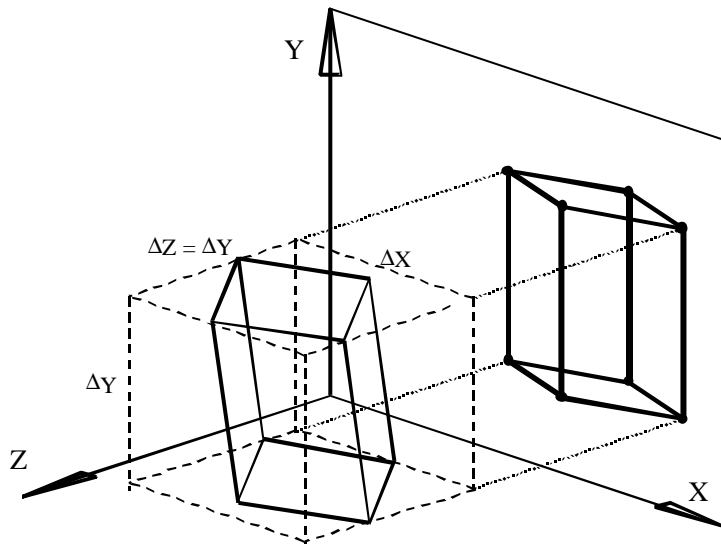


Figura 99. Asunción de proporcionalidad

Además, al implementar este criterio, hay que tener en cuenta que la coordenada “Z” de uno de los vértices se puede fijar a un valor arbitrario sin pérdida de generalidad, dado que la figura no cambia al cambiar ese valor, tan sólo se desplazada en la dirección perpendicular al plano de proyección. Por tanto, se puede fijar $Z=0$ para los vértices de nivel mínimo y $Z=$ máximo $(\Delta x, \Delta y)$ para los vértices de mayor nivel.

Por último, si se asume que los niveles de los vértices están bien escalonados, parece recomendable utilizar un escalonamiento uniforme para asignar las coordenadas Z de todos los niveles intermedios. Entonces $\Delta Z/n$ es el incremento proporcional de coordenada Z entre dos niveles consecutivos. Siendo n el número saltos de nivel (que coincide con el número del nivel más alto, porque la numeración empieza con el nivel 0).

El inflado definido con éste criterio da buenos resultados en cuerpos poliédricos convexos con un alto grado de regularidad. En particular funciona bien para cuerpos normalones y prismáticos. En otros casos, las aproximaciones obtenidas aun pueden ser válidas como modelos previos para comenzar un inflado iterativo por optimización, pero no son válidas como modelos finales porque pueden estar bastante distorsionados.

Sin embargo, existen familias de poliedros a los que, la aplicación del inflado por niveles no permite una clara definición de su tipología o provoca efectos inadecuados. A modo de ejemplo, en la figura 100 se observan dos vértices que serían etiquetados iguales pese a que deberían tener diferente coordenada Z. La razón de esta aparente limitación del método es que se está aplicando a un modelo que no es convexo. En particular, se trata de un poliedro con una “hendidura”.

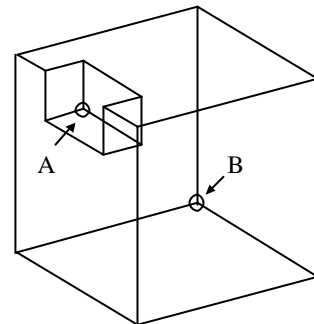


Figura 100. Poliedro con hendidura

El problema es que el método propuesto define a los vértices A y B como nivel 0. Pero, en realidad, deberían tener una coordenada Z distinta. El vértice A pertenece a una ranura, por lo que está necesariamente a mayor coordenada Z que el vértice B, dado que éste es el vértice posterior del elemento macizo que contiene a dicha ranura.

Niveles incrementales.

Una evolución del inflado por niveles de vértices es lo que denominamos estrategia de “niveles incrementales”. La diferencia se centra en que se corrigen las coordenadas Z de los vértices según la distancia que los separa de los vértices a los que están conectados. Es decir, que la coordenada Z que se asigna a un vértice no depende sólo de la tipología del mismo. Depende tanto de la *diferencia de tipología* respecto a los vértices a los que está conectado, como de la distancia que le separa de los mismos.

Esta propuesta se justifica porque consideramos el invariante de proporcionalidad de la proyección axonométrica (que dice que las longitudes de dos enlaces paralelos mantienen la misma relación de proporcionalidad al proyectarlos) y la generalizamos aplicando una regla perceptiva de proporcionalidad, asumiendo que la proporcionalidad se mantiene aunque los enlaces no sean paralelos. Para ello, estamos volviendo a aceptar que la tendencia psicológica al dibujar es seleccionar puntos de vista tales que mantengan las proporciones en las tres direcciones fundamentales.

En consecuencia, la coordenada Z_P de un vértice P se puede obtener de forma incremental respecto a la de otro vértice Q al que esté conectado por una arista, de acuerdo con la expresión:

$$Z_P = Z_Q + PQ * (Nivel_P - Nivel_Q) * \Delta z/n \quad (49)$$

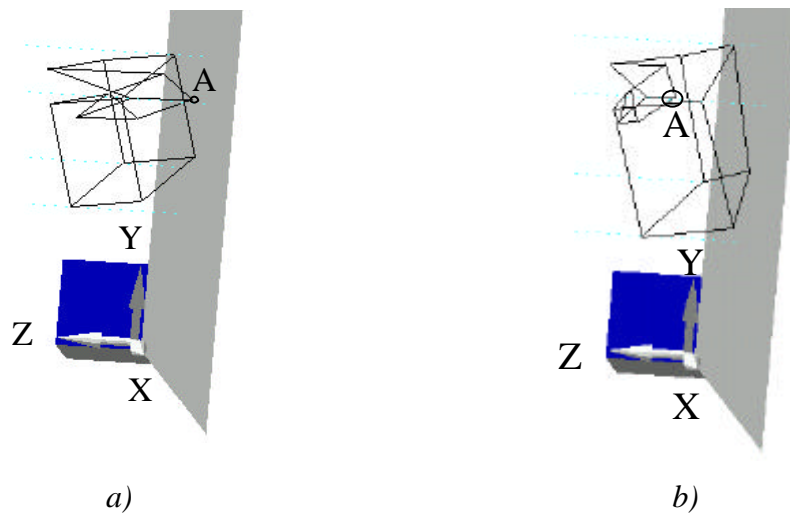
donde:

- PQ representa la longitud del enlace que une los dos vértices en la figura (es decir, la longitud de la “proyección” de la arista).
- Nivel_P-Nivel_Q representa el incremento (número de saltos) de niveles entre los vértices unidos por la arista.
- n es el número máximo de saltos de nivel (que coincide con la cantidad de niveles menos uno). De modo que “ $\Delta z/n$ ” es el incremento proporcional de coordenada Z entre dos niveles consecutivos.

Este método es dependiente de la secuencia seguida para asignación de coordenadas. En efecto, la coordenada Z de cada vértice (salvo la del primero) se asigna a partir de la de uno de los vértices conectados a él, de forma que la secuencia de

asignación de coordenadas da lugar a recorrer todos los vértices de la figura, siguiendo caminos cuya elección es casi aleatoria. En realidad se parte de un vértice de nivel 0 y se va recorriendo la lista de vértices conectados a él. El orden de la lista es el de creación de cada vértice. Por tanto, una misma figura siempre se analiza igual, pero dos figuras iguales que se hayan dibujado con diferente orden darían lugar a resultados distintos. Pese a ello, este método incremental parece dar mejores resultados que el de niveles fijos en figuras cóncavas como la mostrada en la figura 100.

En la figura 101 se muestra la diferencia entre el modelo inflado con asignación de coordenadas fijas y con coordenadas incrementales para el ejemplo de la figura 100. Se observa que el inflado incremental de la figura 101b reduce notablemente la distorsión provocada en la figura 101a por el vértice que en la figura 100 se había etiquetado como vértice A.



*Figura 101. Poliedro con hendidura
a) inflado por niveles fijos b) inflado por niveles incrementales*

La figura 102 representa la evolución del proceso de reconstrucción del modelo mediante inflado previo por niveles incrementales y posterior proceso de optimización.

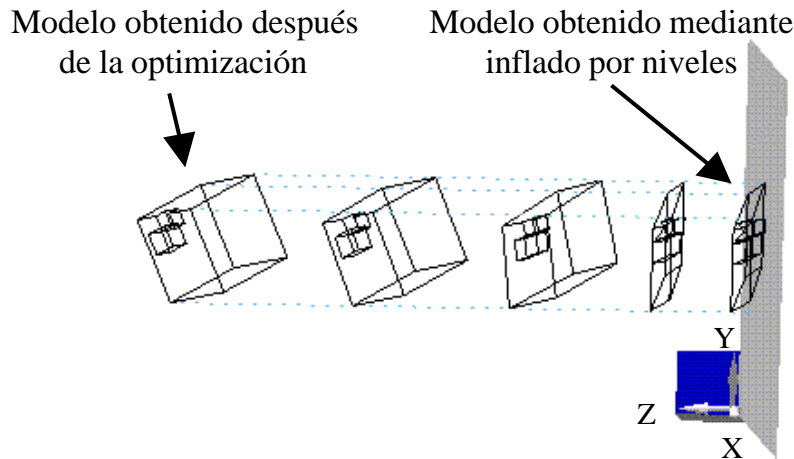


Figura 102. Inflado previo por niveles y optimización

- **Ampliación del método.**

Hemos realizado una doble ampliación al método propuesto. La primera de ellas hace referencia a las tipologías de vértices definidas. Para ello se utiliza la información sobre la visibilidad de las aristas del modelo en la proyección dada que podrá ser adquirida a partir de un algoritmo implementado de detección de aristas ocultas que será descrito en capítulos posteriores.

La información de la visibilidad de las aristas en la imagen, diferenciando entre aristas visibles y ocultas, permite ampliar las tipologías de vértices definidas en la tabla 89 diferenciando entre vértices ocultos (si todas las aristas que concurren en él son ocultas), vértices parcialmente ocultos (si concurren aristas visibles y aristas ocultas), y vértices visibles (si todas las aristas que concurren en el vértice son visibles).

De esta manera se establece una nueva clasificación de tipologías de vértices donde a los vértices ocultos se le asignan los niveles inferiores, es decir, de menor coordenada "Z", y a los vértices visibles se le asignan los niveles superiores.

La nueva clasificación de tipologías quedaría como se muestra en la tabla 103.

Nivel 17	Nivel 16	Nivel 15	Nivel 14	Nivel 13	Nivel 12
Nivel 11	Nivel 10	Nivel 9	Nivel 8	Nivel 7	Nivel 6
Nivel 5	Nivel 4	Nivel 3	Nivel 2	Nivel 1	Nivel 0

Tabla 103. Clasificación por niveles de tipologías de vértices en imágenes con aristas ocultas.

Además se ha realizado una aproximación del método que permite clasificar dentro de las tipologías definidas a aquellos vértices en los que concurren más de tres aristas. Para ello, cuando en un vértice concurren cuatro o más aristas se realiza una selección de las aristas que serán consideradas para definir la tipología del vértice. Esta selección se realiza en los dos siguientes pasos:

- En primer lugar se seleccionan aquellas aristas que abarcan entre sí en mayor ángulo posible (aristas 1 y 5 de la figura 104).
- La tercera aristas seleccionada será aquella cuya diferencia angular se aproxime más al ángulo medio descrito por las anteriormente seleccionadas (arista 2 de la figura 104).

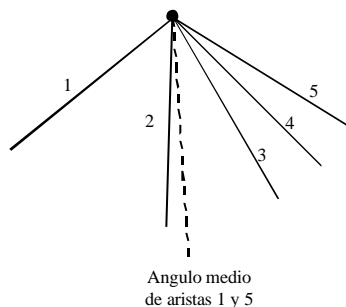


Figura 104. Selección de aristas que definen la tipología de un vértice

5.2.4 Inflado por plano oblicuo.

El inflado por plano oblicuo es un método heurístico que permite la obtención de modelos tentativos de tipología exclusivamente piramidal. Esta estrategia pretende crear un modelo previo que verifique dos propiedades fundamentales de los modelos piramidales: la planicidad del *polígono directriz* de la pirámide y la orientación de la línea que une el vértice de la pirámide y el centro geométrico de su polígono directriz que en adelante denominaremos *eje de la pirámide*.

Por polígono directriz de la pirámide se entiende el conjunto de enlaces que representan la directriz de la superficie reglada.

Evidentemente el primer paso consiste en detectar que nodo es considerado como vértice de la pirámide, y que nodos y enlaces definen la directriz de la pirámide. Para ello se parte de la propia definición de la tipología del modelo. En la tabla 63 se definían como modelos piramidales aquellas representaciones en las que en un nodo concurrían un número mayoritario de enlaces. Dicho nodo será considerado como vértice de la pirámide. Los enlaces que unan cualquier otro nodo de la imagen se consideran directriz de la pirámide.

El método de inflado en sí se realiza en dos fases que garantizan ambas propiedades (planicidad y orientación):

- La planicidad del polígono directriz se asegura asumiendo que la pirámide se encuentra apoyada en un plano π que forma un ángulo de 45° respecto del plano XY, y un ángulo de 90° respecto del plano YZ. Esta inclinación se ha elegido tomando un valor medio para el cabeceo que el modelo sufre cuando es inflado. Conocida la orientación del plano, sólo falta por fijar su posición. Por tanto, asignar una coordenada “Z” arbitraria a uno de los nodos del polígono directriz, permite obtener las coordenadas “Z” asociadas a al resto de los nodos del polígono mediante la ecuación:

$$Z_i = Z_r + (Y_r - Y_i) \operatorname{tg} 45^\circ \quad (50)$$

donde Z_i e Y_i representan la coordenadas Z e Y respectivamente de cada uno de los nodos del polígono directriz, y Z_r e Y_r representan las coordenadas Z e Y,

respectivamente, del nodo del polígono al que se le ha asignado una coordenada Z arbitraria.

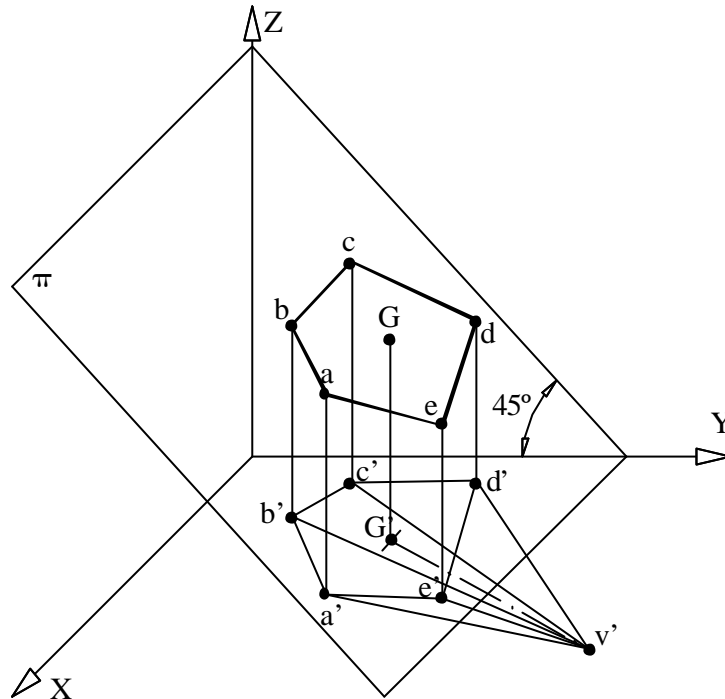


Figura 105. Planicidad del polígono directriz de una pirámide

- En la orientación del eje de la pirámide, se parte de la idea de que la mayoría de las pirámides de uso práctico tienen el eje normal a su directriz, y de la asunción de verticalidad que fue tratada con anterioridad en la regularidad de verticalidad de líneas. Por ello, se supone que una imagen representa la proyección de una pirámide recta si la proyección del eje resulta paralela al eje Y del plano que contiene la imagen, en cuyo caso, dicho eje vendrá representado por una arista que sea perpendicular al plano π que contiene al polígono directriz. Cualquier desviación α que la proyección del eje de la pirámide presente respecto del eje Y en el plano de la imagen, se presupone que representa una desviación de dicho eje frente a la vertical, por lo que se refleja en una desviación del eje de la pirámide en el modelo (figura 106).

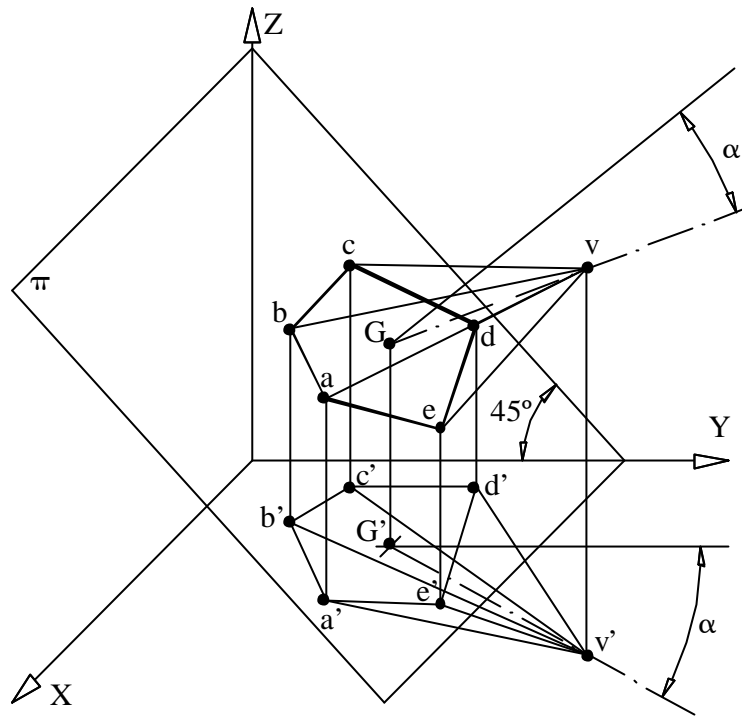


Figura 106. Orientación del eje de la pirámide en el inflado previo por plano oblicuo

DETECCIONES PREVIAS EN LA IMAGEN

Como ya queda dicho, este trabajo se ha centrado en la Reconstrucción Geométrica de sólidos a partir de una única vista que se corresponde con la proyección axonométrica de un modelo.

Sin embargo, tal y como se planteado hasta el momento, el proceso de optimización precisa de información que puede que no se encuentre explícitamente referenciada en la base de datos asociada a la imagen. La imagen se almacena, en formato digital, como un conjunto de enlaces conectados en nodos. Por tanto, la información asociada a vértices y aristas del modelo a reconstruir si que tiene un antecedente explícito en la información asociada a la imagen. Sin embargo, para obtener un modelo BRep se requiere también información de caras, y en la imagen de partida no existe ningún tipo de información que permita la identificación de las caras del modelo. Hay que recordar que las caras se precisan para aplicar regularidades tales como la planicidad de caras o la perpendicularidad de caras.

De igual forma, en los métodos de inflado previo, es decir en los métodos que permitían obtener un punto inicial o tentativo, se hacia referencia a un inflado por niveles en función de la visibilidad de las aristas de la imagen de partida. Para aplicar este método se requiere distinguir entre imágenes de aristas vistas e imágenes de aristas ocultas. En algunos casos, la imagen puede contener tal distinción, porque se suelen representar los enlaces de aristas ocultas por medio de líneas de trazos. Sin embargo, en muchos casos esta información tampoco está explícitamente recogida en la imagen de partida.

Por consiguiente, si se quiere automatizar el proceso de reconstrucción, se precisa de algoritmos que sean capaces de obtener previamente estas informaciones a partir de la imagen. En definitiva, se tratan de algoritmos que permitan obtener una representación BRep completa a partir de una representación alámbrica. La dificultad principal proviene de que se debe resolver el problema en dos dimensiones. Puesto que la información buscada es necesaria, precisamente, para obtener posteriormente un modelo tridimensional.

En este capítulo se exponen los algoritmos que han sido implementados para resolver tales problemas.

6.1 Conceptos básicos y definiciones de un grafo geométrico.

Los algoritmos que se presentan en este capítulo utilizan algunos de los recursos de la teoría de grafos.

Un *grafo* es un conjunto finito de nodos y enlaces. Los nodos se llaman vértices y los enlaces se llaman aristas. De ahí que la analogía entre un grafo y una imagen de un modelo alámbrico sea inmediata: la proyección de una arista del modelo es un enlace de la imagen, y la proyección de un vértice del modelo es un nodo de la imagen.

Algunos autores utilizan el término “unión” para indicar la proyección de un vértice del modelo. Por tanto, nodo o unión se emplea en 2D y se reserva vértice para 3D. Igualmente se usa enlace (en 2D) para indicar la proyección de una arista (3D).

Si consideramos modelos BRep, una región es la proyección de una cara. Y la frontera de una región es el polígono que limita la proyección de una cara del modelo.

No obstante es preciso resaltar algunas diferencias entre la definición genérica de grafo y el uso de estos en la Reconstrucción Geométrica. Un grafo se entiende como un conjunto de nodos y enlaces, donde los nodos son objetos simples que pueden tener un nombre y otras propiedades, y cada enlace es una conexión entre dos nodos. Por tanto, cada nodo contiene una unidad de información y cada enlace establece relaciones entre diferentes unidades de información.

En las imágenes de modelos alámbricos o BRep, la información asociada a cada nodo son las coordenadas (x, y) del mismo respecto a un sistema de referencia. Por tanto, es la *posición* de cada nodo la información que cuenta para reconstrucción.

En la figura 107 se representan dos grafos definidos por los nodos A, B, C, D, E, F, G y H.

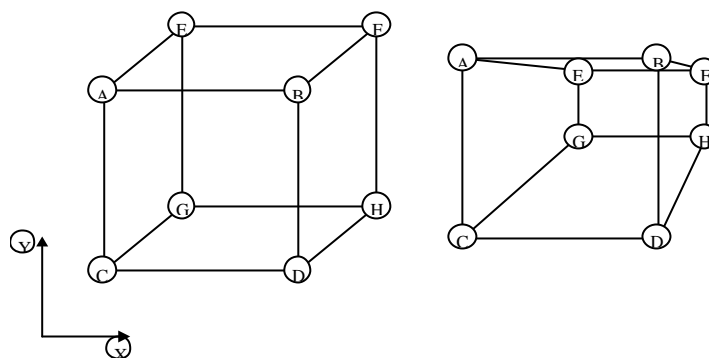


Figura 107. Grafos genéricos/geométricos

De acuerdo con la definición genérica de grafo, ambas imágenes representan un mismo grafo en tanto que el número de nodos es el mismo y las conexiones entre los distintos nodos también son idénticas. Es decir, el nodo A está conectado con los nodos B, C y E, el nodo B está conectado con los nodos A, D y F, el nodo C está conectado con los nodos A, D y G,Sin embargo, tales imágenes no son geoméricamente iguales en tanto que las formas que representan son distintas, porque las coordenadas de sus nodos no son equivalentes.

En definitiva debe establecerse una clara diferencia entre las definiciones de *grafo genérico* y *grafo geométrico*: mientras en la interpretación de un grafo genérico las coordenadas (X,Y) de sus nodos son ignoradas, éstas, tienen una importancia capital en el grafo geométrico. De hecho en geometría métrica los grafos representados en la figura 107 serían interpretados como figuras distintas.

Por otro lado, en geometría proyectiva pueden existir tanto proyecciones idénticas de distintos modelos, como proyecciones distintas del mismo modelo. De hecho, los dos grafos de la figura 107 pueden ser proyecciones diferentes de un mismo modelo.

En la teoría de grafos, hay otras definiciones relevantes para nuestro estudio. Se define el *camino* entre dos nodos de un grafo como una secuencia conexa de enlaces que producen una ruta en el grafo, que comienza en un nodo y termina en otro. Por

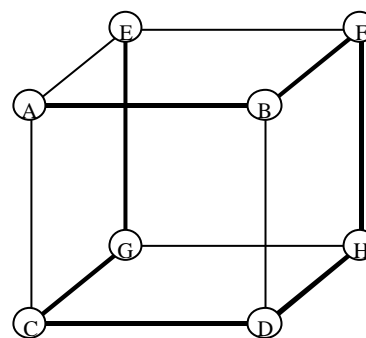


Figura 108. Definición de camino en un grafo

ejemplo ABFHDCGE es un camino desde A a E en la figura 108. Se conoce como *circuito simple* al camino en el que no se repite ningún nodo y como *circuito cíclico* o simplemente *ciclo* a aquel circuito simple con la característica de que el primer y último nodo es el mismo.

Se dice que un grafo es *conexo o simple* si existe un camino desde cada nodo hasta cualquier otro nodo del grafo. De forma intuitiva, si los nodos fueran objetos físicos y los enlaces fueran cadenas que los conectan, un grafo conexo permanecería en una sola pieza si se levantara por uno cualquiera de sus nodos. De igual forma, un grafo que no sea conexo estará constituido por dos o más grafos conexos.

Un grafo simple puede representar a un *poliedro simple* puesto que se reconoce como poliedro simple a aquel que puede deformarse para obtener una esfera, es decir, un poliedro que no tiene agujeros o “tetones”. La representación por fronteras de un poliedro simple satisface la fórmula de Euler, que expresa una relación invariable entre el número de vértices, aristas y caras de un poliedro simple:

$$V - A + C = 2 \quad (51)$$

donde V es el número de vértices, A es el número de aristas y C es el número de caras. De ahí que los poliedros simples sean también llamados poliedros Eulerianos.

Examinando la definición de grafos conexos y poliedros Eulerianos, y teniendo en cuenta los invariantes de proyección, puede afirmarse que cualquier grafo representación (desde un punto de vista general) de un poliedro Euleriano será conexo y viceversa, cualquier poliedro Euleriano, vendrá representado por un grafo conexo.

6.2 Identificación de caras en un grafo.

La detección de las caras de un objeto tridimensional a partir de una proyección axonométrica consiste en seleccionar aquellos circuitos cíclicos que determinan una cara, de entre los otros muchos circuitos existentes en el dibujo.

Cuando el objeto se representa en forma alámbrica, que es la forma más habitual de representación en las fases conceptuales del diseño, esta identificación resulta muy compleja.

El problema puede ser formulado como: dado un dibujo lineal bidimensional que representa una proyección de un modelo alámbrico, se requiere identificar aquellos circuitos cíclicos que corresponden a las caras reales de las que consta el objeto y que concuerden con la interpretación dada por un observador humano.

El planteamiento y solución del problema se representa en la figura 109. En la parte izquierda de la figura se representa el grafo de entrada consistente en un conjunto de enlaces y nodos. En la parte derecha de la figura se representan los circuitos cíclicos, obtenidos a partir del grafo, que identifican las caras en la imagen. Es importante resaltar que tanto la información de la imagen de la derecha como la de la izquierda son bidimensionales y que no se ha utilizado ninguna información del modelo tridimensional que representa.

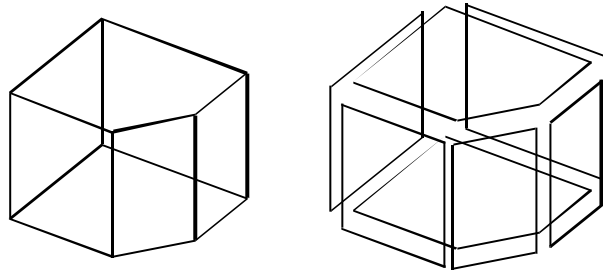


Figura 109. Determinación de caras en un grafo

6.2.1 Antecedentes en la detección de caras de un poliedro.

Por su parte el grupo de investigación REGEO ha implementado un identificador de caras 2D como módulo inicial de un reconstructor de poliedros a partir de una axonométrica oblicua. Dicho algoritmo comenzaba deduciendo propiedades del grafo geométrico como paralelismo entre enlaces, longitudes de enlaces, valencia de vértices, etc. A continuación se ejecutaba un algoritmo de búsqueda en profundidad de “bucles de caras”, a partir del cual se iban detectando ordenadamente caras de 3 lados, 4 lados, 5 lados, ...

Sin embargo, este detector de caras no era completamente automático y presenta cierta interacción con el usuario que, en un tercer paso, debía validar algunos de los bucles de caras [Gomi97(2)].

La detección de caras en una imagen es un problema que ha sido tratado ya con anterioridad por otros autores. Leclerc [Lecl92] es la primera referencia que hemos podido encontrar donde se detalla un algoritmo para la detección de caras en un modelo alámbrico, donde cada cara viene definida por un conjunto de vértices (nodos del grafo que definen un circuito cíclico). En su algoritmo se asume:

- Dos enlaces paralelos en un dibujo lineal representan aristas paralelas en el espacio. Los enlaces que conecten dos nodos de dos enlaces paralelos en el dibujo, definirán un plano en común con los enlaces paralelos.
- Un circuito cerrado y convexo que no tenga circuitos internos corresponde al contorno de una cara plana en el correspondiente modelo tridimensional. Un circuito interno a un polígono convexo se define como un circuito en el que todos los nodos son internos al polígono y no adyacentes con los nodos del polígono.
- El contorno formado por un circuito simple de líneas del dibujo que no contenga enlaces internos, se corresponde con una cara plana en el correspondiente modelo tridimensional reconstruido.

Cualquier circuito que verificase algunas de las asunciones anteriormente planteadas era considerado como cara del modelo tridimensional.

En los casos más sencillos el algoritmo parece funcionar bastante bien. Sin embargo está basado en reglas heurísticas y por tanto sujeto a posibles errores en la detección de caras. En la figura 110 se muestra un contraejemplo en el que un circuito cerrado representa erróneamente una cara del modelo.

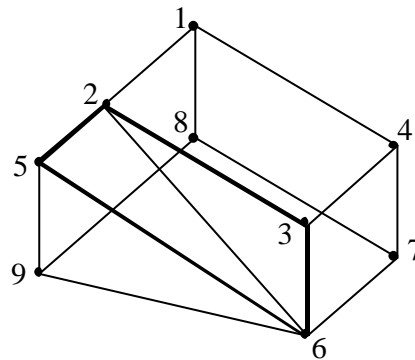


Figura 110. Contraejemplo del algoritmo de detección de caras propuesto por Leclerc

Tal y como puede observarse, los enlaces 2-3 y 5-6 representan aristas paralelas y están conectadas a través del enlace 3-6. Por tanto todos los enlaces deben pertenecer a una misma cara. Además el enlace 2-5 cierra un circuito conexo que no contiene a ningún otro circuito interno, ni a ningún otro enlace del grafo (el enlace 2-6 no se considera interno por tener nodos adyacentes al circuito analizado).

Posteriormente Sphitalni y Lipson [Sphi96] desarrollaron un nuevo algoritmo que permite identificar las caras definidas en una imagen. Desde nuestro punto de vista, este algoritmo basado en reglas geométricas es más eficiente y es por ello que ha sido seleccionado para implementarlo en REFER. En este momento vamos a exponer el algoritmo propuesto por estos autores. Posteriormente en el capítulo de ensayos y validaciones se discutirán sus inconvenientes y las mejoras que hemos implementado.

El algoritmo propuesto esta dividido en las siguientes fases:

- Identificación de circuitos fundamentales.
- Generación de las caras potenciales del grafo.
- Selección de las caras del modelo.
- Selección de la solución final.

6.2.2 Identificación de circuitos fundamentales.

En esta fase se identifican una serie de circuitos que servirán de base para la determinación de un potencial conjunto de caras.

Se denominan *circuitos fundamentales* de un grafo a todos los componentes de un conjunto de circuitos cíclicos tal que todos ellos poseen un enlace que no se encuentra en el resto y entre todos recorren todas los enlaces del grafo (figura 111). El número de circuitos fundamentales de un grafo se obtiene mediante la expresión $EN+1$; siendo E el número de enlaces y N el número de nodos. Este número se denomina *número ciclomático* de un grafo.

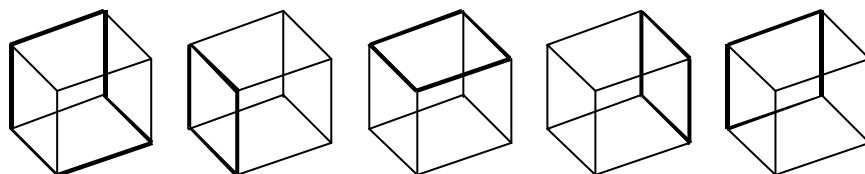


Figura 111. Circuitos fundamentales en un modelo prismático.

El conjunto de circuitos fundamentales no es único para un mismo grafo, en tanto que depende de la secuencia de búsqueda del mismo.

La característica fundamental por la que los circuitos fundamentales son importantes es porque por combinaciones de ellos podemos obtener cualquier otro circuito del grafo.

6.2.3 Generación de las caras potenciales del grafo.

En esta fase se calculan el resto de circuitos cíclicos posibles en el grafo y que constituirán todas las *caras potenciales* del modelo. Entendiendo por caras potenciales los circuitos candidatos a ser caras del modelo.

Las caras potenciales se obtienen por combinación de circuitos fundamentales. Por lo que se denomina *espacio de circuitos* al conjunto de caras potenciales y *base del espacio* al conjunto de circuitos fundamentales que los genera. No obstante, debemos tener presente que no todas las combinaciones de circuitos fundamentales dan como resultado una cara potencial, por lo que son adoptados los siguientes criterios:

- Si como resultado de la combinación de dos o más circuitos fundamentales el circuito resultante tiene aristas que se interceptan entre sí, el circuito resultante es eliminado del conjunto de caras potenciales.
- Si como resultado de la combinación de dos o más circuitos fundamentales se obtiene más de un circuito, el circuito resultante es eliminado del conjunto de caras potenciales.

La combinación de circuitos fundamentales se realiza del siguiente modo:

- Dados dos circuitos fundamentales obtendremos el circuito resultante tomando las aristas del primer circuito y añadiendo las del segundo del tal modo que si alguna arista se repite en los dos se anula y no formaría parte del circuito resultante.
- En el caso de que se realice una combinación partiendo de más de dos circuitos, si una arista coincide “n” veces entre todos los circuitos, se distinguen dos casos: si “n” es un número par en el circuito resultante se anulará la arista, si es impar la arista formará parte del circuito resultante.

Para aclarar éstas reglas veamos un ejemplo ilustrativo partiendo de la figura 112.

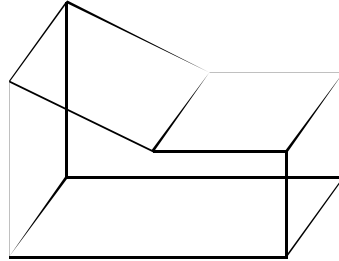


Figura 112. Figura Base

En la figura 113 se representan tres posibles circuitos fundamentales.

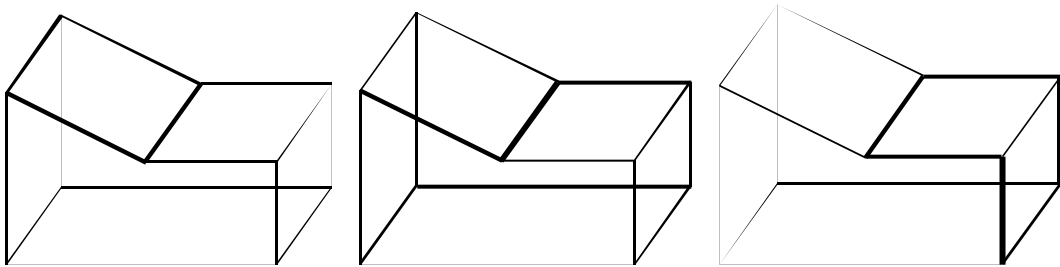


Figura 113. Circuitos fundamentales

el circuito resultante de combinar estos tres circuitos fundamentales (denotados por las aristas remarcadas en la figura) sería:

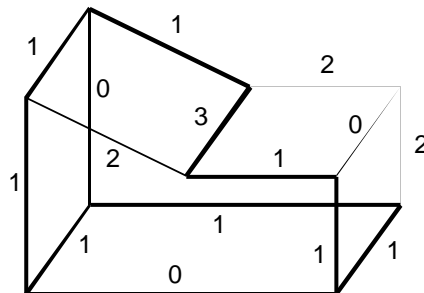


Figura 114. Combinación de circuitos fundamentales

Vemos que las aristas en que hay un número impar, que indica el número de aristas coincidentes de los distintos circuitos, forman parte del circuito resultante y el resto no.

6.2.4 Selección de las caras del modelo.

Hasta ahora hemos visto que en una proyección a partir de un punto de vista no accidental de un modelo tridimensional, las distintas caras planas del modelo siempre se corresponden con circuitos de aristas cíclicos que no se interceptan entre sí. Sin embargo, no todo circuito cíclico de aristas que no se intercepten entre sí se corresponde con caras planas del modelo tridimensional. De hecho, la gran mayoría de estos circuitos no corresponden a caras del modelo. Como ejemplo, en la figura 115 se representan los 15 circuitos cíclicos obtenidos como combinación de los circuitos fundamentales de la figura 111. Se observa que muchos no corresponden a caras del cubo que cualquier observador ve en la figura.

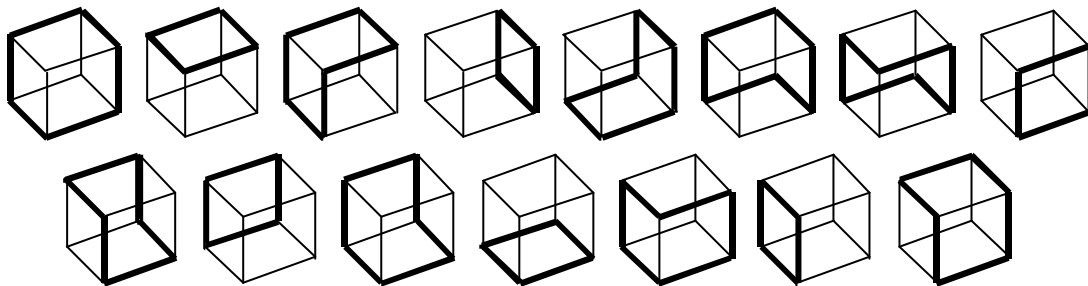


Figura 115. Circuitos de aristas cerrados y no interceptados entre sí en la proyección de un cubo

Por consiguiente el problema de identificación de caras de un objeto general puede ser definido en esta etapa como un problema de selección.

De forma genérica se puede establecer que el número “A” de aristas y el número “C” de caras se podrán expresar en función del número “V” de vértices. Y su proyección tendrá “m” circuitos no interceptados entre sí, siendo “m” una función exponencial de V. Del conjunto “m” de caras potenciales, únicamente un subconjunto “k” corresponderá con caras reales, siendo $0 \leq k \leq m$.

El método consiste básicamente en recorrer un árbol de búsqueda mientras se evalúan condiciones de compatibilidad entre las caras. Definido así el espacio del problema, deben ahora definirse las condiciones de compatibilidad y el dominio de la búsqueda para reducir el tiempo de computo.

- ***Ecuaciones de rango máximo.***

La ecuación de Euler no puede proporcionar ninguna información acerca del número de caras del modelo, en tanto que éste no tiene porqué ser necesariamente un poliedro cerrado, por lo que se precisa un método más complejo para determinar el número de caras. Para la formulación del método se establece la siguiente notación:

- $R(a)$ es el *rango de una arista* “a”, e indica el número de caras de las que forma parte la arista.
- $R(v)$ es el *rango de un vértice* “v”, e indica el número de caras de las que forma parte el vértice.
- $G(v)$ es el *grado de un vértice* “v”, e indica el número de aristas de las que forma parte el vértice.

A partir de ciertas condiciones que relacionan los tres parámetros descritos arriba, podemos acotar el valor máximo que pueden tener los parámetros, y, en consecuencia, podemos establecer un *sistema de ecuaciones de rango máximo* que permite determinarlos:

- Cada cara que contenga a un vértice “v”, debe contener también a dos aristas que concurran en “v”. De esta manera, el número de caras que contienen al vértice “v” está limitado por el número de permutaciones de pares de aristas que concurren en el vértice “v”.

$$R(v) \leq \frac{1}{2} [G(v) \cdot (G(v)-1)] \tag{52}$$

- De igual forma, cualquier cara que contenga a una arista “a” debe también contener a otra arista concurrente con “a”, por tanto, el máximo número de caras está limitado por el número de aristas que se unan en un vértice.

$$R(a) \leq G(v_1) - 1 \quad \text{y} \quad R(a) \leq G(v_2) - 1 \tag{53}$$

es decir:

$$R(a) \leq \min [G(v_1), G(v_2)] - 1 \quad (54)$$

siendo v_1 y v_2 los vértices de la arista “a”.

- Determinado $R(a)$ es evidente que cada cara que contenga a un vértice también contendrá a dos aristas que concurren en dicho vértice, y por tanto, para todas las aristas que se encuentre en un vértice “v” deberá verificarse:

$$R(v) = \frac{1}{2} \sum R(a) \quad (55)$$

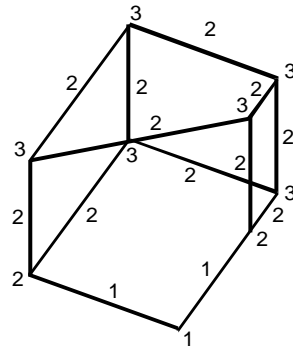
Similarmente, cada cara que contenga a una arista debe también pasar por los vértices de sus dos extremos y por consiguiente:

$$R(a) \leq \min [R(v_1), R(v_2)] \quad (56)$$

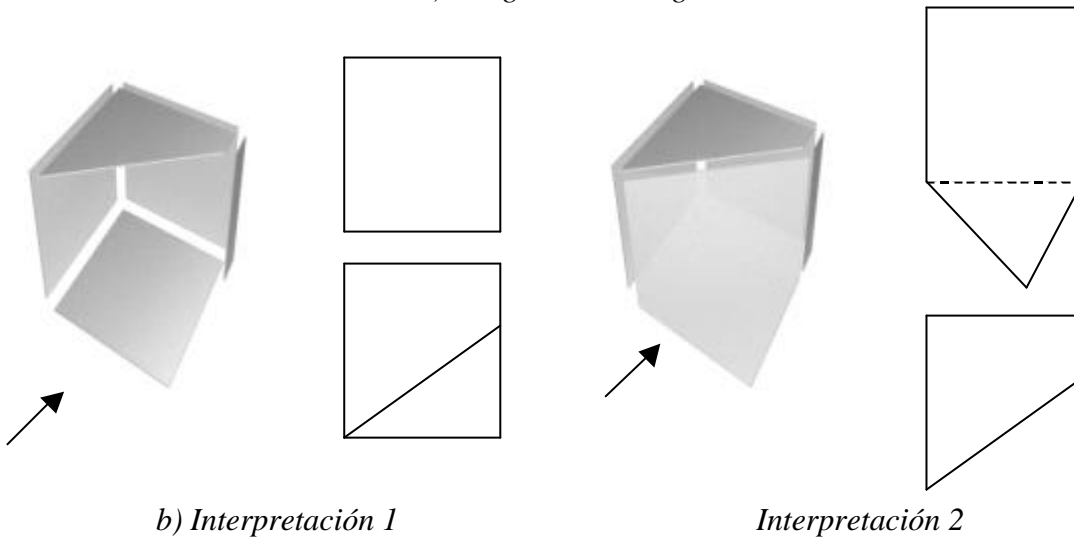
En la implementación práctica el algoritmo sigue las siguientes fases:

1. Se detecta el grado de cada vértice consultando en la base de datos del grafo el número enlaces que concurren en cada nodo.
2. Se computa una estimación preliminar de los rangos de vértices y aristas a partir de las expresiones dadas en (52) y (54).
3. Se aplica de manera iterativa las ecuaciones (55) y (56) hasta que se llega al equilibrio, es decir, hasta que los rangos de las aristas y vértices verifican las ecuaciones de rango máximo.

Como muestra de la aplicación de las ecuaciones de rango máximo, en la figura 116 se muestran los resultados de este método para un *modelo de calderería* (modelo “origami”, no sólido y de paredes de espesor despreciable). La figura 116a muestra los rangos máximos calculados en el grafo. Las figuras 116b y 116c muestran un conjunto de caras potenciales que concuerdan con estos rangos y que son seleccionadas como posibles soluciones al problema planteado.



a) Rangos de la imagen



b) Interpretación 1

Interpretación 2

Figura 116. Aplicación de la ecuaciones de rango máximo en un objeto de calderería

Para reducir el número de posibles soluciones al problema, dichos autores aplican una nueva estrategia que permite eliminar algunas de las posibles soluciones. Esta estrategia se establece a partir del teorema de la adyacencia:

“Si dos caras adyacentes tienen más de una aristas en común, ambas aristas deben ser colineales”

De esta forma se eliminan como posibles soluciones todas aquellas configuraciones que no verifiquen el teorema.

- ***Función de conformidad.***

Dado que las ecuaciones de rango máximo sólo establecen un límite superior de los rangos, estos permiten un gran número de soluciones, incluyendo la solución trivial de cero caras, que implica que el objeto es una estructura de esqueleto sin ninguna cara.

Esta solución es matemáticamente correcta pero no es probable que sea seleccionada por el observador. Para superar esta dificultad, se usa una regla heurística que intenta imitar el razonamiento psicológico de un observador humano. Según nuestras observaciones, un sujeto humano típicamente selecciona una configuración que verifique:

*“Tantos vértices como sean posibles
participarán en tantas caras como sean posibles”*

La función objetivo, entonces, es la que concuerda más estrechamente con los límites superiores derivados de las ecuaciones de rango máximo. Dado un subconjunto “x” de caras potenciales, y calculados los rangos máximos de vértices y aristas, se puede calcular una función de conformidad $g(x)$ minimizando la suma de las desviaciones de los rangos actuales a partir de los límites superiores de los rangos.

$$g(x) = \sum |R^+(a) - R(a)| + \sum |R^+(v) - R(v)| \quad (57)$$

donde la suma se realiza sobre todos los nodos y enlaces del grafo. La función de conformidad es baja para un buen ajuste y alta para un mal ajuste.

6.2.5 Selección de la solución final.

A pesar de la función de conformidad comentada con anterioridad, el algoritmo puede encontrar más de una combinación de caras potenciales con idéntico valor cuantitativo de la función. Es por ello que se precisa de algún otro método para la selección de la solución más plausible.

Esta determinación está basada en la percepción de las regularidades de la imagen. El concepto de regularidades de la imagen ya ha sido comentado con anterioridad y cuantifica el cumplimiento de ciertas relaciones geométricas entre

entidades en el plano de proyección y los modelos espaciales. En este trabajo, la ortogonalidad oblicua ha sido seleccionada como método para determinar el modelo más adecuado. Es decir, ante un conjunto de modelos con idénticos valores de la función de conformidad, se evalúa dicha regularidad para cada uno de los modelos y se selecciona aquel cuya evaluación sea mínima.

6.2.6 Reducción del espacio de circuitos.

El método descrito ha sido probado y resulta válido tanto para modelos cerrados como para modelos abiertos (o de calderería). Sin embargo, y a pesar de las limitaciones impuestas al despliegue del árbol de combinaciones dado por las ecuaciones de rango máximo y el teorema de la adyacencia, el proceso resultante sigue presentando un alto coste computacional. Por ello, en este apartado se describe una estrategia propuesta por Lipson y Sphitalni cuya finalidad es reducir el tamaño de caras potenciales.

Se denominan caras no plausibles a aquellas que es improbable que sean interpretadas como caras válidas aplicando criterios perceptuales.

El método propuesto busca en la lista de caras potenciales para encontrar grupos en los que una cara contiene completamente a otras caras adyacentes y comparta al menos una arista con cada una de las caras que contiene. En la figura 117 se muestra un grupo de caras potenciales en las cuales una de las caras (figura 117c) contiene completamente a otras dos adyacentes (figura 117a y figura 117b) compartiendo aristas con cada una de ellas.

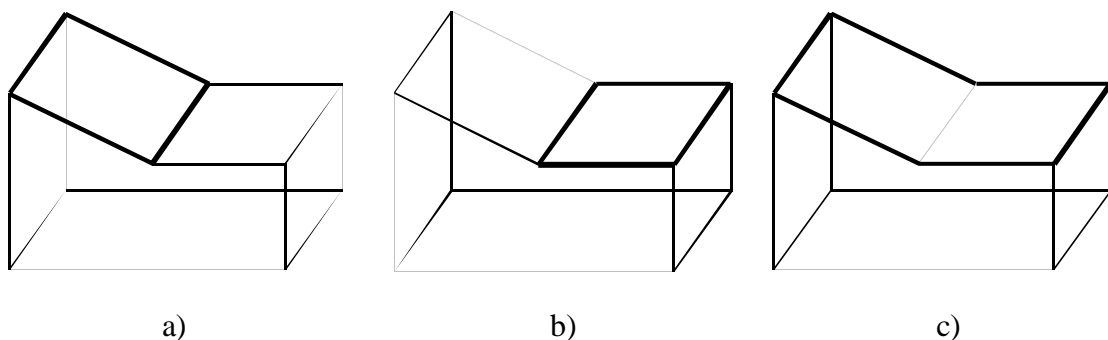


Figura 117. Eliminación de caras potenciales no plausibles

De acuerdo al teorema de adyacencia, la cara potencial que contiene a las otras dos no puede coexistir con ellas. Queda decidir si es la cara que contiene o las caras contenidas las que permanecerán en la lista de caras potenciales.

Seleccionar las dos caras que están contenidas puede ser más útil para reducir la función de conformidad que seleccionar la cara que las contiene. Las dos caras juntas aumentan los rangos de más vértices y aristas debido a que su arista común no es parte de la cara que contiene. Este análisis local hace posible determinar con anticipación que el proceso de búsqueda descartará la cara más grande, haciendo segura su omisión del grupo de caras potenciales, con anticipación.

6.3 Determinación de las aristas ocultas de un grafo geométrico.

Tal como se ha justificado al principio del capítulo, en los métodos de inflado por niveles se puede sacar provecho de la información de aristas ocultas. Para ello, se requiere distinguir entre imágenes de aristas vistas e imágenes de aristas ocultas. En la mayoría de casos esta información no está explícitamente recogida en la imagen de partida, por lo que debe ser obtenida a partir de la imagen 2D. No hemos encontrado ningún algoritmo con un propósito semejante (obtener aristas ocultas en una representación axonométrica de un modelo BRep del cual no se tiene información tridimensional), por lo que hemos desarrollado un algoritmo propio.

El algoritmo que hemos desarrollado para la detección automática de aristas ocultas en una imagen 2D, precisa del conocimiento de caras definidas en la imagen que ha sido tratado con antelación. El algoritmo evalúa la visibilidad de todas las aristas representadas en la imagen, por lo que se precisa de la representación de todas estas, sea cual sea su visibilidad.

La estrategia propuesta está basada en un conjunto reglas heurísticas que son aplicadas a las distintas configuraciones posibles del modelo (entendiendo como configuración del modelo, una combinación de aristas visibles y ocultas). La estrategia se divide en dos fases:

- Detección de aristas total o parcialmente ocultas.
- Segmentación de las aristas parcialmente ocultas.

6.3.1 Detección de aristas total o parcialmente ocultas.

En esta etapa se trata de identificar todas aquellas aristas de la imagen que bien parcialmente o en su totalidad resultan ocultas en el modelo. Se propone un método que realiza una búsqueda exhaustiva y selecciona aquellas configuraciones que verifican un conjunto de reglas heurísticas.

El mayor inconveniente que presenta es el elevado número de configuraciones posibles, dado que este crece de manera exponencial en la forma 2^N , siendo “N” el número de aristas del modelo, lo que en ocasiones provoca que el algoritmo resulte inviable.

Para paliar este problema se emplean dos estrategias:

- Reducción del número de aristas que conforman las posibles configuraciones del modelo.
 - Tratamiento agrupado de subconjuntos de aristas.
-
- ***Reducción del número de aristas que conforman las posibles configuraciones del modelo.***

En esta fase, que intenta reducir el número de aristas de modelo, se establece la visibilidad de ciertas aristas que dada su situación particular en la imagen permiten definir directamente si se tratará de una arista visible o parcialmente oculta. Los criterios establecidos para determinar la visibilidad de dichas aristas son los siguientes:

1. Se parte del convenio existente a la hora de interpretar una determinada proyección de considerar que el modelo está siendo visto desde arriba. En otras palabras, asumimos que el objeto está representado como sostenido en la palma de la mano, estando ésta a una altura inferior a la de nuestros ojos, lo que nos hace establecer como norma de partida el considerar el nodo de mayor coordenada “Y” del grafo, como nodo siempre visible del modelo. Es más, se considera que ese nodo pertenece a una cara visible: la “cara superior”. Por ello, obtenido dicho nodo se intenta identificar aquella cara superior visible del modelo que contenga a este nodo. Para ello se adopta el criterio de definir como “cara superior de la imagen” aquella cara definida por dos enlaces que, convergiendo en el nodo, abarquen entre sí en mayor ángulo posible.

En la figura 118 se muestran dos ejemplos en la que se ha detectado su cara superior.

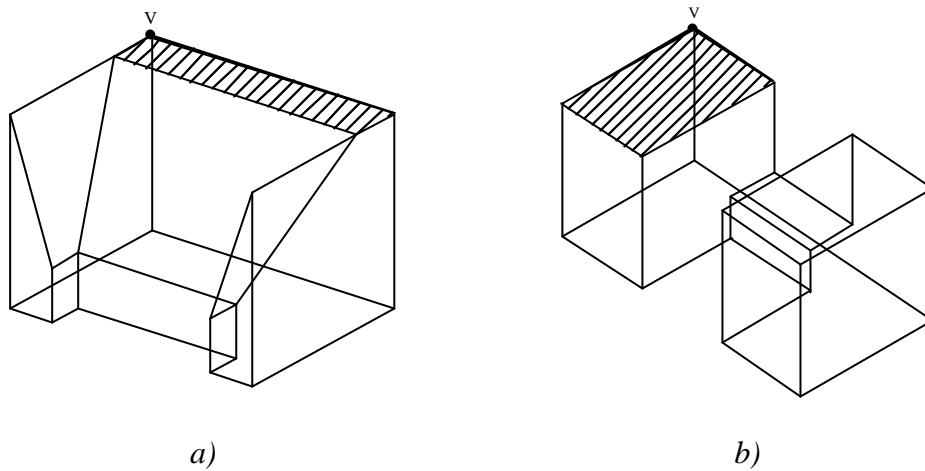


Figura 118. Definición de cara superior en una imagen

Detectada la cara superior de la imagen se analizan las aristas que definen dicha cara de acuerdo con los siguientes criterios:

- Si la arista no intercepta a ninguna otra arista del grafo, se considerará arista visible.
- Si la arista intercepta con otra arista del grafo que converge en un vértice de la cara defina como cara superior del modelo y se extiende hacia un vértice de menor coordenada “y”, la arista de la cara se considerará como visible y la arista interceptada como oculta. Además, cualquier otra arista del grafo que converja con la definida como oculta en un vértice que no pertenece a la cara superior, será también oculta.
- El resto de las aristas de la cara definida como cara superior de la imagen serán definidas como aristas ocultas.

De acuerdo con los criterios establecidos, las aristas de la cara superior de las imágenes de la figura 118 quedarían definidas como se muestran en la figura 119.

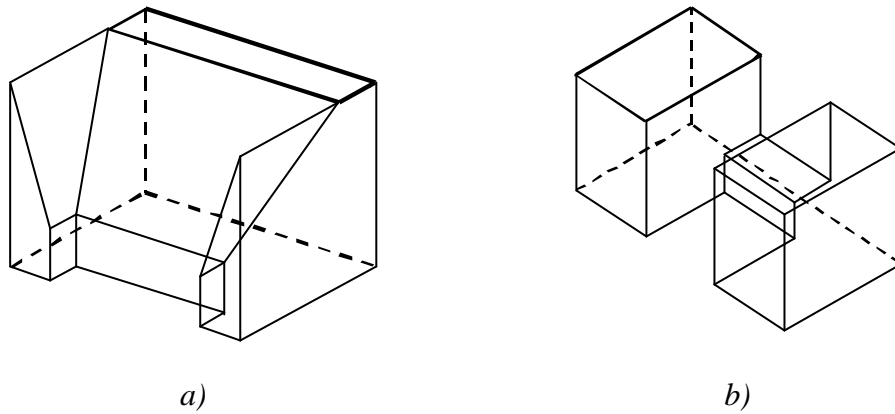


Figura 119. Estilos de las aristas de la cara superior de la imagen

2. Se establece como criterio que cualquier arista perteneciente al contorno aparente de la imagen y que no intercepte con otra arista del contorno aparente, será definida como arista visible.

Si dos aristas del contorno aparente se interceptan entre sí se determina si en los vértices que definen dichas aristas convergen otras aristas del contorno aparente de modo que:

- Si esta propiedad sólo se verifica en una de las aristas que se interceptan, la arista que verifica dicha propiedad será visible y la arista que intercepta oculta.
- Si esta propiedad no se verifica en ninguna de las aristas o si se verifica en ambas, se analiza el etiquetado de cada uno de los vértices de las aristas en su extremo conflictivo (extremo al que no llega ninguna otra arista del contorno aparente), de manera que aquella arista que tenga un etiquetado superior será visible y la otra oculta.

El algoritmo que identifica las aristas de la imagen que forman parte del contorno aparente será descrito posteriormente.

Como puede verse en los resultados mostrados en la figura 120a, las aristas 1-2 y 3-4 se interceptan entre sí y a ninguno de los vértices conflictivos (2 y 4) llegan otras aristas del contorno aparente, por lo que la definición de niveles es el criterio adoptado para decidir la visibilidad de las aristas. En la figura 120b las aristas 1-2 y 3-4 también se interceptan entre sí, pero en el vértice 4 concurre otra arista del contorno aparente (4-5), por lo que la arista 3-4 será definida como visible y la arista 1-2 como oculta.

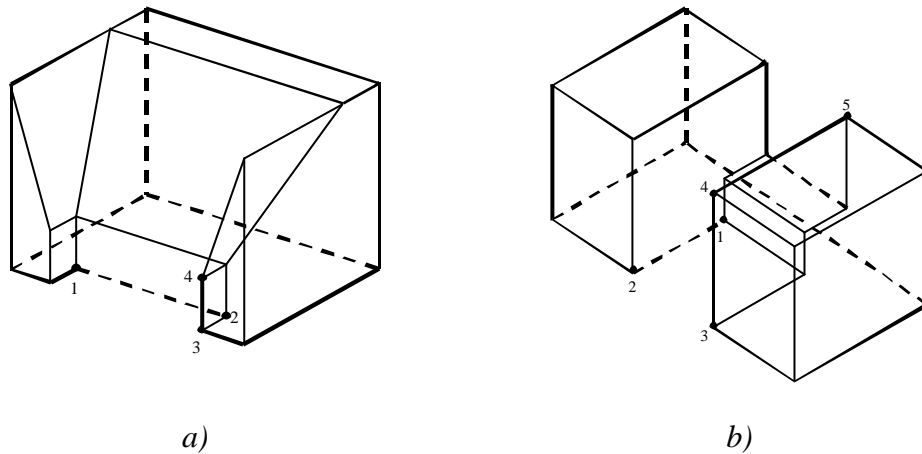


Figura 120. Estilos de las aristas del contorno aparente

3. Se examinan las caras definidas en la imagen de manera que, si una cara tiene definidas como visibles al menos dos de sus aristas y el resto de sus aristas no están definidas como ocultas, si las aristas aún sin definir no se interceptan a ninguna otra arista del grafo, o sólo lo hacen con aristas definidas como ocultas, toda la cara se considerará visible y por tanto todas sus aristas serán también visibles (figura 121).

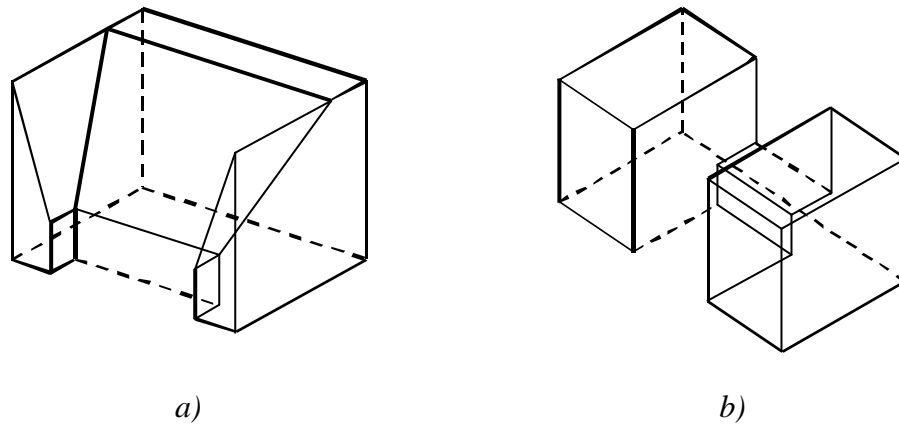


Figura 121. Definición de caras visibles del modelo

Mediante la aplicación de estos criterios, se ha conseguido reducir notablemente el número de aristas del modelo pendientes de analizar y por consiguiente el número posible de configuraciones. En el modelo analizado en la figura 121a se ha reducido el

número de aristas a computar de 29 (536.870.912 posibles combinaciones) a 8 (256 combinaciones). El modelo de la figura 121b ha reducido el número de aristas a computar de 30 (1.073.741.824 posibles combinaciones) a 11 (2.048 posibles combinaciones).

- **Tratamiento agrupado de subconjuntos de aristas.**

Mediante este método se pretenden modificar los estilos de aristas de manera agrupada identificando las caras a las que pertenecen. De esta manera se reduce el número de posibles combinaciones a 2^C , siendo “C” el número de caras que es notablemente inferior al número de aristas.

Se establece la siguiente terminología:

- Caras visibles: aquellas caras limitadas por aristas visibles.
- Caras ocultas: aquellas caras limitadas por aristas en las que al menos una de ellas es oculta.

Se establecen las distintas configuraciones de la imagen como una combinación de caras visibles (V) y ocultas (O) (tabla 122).

Caras	Permutaciones												
	P ₁	P ₂	P ₃	P ₄	P ₅	P ₆	P _{n-5}	P _{n-4}	P _{n-3}	P _{n-2}	P _{n-1}	P _n
C ₁	O	O	O	O	O	O	O	O	O	O	O	O
C ₂	O	O	O	O	O	O	O	O	O	O	O	V
C ₃	O	O	O	O	O	O	O	O	O	O	V	O
.....
C _{n-2}	V	V	V	V	V	V	V	V	V	V	O	V
C _{n-1}	V	V	V	V	V	V	V	V	V	V	V	O
C _n	V	V	V	V	V	V	V	V	V	V	V	V

Tabla 122. Permutaciones para modelo de "n" caras

Para cada una de las posibles configuraciones se definen como visibles todas las aristas que pertenezcan a alguna cara visibles. El resto de las aristas se definen como ocultas.

Así, para un modelo de cuatro caras, se definen las 16 configuraciones que se muestran en la tabla 123.

Visibilidad de caras	Modelo resultante	Visibilidad de caras	Modelo resultante	Visibilidad de caras	Modelo resultante
a)		b)		c)	
d)		e)		f)	
g)		h)		i)	
j)		k)		l)	
m)		n)		p)	
q)					

Tabla 123. Permutaciones en un modelo de cuatro caras

Como se muestra en la tabla 123, las 16 configuraciones posibles dan como resultado algunos modelos comunes que deben ser identificados y eliminados para evitar una posterior evaluación que resultaría redundante. Así, para el modelo que estamos analizando únicamente serán evaluadas las 12 configuraciones que se muestran en la tabla 124 y que constituyen los distintos modelos conseguidos por la combinación de caras visibles y caras ocultas:

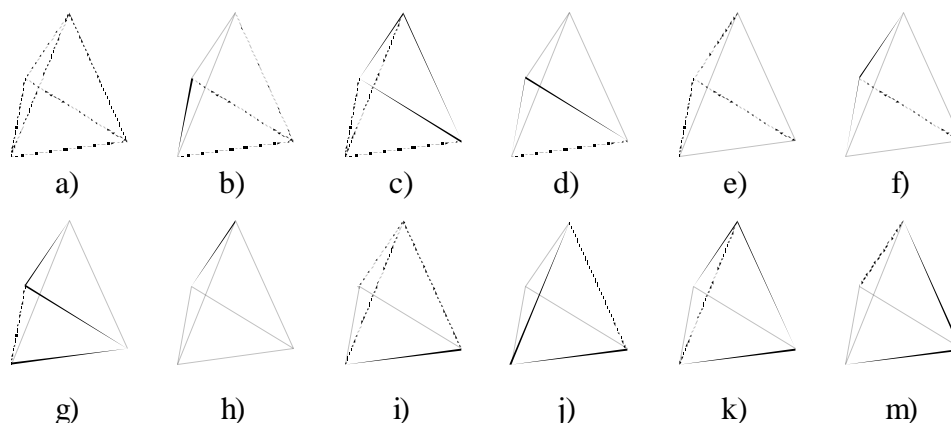


Tabla 124. Configuraciones a evaluar del modelo de cuatro caras

Eliminadas las configuraciones redundantes, debe establecerse un proceso para seleccionar las soluciones plausibles. En nuestro caso, el criterio se ha establecido a partir de un conjunto de reglas heurísticas de modo que si alguna de las configuraciones verifica el conjunto de reglas propuestas dicha configuración es considerada válida.

El conjunto de reglas heurísticas aplicadas difiere según el método de análisis seleccionado.

- Para el caso de reducción de aristas de la imagen se aplicarán los siguientes criterios:
 1. En el modelo definido ningún par de aristas visibles deben interceptarse entre sí.
 2. Cualquier vértice en el que todas las aristas que concurran sean ocultas debe ser interior a una cada visible
 3. Si en el modelo definido existen aristas ocultas, estas deben verificar alguna de las siguientes condiciones:
 - En uno de sus extremos todas las aristas son ocultas.

- En sus dos vértices deben concurrir al menos otra arista oculta o cortar a dos aristas visibles de una misma cara.
4. En el modelo definido, si una cara tiene definidas todas sus aristas como visibles, dicha cara no deben contener vértices en los que concurren aristas visibles.
-
- Para el caso de tratamiento agrupado de subconjuntos de aristas se aplicarán los siguientes criterios:
 1. Ninguna arista del contorno aparente de la imagen debe ser definida como oculta.
 2. Ningún par de aristas definidas visibles deben interceptarse entre sí.
 3. Ninguna cara definida visible debe contener en su interior vértices en los que converja alguna arista visible.
 4. Si una arista oculta no intercepta a ninguna arista visible, todas las aristas que concurren en uno de sus vértices deben ser ocultas.

Aplicando las reglas anteriormente comentadas a las figuras de la tabla 124 se obtendrían los siguientes resultados:

- Mediante la primera de las reglas, los modelos: a, b, c, d, e, g, i, j, y m, son rechazados al no verificarse la propiedad del contorno aparente.
- La segunda de las reglas establece que dos aristas visibles nunca pueden interceptarse, por lo que el modelo “h” es rechazo.
- Por último la tercera regla no produce variación alguna en los resultados obtenidos, en tanto que los modelos “f” y “k” verifican la condición que esta regla impone, dándose ambos como soluciones válidas al problema.

6.3.2 Segmentación de las aristas parcialmente ocultas.

El proceso de segmentación de aristas parcialmente ocultas permite determinar aquellas porciones de las aristas que realmente resultan ser ocultas. El proceso ejecuta una segmentación de todas aquellas aristas definidas como parcialmente oculta en la fase anterior del algoritmo, en tantos segmentos como aristas visibles intercepte y analiza la visibilidad de cada uno de los segmentos en los que ha sido dividida la arista.

Si la arista no intercepta a ninguna arista definida en el grafo como visible, toda la arista se considera oculta.

La visibilidad de cada segmento se establece según las siguientes condiciones:

- Si el segmento analizado tiene un vértice común con la arista inicialmente definida como parcialmente oculta, en el cual todas las aristas que convergen han sido definidas como parcialmente ocultas, dicho segmento se considera oculto.
- Cuando no se verifica la condición anterior, se comprueba si el punto medio del segmento es interior a una cara que tenga definidas todas sus aristas como visibles. Si es así el segmento se considera como oculto, en caso contrario será considerado como visible.

Así, para el ejemplo de la figura 125a obtenido tras aplicar la primera fase del algoritmo, son evaluadas las aristas 1-18, 4-18, 17-18, 6-15, 11-15 y 14-15. Aristas que estando definidas como parcialmente ocultas interceptan aristas del grafo que han sido definidas previamente como visibles.

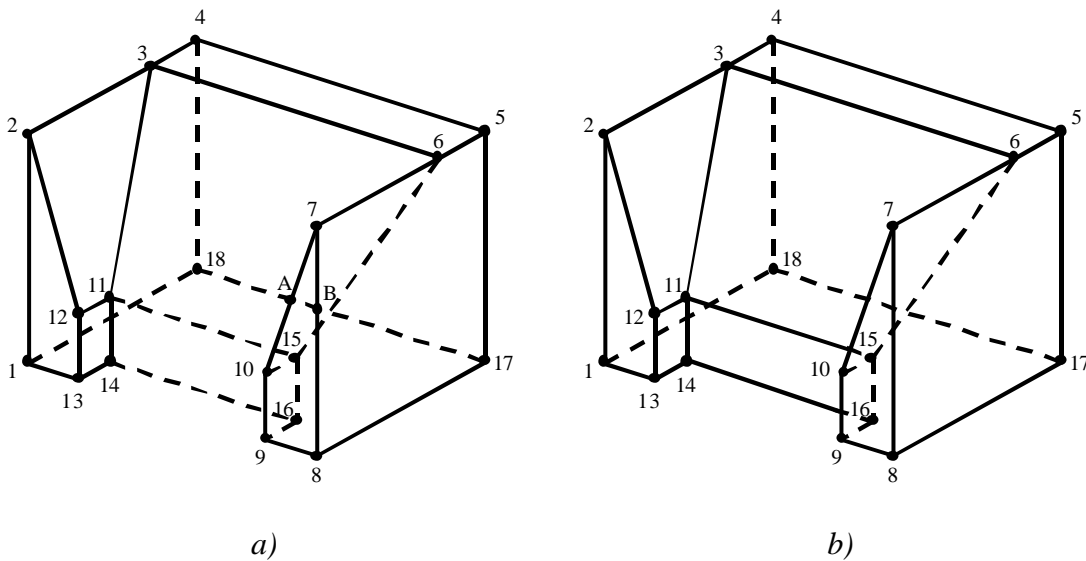


Figura 125. Segmentación de aristas parcialmente ocultas

Para ilustrar más claramente las condiciones anteriormente expuestas, analizaremos la segmentación de la arista 18-17. Dicha arista define parcialmente

oculta, es dividida en tres segmentos definidos por 18-A, A-B, B-17, como consecuencia de su intersección con aristas definidas como visibles (figura 125a).

Dichos segmentos son definidos todos ellos como ocultos, en tanto que verifican las siguientes condiciones:

- El segmento 18-A tiene un vértice (18) común con la arista inicialmente definida como parcialmente oculta (18-17) en el que todas las aristas que concurren son parcialmente ocultas.
- El segmento A-B es oculto en tanto que su punto medio es interior a una cara cuyas aristas han sido definidas como visibles (7-8, 8-9, 9-10, 10-7).
- El segmento B-17 es oculto en tanto que su punto medio es interior a una cara cuyas aristas han sido definidas como visibles (5-6, 6-7, 7-8, 8-17, 17-5).

Los resultados de la segmentación de las aristas parcialmente ocultas del modelo se muestran en la figura 125b.

El proceso de segmentación presenta otro aspecto importante además de la definición de visibilidad de cada arista. Este aspecto está unido a la selección de soluciones plausibles en la detección de aristas ocultas. Se expresa como una nueva regla heurística a verificar por aquellas imágenes que se considerarán como válidas:

Realizada la segmentación de aristas parcialmente ocultas, ningún segmento definido como visible debe quedar colgante en el modelo, es decir, que cualquier segmento definido visible ha de compartir sus vértices con otros segmentos o aristas también visibles.

Si esta regla no se verifica, el modelo es excluido como solución válida.

6.3.3 Determinación de las aristas del contorno aparente.

Para la obtención de aristas ocultas de una imagen, se precisa como hemos visto determinar las aristas que forman el contorno aparente de la imagen. A continuación se expone la técnica que se ha empleado para la obtención de tal contorno aparente.

La técnica empleada para detectar las aristas del contorno aparente de una imagen proyección axonométrica de un sólido, es análoga a los algoritmos de línea de barrido para la determinación de superficies visibles, desarrollados originariamente por Wylie, Romney, Evans y Erdahl y recogidos en [Fole96].

Para su determinación se precisa de: la lista de caras a las que pertenece cada arista de la imagen (figura 126), una lista auxiliar que nos indica las caras en cuyo interior nos encontramos cada vez que analizamos una arista, y una *bandera booleana*, con valor inicial falso, que nos informa si nos encontramos en el interior del modelo. El valor de la bandera booleana deberá ser falso cuando la lista auxiliar de caras esté vacía y verdadero cuando la lista auxiliar de caras contenga algún elemento.

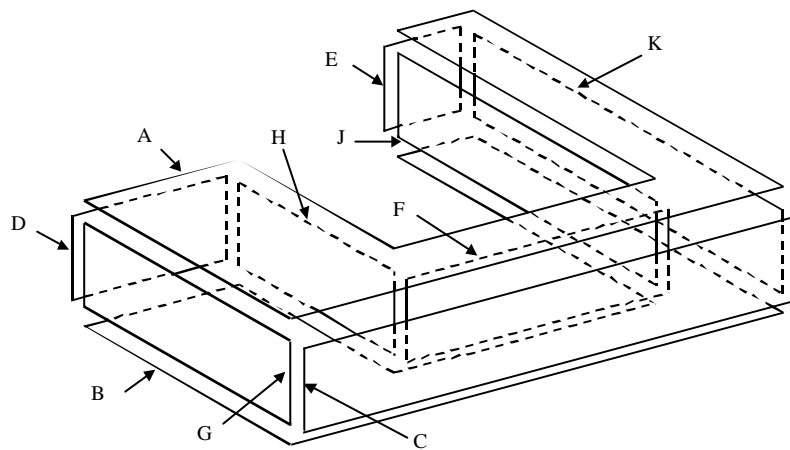


Figura 126. Representación en “explosión” de las caras identificadas en una imagen 2D de un poliedro

El método consiste en realizar una serie de barridos horizontales detectando de manera ordenada las intersecciones con cada una de las aristas que forman la imagen. Es decir, se trazan aristas horizontales sobre la imagen y se determinan los puntos de intersección con cada uno de los enlaces de grafo. Para cada arista interceptada se comprueba:

- Si la bandera booleana tiene valor falso, la arista se identifica como perteneciente al contorno aparente.
- Se comprueba si existen en la lista auxiliar las caras que contienen a la arista interceptada. Si existen, se eliminan de la lista auxiliar, si no existe se añade a la lista auxiliar.

- Se comprueba el valor de la bandera booleana. Si su valor es falso, se identifica la arista interceptada como arista de contorno aparente.

Como se muestra en la figura 127 los saltos entre líneas consecutivas de barrido vienen determinados por la coordenada y de la recta de barrido. La ordenada de los barridos horizontales viene dada en función de la ordenada de los distintos vértices de la imagen. En otras palabras, no se utiliza un salto uniforme, sino que se barre inmediatamente por encima e inmediatamente por debajo de cada vértice de la imagen. Para ello se clasifican en orden ascendente las distintas ordenadas de la imagen ($Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7, Y_8, Y_9, Y_{10}, Y_{11}, Y_{13}, Y_{14}, Y_{15}, Y_{16}$) y se hacen barridos entre cada dos ordenadas consecutivas y distintas entre sí: $L_1=Y_1+\xi, L_2=Y_2+\xi, L_3=Y_3+\xi, L_4=Y_4+\xi, L_5=Y_5+\xi, L_6=Y_6+\xi, L_7=Y_7+\xi, L_8=Y_8+\xi, L_9=Y_9+\xi, L_{10}=Y_{10}+\xi, L_{11}=Y_{11}+\xi, L_{12}=Y_{13}+\xi, L_{13}=Y_{14}+\xi, L_{14}=Y_{15}+\xi$, siendo ξ un valor determinado que asegura la traza del rayo entre dos ordenadas consecutivas de vértices de la imagen. En el ejemplo se puede observar que la ordenada Y_{12} no se ha considerado, dado que su valor es igual al de la ordenada Y_7 .

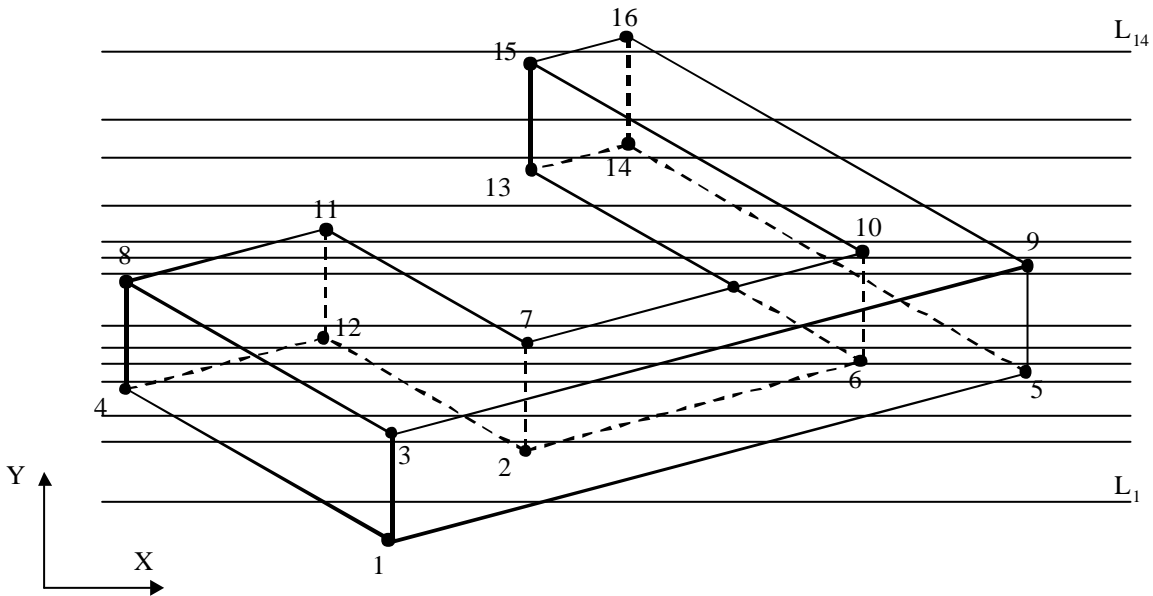


Figura 127. Barridos horizontales en una imagen 2D de un poliedro

En la tabla 128 se representa el proceso para el modelo de la figura 127.

Barrido	Bandera	Arista	Contorno	Aux	Barrido	Bandera	Arista	Contorno	Aux	
L ₁	F	1-4	1-4	B,G		V	7-11	7-11	-	
	V	1-3		B,C		F	7-10	7-10	A, F	
	V	1-5	1-5	-		V	3-9		F, C	
L ₂	F	1-4		B, G		V	6-13		F, C, B, J	
	V	1-3		B, C		V	6-10		C, B	
	V	2-12		C, H		V	5-14		C, K	
	V	2-7		C, F		V	5-9		-	
	V	2-6		C, B		L ₈	F	8-11	8-11	A, D
	V	1-5		-		V	11-12		A, H	
L ₃	F	1-4		B, G		V	7-11		-	
	V	3-8		B, A	F	6-13	6-13	B, J		
	V	3-9		B, C	V	7-10		B, J, A, F		
	V	2-12		C, H	V	5-14		J, A, F, K		
	V	2-7		C; F	V	6-10		A, K		
	V	2-6		C, B	V	3-9		K, C		
	V	1-5		-	V	5-9		-		
L ₄	F	4-8	4-8	D, G	L ₉	F	8-11		A, D	
	V	4-12		G, B	V	11-12		A, H		
	V	3-8		B, A	V	7-11		-		
	V	2-12		A, H	F	6-13		B, J		
	V	2-7		A, F	V	5-14		J, K		
	V	3-9		F, C	V	7-10		J, K, A, F		
	V	2-6		C, B	V	6-10		K, A		
	V	1-5		-	L ₁₀	F	8-11		A, D	
L ₅	F	4-8		D, G	V	11-12		A, H		
	V	4-12		G, B	V	7-11		-		
	V	3-8		B, A	F	6-13		B, J		
	V	2-12		A, H	V	5-14		J, K		
	V	2-7		A, F	V	10-15		K, A		
	V	3-9		F, C	V	9-16		-		
	V	2-6		C, B	L ₁₁	F	6-13		B, J	
	V	5-14		C, K	V	5-14		J, K		
	V	5-9	5-9	-	V	10-15		K, A		
L ₆	F	4-8		D, G	V	9-16		-		
	V	3-8		D, A	L ₁₂	F	13-15	13-15	J, E	
	V	4-12		A, B	V	13-14		J, B		
	V	2-12		A, H	V	5-14		J, K		
	V	2-7		A, F	V	10-15		K, A		
	V	3-9		F, C	V	9-16		-		
	V	6-13		F, C, J, B	L ₁₃	F	13-15		J, E	
	V	6-10		C, B	V	14-16		J, K		
	V	5-14		C, K	V	10-15		K, A		
	V	5-9		-	V	9-16		-		
L ₇	F				L ₁₄	F	15-16	15-16	A, E	
	F	4-8		D, G	V	14-16		A, K		
	V	3-8		D, A	V	9-16		-		
	V	11-12		A, H						

Tabla 128. Obtención de aristas pertenecientes al contorno aparente

Como resultado del proceso anteriormente descrito se obtienen las aristas de contorno aparente 1-4, 1-5, 4-8, 5-9, 7-11, 7-10, 8-11, 6-13, 9-16, 13-15 y 15-16.

El proceso descrito no obstante puede producir errores en su ejecución y no detectar ciertas aristas del contorno aparente en imágenes que verifiquen que dichas aristas se intercepten entre sí. Por consiguiente a la lista de ordenadas de vértices, será preciso añadir los puntos de intersección entre aristas para evitar la posible pérdida de detección de aristas del contorno aparente en casos como el que se muestra en la figura 129.

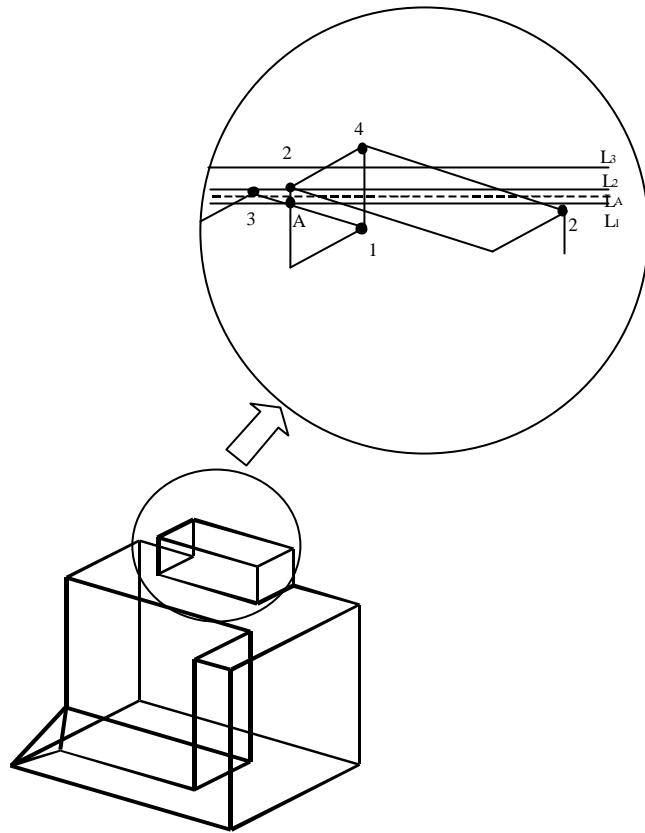


Figura 129 Adición de barridos horizontales

Si examinamos los tres últimos barridos horizontales L_1 , L_2 , L_3 que se ejecutarían en la imagen de la figura 129, como consecuencia de los vértices 1, 2, 3, la arista definida por 1,3 no sería detectada como perteneciente al contorno aparente, dado que el punto interceptado por el barrido horizontal L_1 resulta interior a una cara. Ese accidente provocado por la intersección de dos aristas del contorno aparente se ha remediado añadiendo barridos horizontales adicionales tales como L_A cuya inclusión surge de la consideración del vértice "A".

Por tanto el procedimiento utilizado ha sido el realizar tantos barridos como nodos tiene el gráfico, además de las intersecciones producidas entre los enlaces. Es decir, las líneas de barrido se generan a partir de los nodos y las intersecciones entre enlaces del grafo.

Los algoritmos propuestos en esta tesis han sido ensayados y validados en REFER mediante la ejecución de 138 ejemplos de modelos poliédricos eulerianos. En este capítulo se exponen los resultados y las conclusiones a las que hemos llegado para la reconstrucción de modelos de las diferentes tipologías definidas.

7.1 Algoritmos de optimización.

Como ya queda dicho, las técnicas de optimización son una buena alternativa para la Reconstrucción Geométrica. Sin embargo, se precisa que dichas técnicas sean capaces de conseguir el mínimo global de la función objetivo y descarten cualquier mínimo local. Algunos de los problemas particulares que se presentan en el caso de la reconstrucción son:

1. Algunas de las regularidades que componen la función a optimizar (planicidad de caras, paralelismo de aristas, isometría...) son trivialmente verificadas en la imagen por lo que su coste “cero” provoca que la imagen de partida sea un mínimo local.
2. La función objetivo, que como vimos viene expresada en términos de regularidades de la imagen, precisa de una adiestrada y correcta ponderación de regularidades que impiden la automatización del proceso. La ponderación debe hacerse en función de las características del modelo.

Sin embargo, ni los diferentes algoritmos trabajados en esta tesis ni los utilizados por autores anteriores, aseguran la consecución de mínimos globales para el problema particular de la reconstrucción. Por una parte Hill-Climbing, por su propia naturaleza de “descenso” le resulta muy fácil llegar a mínimos locales. Por otra parte los algoritmos Simulated Annealing si bien destacan por su técnica para escapar de mínimos locales, debe resaltarse que tienen una convergencia asintótica solamente aproximada, es decir, aunque el algoritmo es asintóticamente un algoritmo de optimización, cualquier implementación del mismo resulta en un algoritmo de aproximación. Es evidente que el

número de aproximaciones en cada escalón de temperatura sólo puede ser finito, y que la condición idónea de que $\lim T_k = 0$ cuando $K \rightarrow \infty$ sólo puede ser aproximada con un número finitos de valores de T. Por consiguiente y debido a estas aproximaciones, no se puede garantizar que el algoritmo converja al mínimo global con probabilidad 1.

Es por ello que los algoritmos de optimización deben ser asistidos por métodos auxiliares que ayuden a conseguir el mínimo deseado. Las técnicas que han sido propuestas en este trabajo consisten en realizar un inflado inicial para acercar al modelo hacia una configuración próxima al modelo final, de manera que el proceso de optimización parta de puntos iniciales tan próximos como sea posible al óptimo global. Es decir que las variables Z iniciales, se encuentren en un entorno próximo a su valor final.

7.2 Método de inflado previo axonométrico.

La aplicación del método de inflado axonométrico permite la reconstrucción directa de modelos normalones y cuasi-normalones, sin necesidad de procesos de optimización. De hecho, la aplicación de tales procesos después de varios ensayos con una variada combinación de regularidades, muestran que el proceso de optimización finaliza sin provocar importantes modificaciones en el modelo.

7.2.1 Modelos de tipología normalón.

De los 138 ejemplos poliédricos ensayados en REFER, 28 han sido detectados como normalones. En las tablas 130, 131 y 132 se muestran los ejemplos de tipología normalón ensayados por autores anteriores.

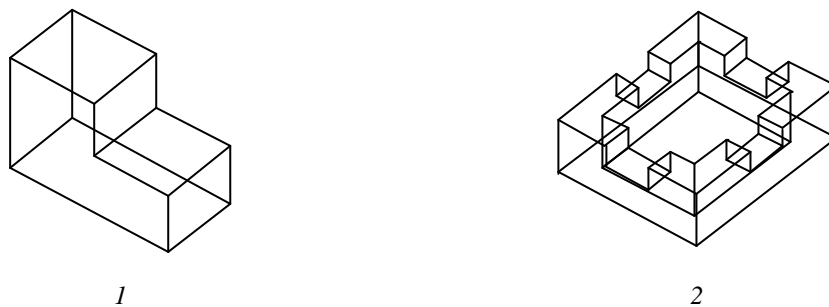


Tabla.130. Modelos de tipología normalón propuestos por Lamb y Bandopadhyay

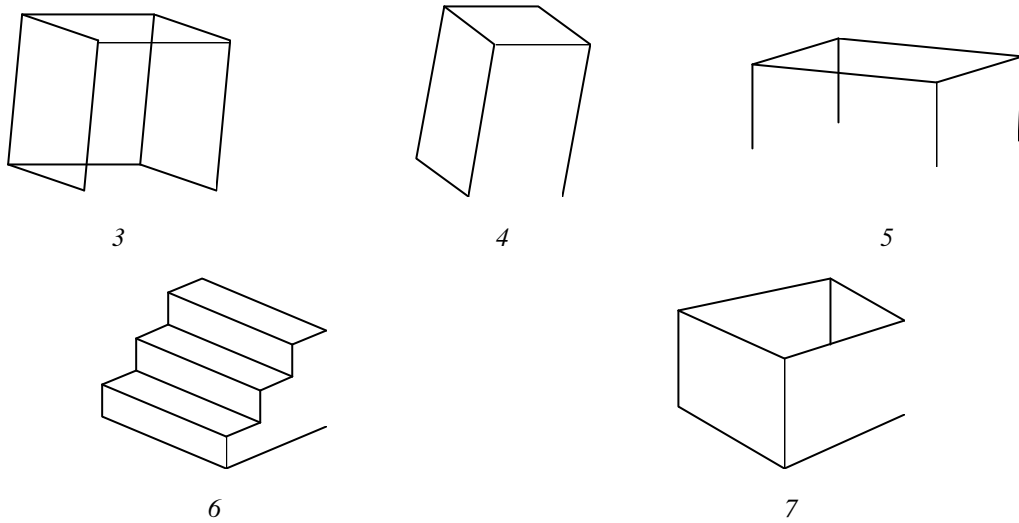


Tabla 131. Modelos de tipología normalón propuestos por Marill



Tabla 132. Modelos de tipología normalón propuestos por Lipson y Shpitalni

Todos ellos han sido reconstruidos con éxito a partir de la configuración por defecto de REFER, a los que detectando como modelos de tipología normalón, aplica un inflado previo axonométrico obteniendo un modelo válido sin necesidad de costosos procesos de optimización ni interacción alguna del usuario.

En la tabla 133 se muestran otros 19 ejemplos que han sido ensayados en este trabajo y que también han sido reconstruidos con éxito.

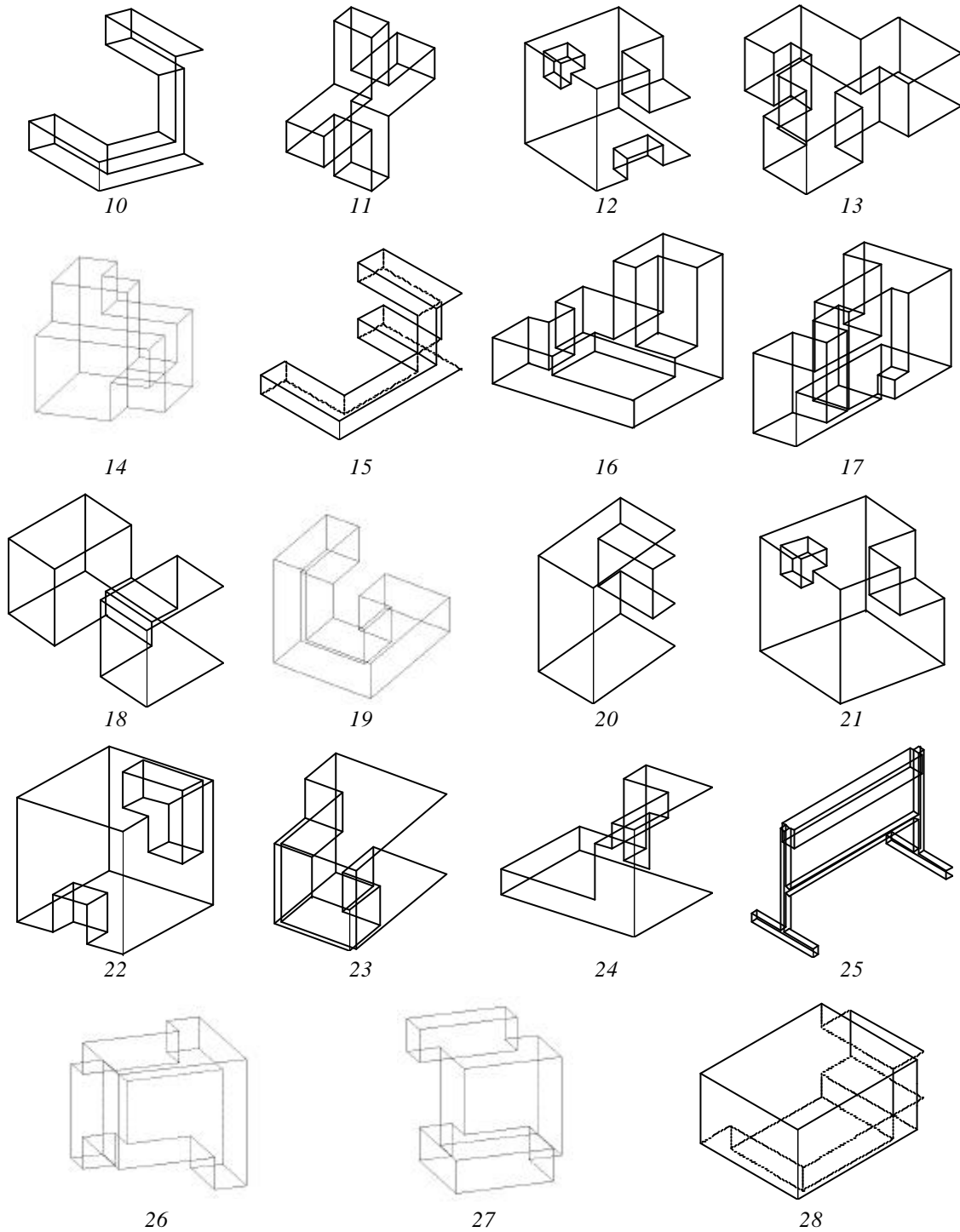
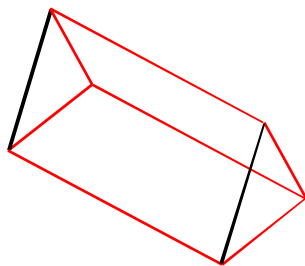


Tabla 133. Modelos de tipología normalón propuestos en esta tesis

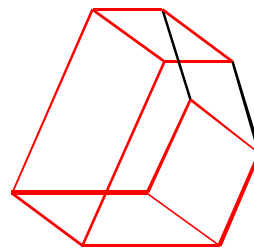
7.2.2 Modelos de tipología cuasi-normalón.

De los 138 ejemplos poliédricos ensayados en REFER, 54 han sido detectados como cuasi-normalones. Los enlaces rojos son considerados paralelos a tres direcciones principales permitiendo la evaluación del modelo. Los enlaces negros representan aristas del modelo que son eliminadas en la aplicación del método.

En las figuras 134 y 135 y en la tabla 136, se muestran los ejemplos de tipología cuasi-normalón ensayados por autores anteriores. Todos ellos se Reconstruyen con éxito mediante inflado previo axonométrico.



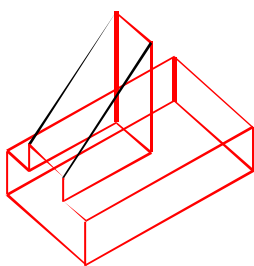
29



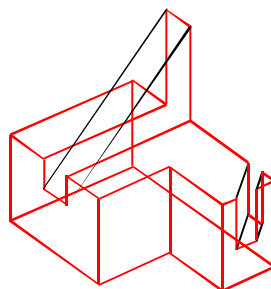
30

Figura 134. Modelo de tipología cuasi-normalón propuesto por Lamb y Bandopadhyay

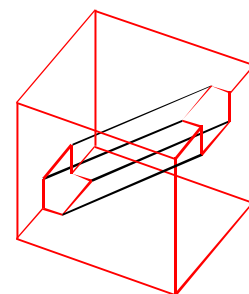
Figura 135. Modelo de tipología cuasi-normalón propuesto por Leclerc y Fischler



31



32



33

Tabla 136. Modelos de tipología cuasi-normalón propuestos por Lipson y Shpitalni

Las tablas 137a, 137b y 137c muestran otros 50 modelos propuestos para su ensayo en esta tesis, que REFER detecta de naturaleza cuasi-normalón y que han sido reconstruidos con éxito, validando el método propuesto.

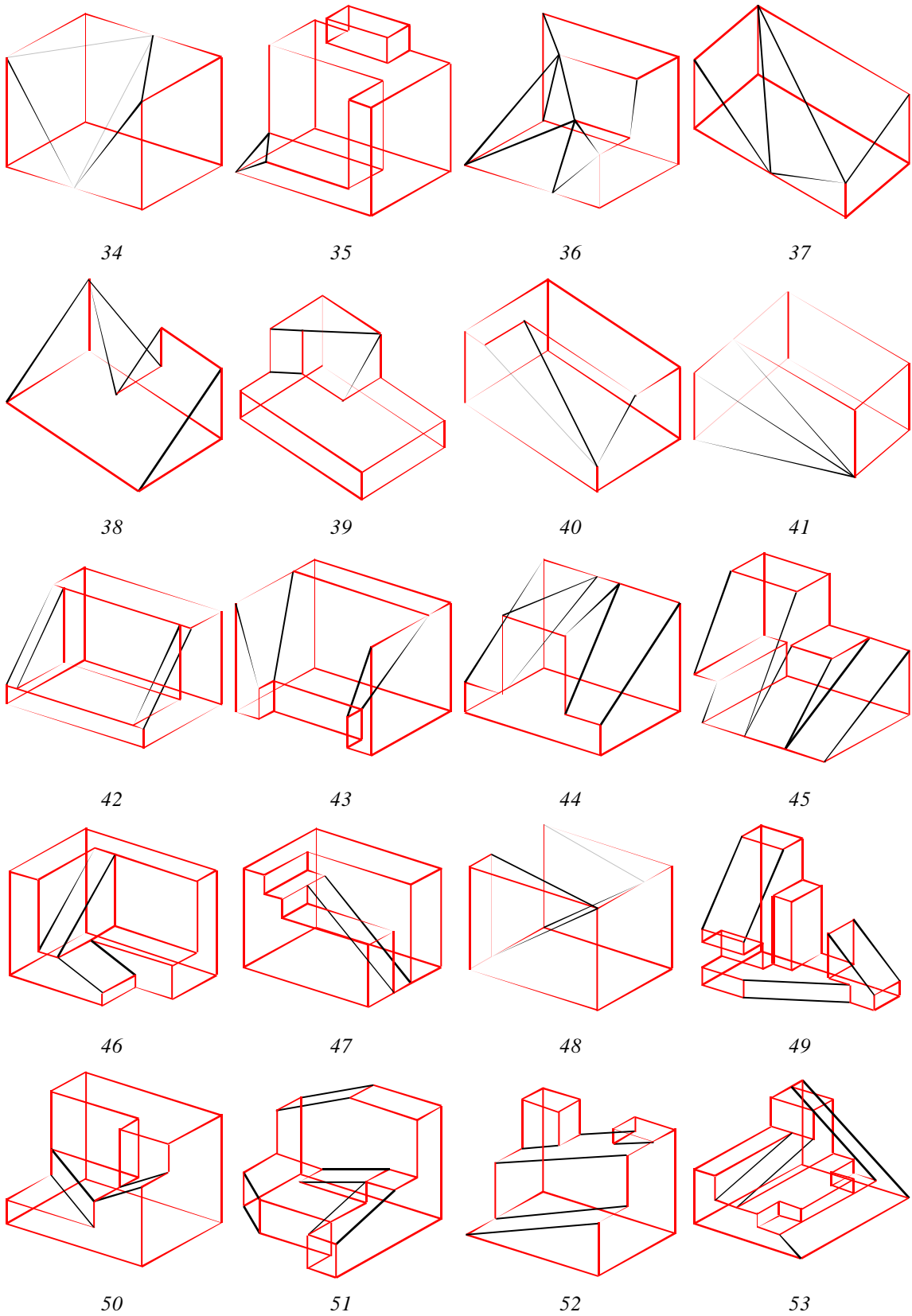


Tabla 137a. Modelos cuasi-normalones propuestos en esta tesis

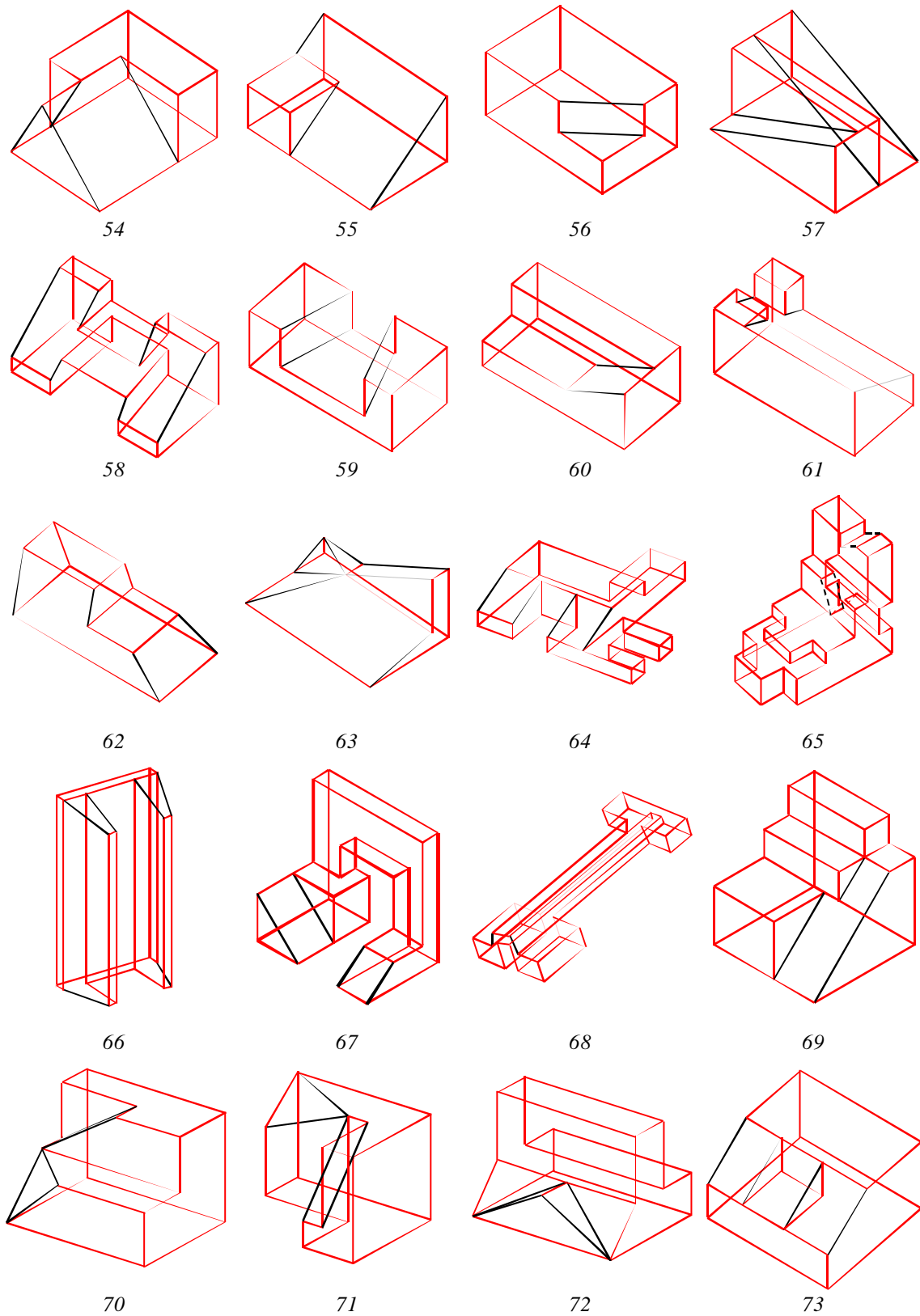


Tabla 137b. Modelos cuasi-normalones propuestos en esta tesis

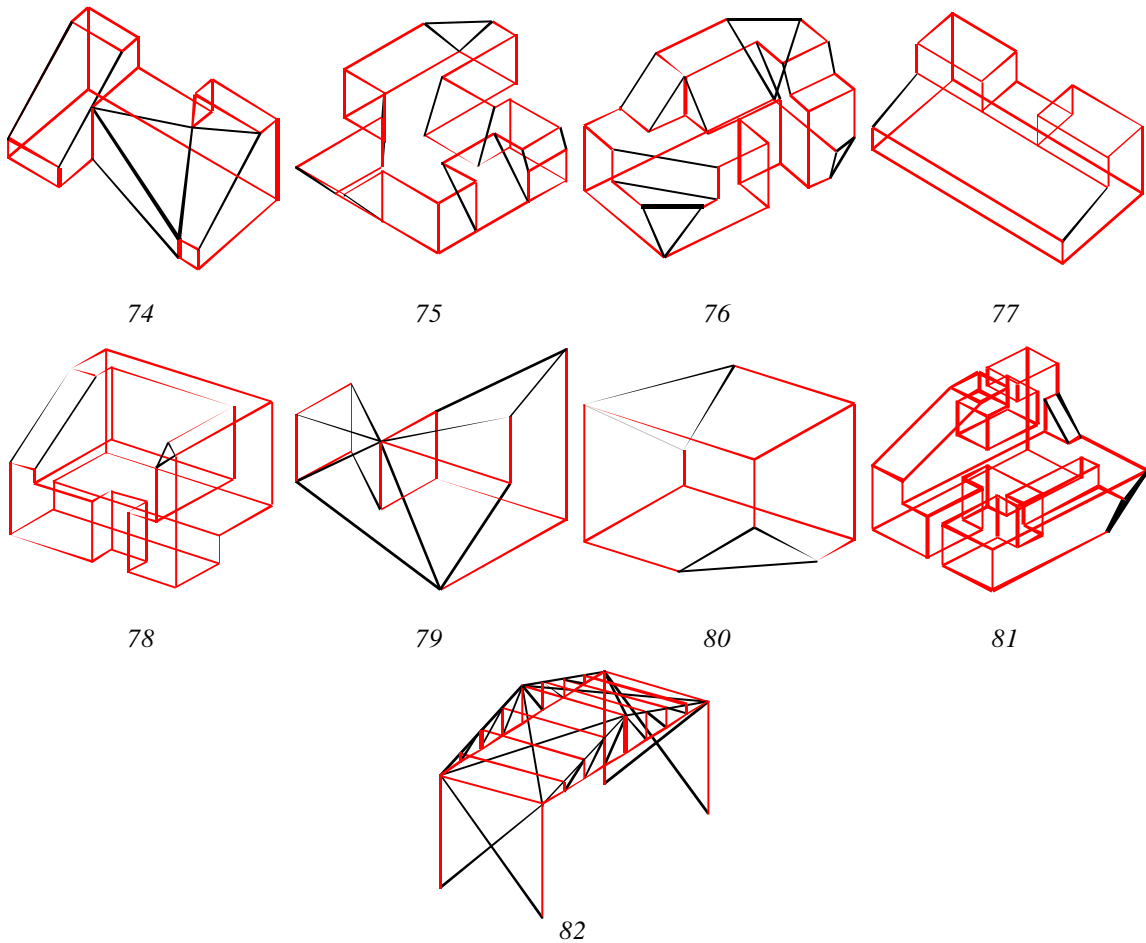


Tabla 137c. Modelos cuasi-normalones propuestos en esta tesis

7.2.3 Limitaciones del método.

El inflado previo axonométrico, que como hemos visto permite la reconstrucción de un elevado número de poliedros, presenta algunos problemas. Tres son las limitaciones que hemos podido encontrar.

La primera de ellas parte de la propia definición de la tipología de un modelo. Un modelo se supone de naturaleza cuasi-normalón si eliminando todas aquellas aristas que no son paralelas a tres direcciones principales, no queda aislado ningún nodo de la imagen. Las direcciones que consideramos como principales quedan definidas como aquellas que se repiten en la imagen con una mayor frecuencia, lo que hace que ciertos

modelos puedan ser erróneamente considerados de tipología cuasi-normalón, por haber seleccionado direcciones principales erróneas.

Dado que tales direcciones principales son consideradas para aplicar el criterio de perpendicularidad entre aristas, el inflado axonométrico daría como resultado un modelo no acorde con la percepción visual humana.

En la imagen de la figura 138a, se representa un modelo propuesto por Lipson y Shpitalni que pone de manifiesto esta limitación. Las direcciones d_1 , d_2 y d_3 son consideradas por nuestro algoritmo como direcciones principales, dado que la frecuencia con la que se repiten sus orientaciones es superior a la dada por la dirección d_4 .

Como consecuencia se obtiene el modelo Reconstruido de la figura 138b.

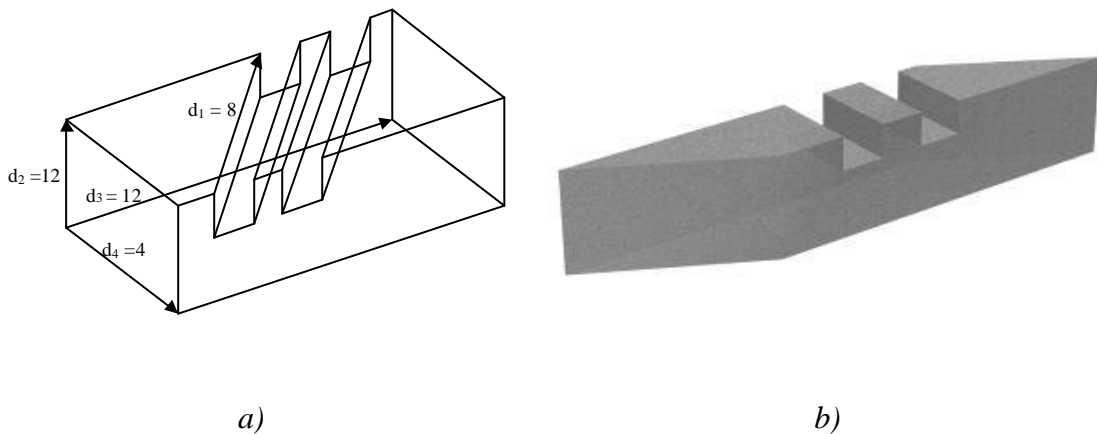


Figura 138. Error en la detección de direcciones principales

La segunda de sus limitaciones es aquella que obliga a que siempre que se detecten tres direcciones principales, éstas se hacen corresponder con aristas tridimensionales del modelo perpendiculares entre sí.

Una consecuencia directa de esta limitación puede verse en el modelo 29, propuesto en el trabajo de Lamb y Bandopadhyay y que nuestro algoritmo detecta como cuasi-normalón, donde el inflado previo axonométrico de la imagen (figura 139a) genera un modelo de aristas ortogonales (figura 139b).

No obstante esta limitación no es objeto del presente trabajo ya que nuestro objetivo se centra en conseguir una topología del modelo acorde con la imagen. Los ajustes finales del mismo tales como dimensiones lineales y angulares serán objeto de un proceso posterior en la etapa de refinado del modelo.

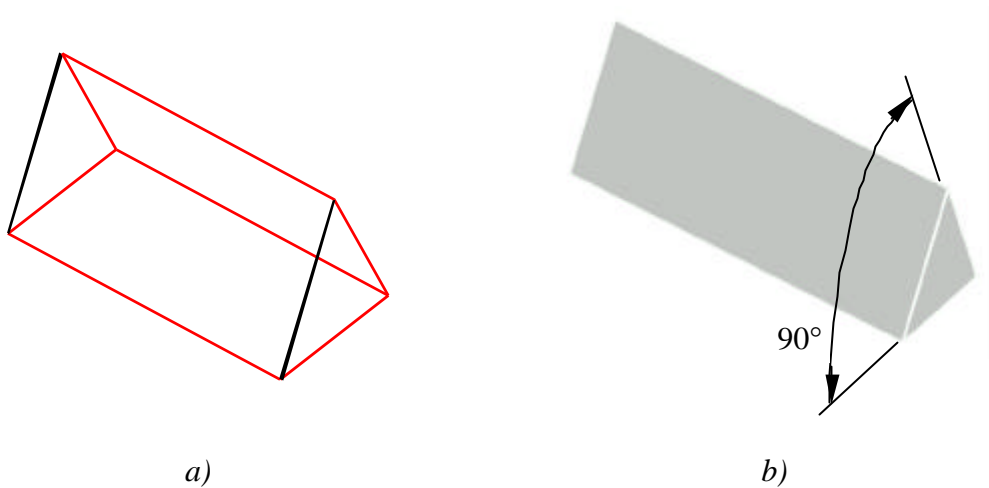


Figura 139. Efectos del criterio de perpendicularidad

La tercera y última limitación deriva de la propia expresión para obtener el ángulo que cada arista del modelo forma con el plano que contiene su proyección. Como se vio en el capítulo 5 dicha expresión viene dada en la forma:

$$\phi = \arcsen \sqrt{\cotg \alpha'_2 \cotg \beta'_2} \quad (45)$$

Del análisis del dominio de la función en el conjunto de los números reales, puede observarse que la función deja de existir para aquellos valores de α_2 y β_2 tales que verifiquen alguna de las siguientes condiciones:

$$(\cotg \alpha_2 * \cotg \beta_2)^{1/2} > 1 \Rightarrow \cotg \alpha_2 * \cotg \beta_2 > 1 \quad (58)$$

$$\cotg \alpha_2 * \cotg \beta_2 < 0 \quad (59)$$

De la expresión dada en (58) se deducen los siguientes valores de ángulos para los que la función no puede ser evaluada.

$$\cotg \alpha_2 * \cotg \beta_2 > 1 \Rightarrow \cotg \alpha_2 > \frac{1}{\cotg \beta_2} \Rightarrow \cotg \alpha_2 > \tg \beta_2 \quad (60)$$

es decir, si la cotangente de un ángulo es mayor que la tangente del otro ángulo, la expresión no tendrá valor en el campo real y por tanto el modelo no podrá ser reconstruido.

En la figura 140 se representan las funciones tg(a) y ctg(a) que nos indican las posibles combinaciones de ángulos en las que el inflado previo axonométrico no puede ser aplicado.

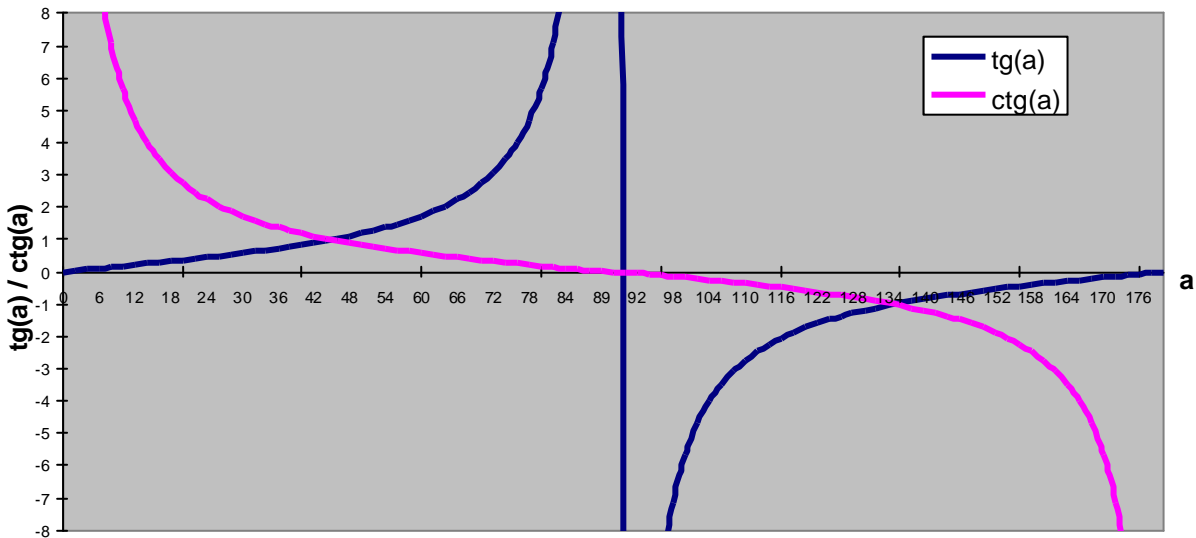


Figura 140. Combinaciones de ángulos en los que el inflado previo axonométrico no puede ser aplicado

De igual forma, la ecuación dada en (59) se verificará cuando se cumpla alguna de las siguientes condiciones:

$$\cotg \alpha_2 < 0 \Rightarrow \frac{\cos \alpha_2}{\text{sen } \alpha_2} < 0 \Rightarrow \begin{cases} \cos \alpha_2 > 0 \\ \text{sen } \alpha_2 < 0 \end{cases} \quad (61)$$

$$\begin{cases} \cos \alpha_2 < 0 \\ \text{sen } \alpha_2 > 0 \end{cases} \quad (62)$$

$$\cotg \beta_2 > 0 \Rightarrow \frac{\cos \beta_2}{\text{sen } \beta_2} > 0 \Rightarrow \begin{cases} \cos \beta_2 > 0 \\ \text{sen } \beta_2 > 0 \end{cases} \quad (63)$$

$$\begin{cases} \cos \beta_2 < 0 \\ \text{sen } \beta_2 < 0 \end{cases} \quad (64)$$

o bien:

$$\cotg \alpha_2 > 0 \Rightarrow \frac{\cos \alpha_2}{\text{sen } \alpha_2} > 0 \Rightarrow \begin{cases} \cos \alpha_2 > 0 \\ \text{sen } \alpha_2 > 0 \end{cases} \quad (65)$$

$$\begin{cases} \cos \alpha_2 < 0 \\ \text{sen } \alpha_2 < 0 \end{cases} \quad (66)$$

$$\cotg \beta_2 < 0 \Rightarrow \frac{\cos \beta_2}{\text{sen } \beta_2} < 0 \Rightarrow \begin{cases} \cos \beta_2 > 0 \\ \text{sen } \beta_2 < 0 \end{cases} \quad (67)$$

$$\begin{cases} \cos \beta_2 < 0 \\ \text{sen } \beta_2 > 0 \end{cases} \quad (68)$$

Dado que los ángulos medidos en la imagen son siempre considerados en el rango $(0^\circ, 180^\circ)$, las expresiones (61), (64), (66) y (67) no podrán darse en ningún caso, reduciendo así los posibles casos de “no evaluación” del inflado previo axonométrico a los dados en las expresiones (62), (63), (65) y (68) que a continuación se analizan.

Las expresiones (62) y (63) se verificarán para un rango de valores dado por:

$$90^\circ < \alpha < 180^\circ \quad \text{y} \quad 0^\circ < \beta < 90^\circ$$

Las expresiones (65) y (68) se verificará para un rango de valores dado por:

$$0^\circ < \alpha < 90^\circ \quad \text{y} \quad 90^\circ < \beta < 180^\circ$$

Es decir, siempre que uno ángulo “x” esté comprendido entre 0° y 90° , y el otro ángulo “y” esté comprendido entre 90° y 180° , el inflado previo axonométrico no podrá ser aplicado. Esta condición queda también ratificada en la gráfica mostrada en la figura 140, donde para dichos valores la $\text{ctg}(x)$ es superior a la $\text{tg}(y)$.

Utilizando términos de la geometría, puede decirse que es inflado previo axonométrico sólo es válido cuando la imagen corresponde a una proyección axonométrica ortogonal (véase anexo 3). No obstante, para ciertas axonometrías oblicuas en los que los ángulos no exceden excesivamente a una axonometría ortogonal, el método propuesto, aunque de manera aproximada, reconstruye las imágenes con éxito (modelos 3 y 5).

Esta limitación explica porqué algunas representaciones a pesar de corresponder a modelos de tipología normalón o cuasi-normalón, no pueden ser reconstruidas mediante la aplicación del inflado previo axonométrico. Tal es el caso de la imagen que se presenta en la figura 141 propuesta por Lamb y Bandopadhyay donde aristas como “a” no pueden ser orientadas a partir de la evaluación de los ángulos que esta forma con las aristas b y c.

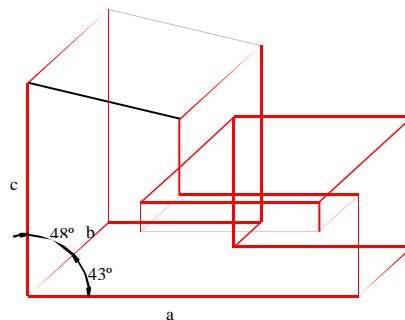


Figura 141. Modelo de tipología cuasi-normalón no reconstruible por inflado previo axonométrico

7.3 Método de inflado previo por plano oblicuo.

El inflado previo por plano oblicuo es un método heurístico aplicable exclusivamente a modelos de tipología piramidal. Mediante dicho método se consigue

una configuración inicial del modelo muy próxima al óptimo global de la función objetivo.

De los 138 ejemplos ensayados en REFER, 8 han sido detectados como modelos de tipología piramidal y todos ellos han sido reconstruidos con éxito. La tabla 142 muestra los ejemplos ensayados.

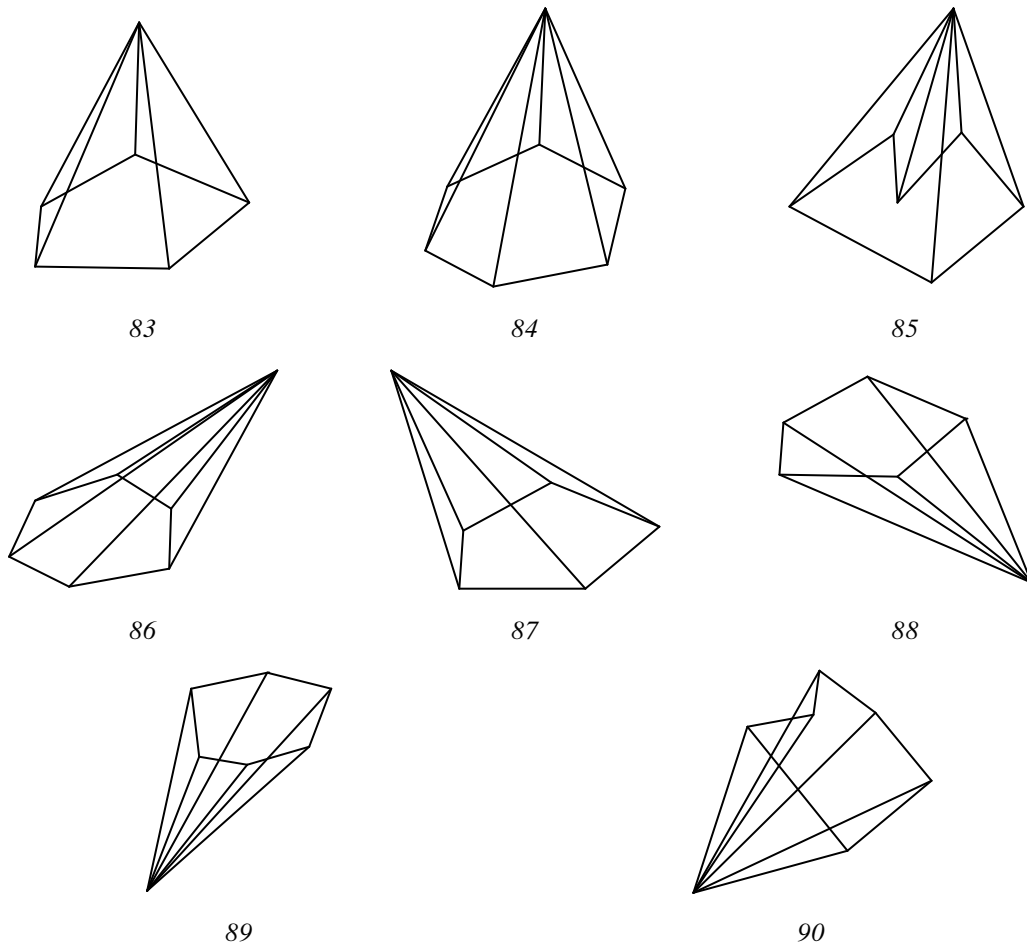


Tabla 142. Modelos piramidales propuestos en esta tesis

En la figura 143 se ha Reconstruido el modelo 83 que muestra la validez del método.

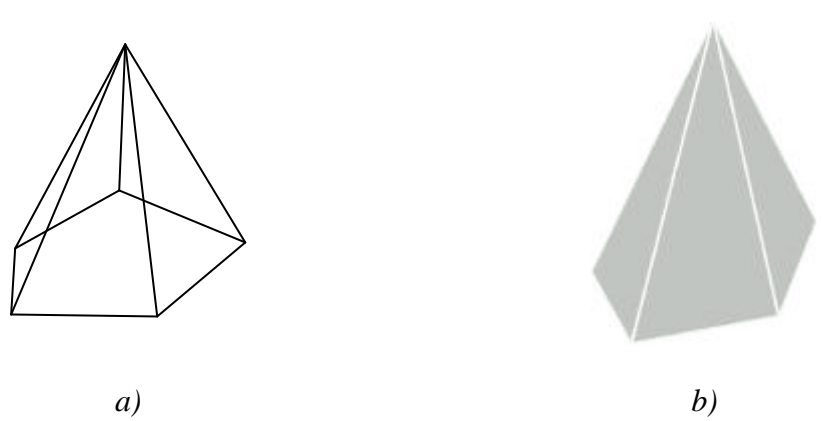


Figura 143. Reconstrucción de un modelo piramidal con inflado previo por plano oblicuo

Sin embargo este método tiene una limitación para el caso de pirámides de base triangular, que proviene de la propia definición de la tipología del modelo. Un objeto se define de tipología piramidal cuando el número de aristas convergentes en un vértice es superior en un cierto valor al número de aristas convergentes en el resto de los vértices del modelo.

Como consecuencia de tal definición los modelos de la figura 144 ensayados por Marill no serian detectados como modelos de tipología piramidal, dado que el número de aristas que convergen en cada uno de los vértices es el mismo, si bien dicha tipología es patente desde el punto de vista de la percepción visual humana.

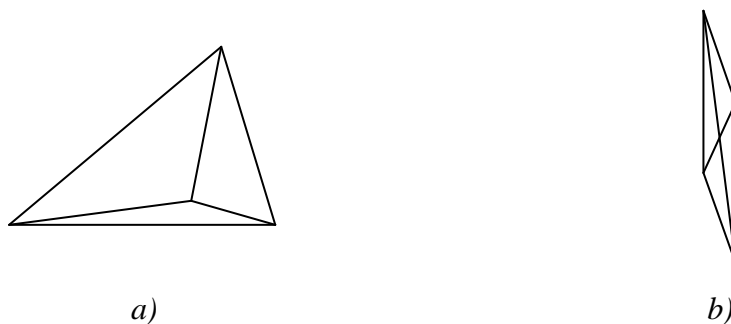


Figura 144. Modelos piramidales detectados con error en la definición de su tipología

7.4 Método de inflado previo por niveles.

La aplicación del método de inflado previo por niveles permite obtener modelos que a pesar de no corresponder con el diseño final que el ser humano interpretaría, como ocurre en el caso de inflado previo axonométrico, suponen puntos iniciales válidos para escapar del óptimo trivial de la imagen y conseguir que el proceso de optimización garantice la consecución de mínimos globales.

Este inflado previo basado en reglas heurísticas puede ser aplicado a todos los modelos con independencia de la tipología de los mismos, si bien en REFER es utilizado en aquellos modelos cuya tipología difiere de la piramidal, normalón o cuasi-normalón, en cuyo caso los inflados previos aplicados son los que anteriormente se han expuesto por ofrecer mejores resultados y menor coste computacional.

El método de inflado previo por niveles presenta sus mejores resultados en modelos convexos de tipología prismática. En la figura 145 se representa un modelo prismático propuesto por Leclerc y Fischler y la reconstrucción del mismo que valida el método propuesto.

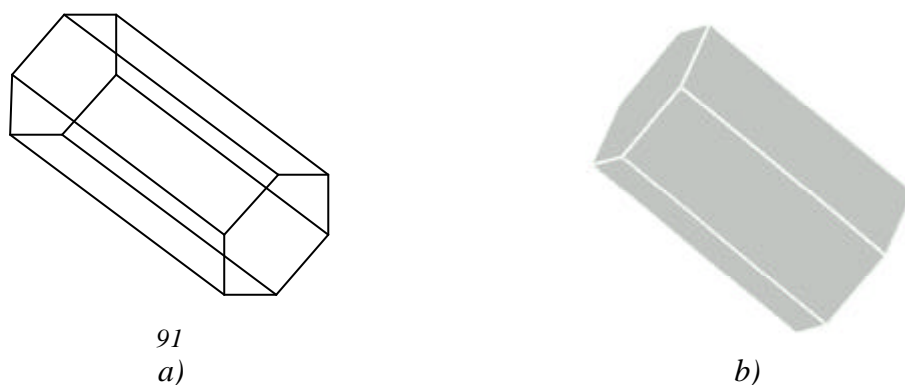


Figura 145. Reconstrucción de un modelo prismático con inflado previo por niveles

En la tabla 146 se muestran otros modelos convexos de tipología prismática que han sido propuestos en el desarrollo de esta tesis y que también han sido reconstruidos con éxito mediante la aplicación del inflado previo por niveles.

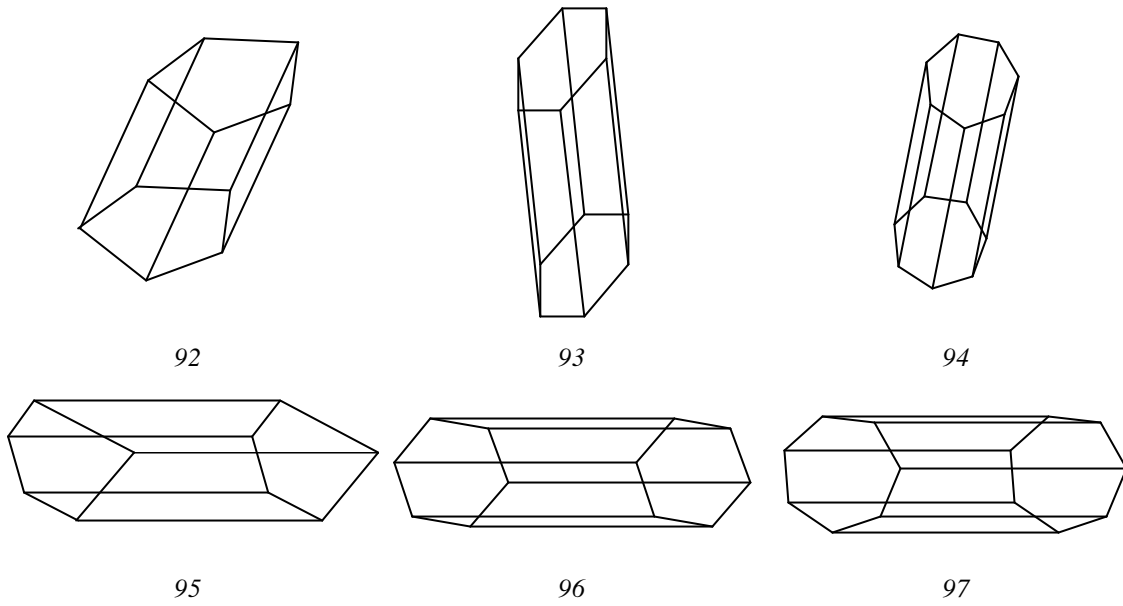


Tabla 146. Modelos prismáticos convexos propuestos en esta tesis

El ensayo de estos modelos pone de manifiesto que la aplicación del método de inflado previo a modelos prismáticos convexos, genera modelos tan próximos al modelo acorde con la percepción visual humana que el posterior proceso de optimización finaliza sin provocar cambios importantes en el mismo. No obstante, es importante notar que se requiere que todos los vértices de la directriz del modelo estén clasificados en tipologías de forma simétrica y correlativa (figura 147).

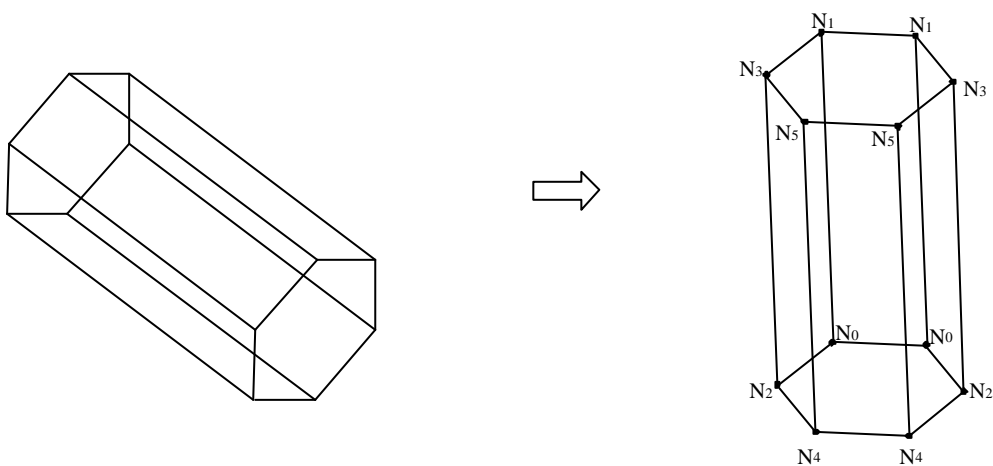


Figura 147. Modelo prismático con tipologías en vértices simétricas y correlativas

Por el contrario, en aquellos modelos en los que tal coincidencia no existe, se producen distorsiones iniciales en el modelo caracterizadas fundamentalmente por incumplir la condición de planicidad de caras, principalmente en las directrices del prisma. Estas distorsiones no pueden ser corregidas durante el proceso de optimización debido a la poca influencia del coste de la regularidad de planicidad de caras cuando los desplazamientos efectuados son de pequeña longitud (esta cuestión será tratada posteriormente con mayor amplitud).

Como consecuencia se precisa de alguna herramienta de refinado que permitan mejorar el modelo obtenido. En esta tesis se ha implementado una aproximación a una de las herramientas propuestas para el refinado del modelo y que hemos denominado refinado por planicidad de caras. El algoritmo implementado será analizado posteriormente.

En los modelos 93 a 96 de la tabla 146 es donde se hace más patente el efecto de distorsión de la planicidad de caras y la mejora que el efecto de refinado propuesto supone en el modelo final reconstruido. Para ilustrar mejor estos efectos, en la figura 148 se muestran los modelos reconstruidos sin refinado por planicidad de caras y con refinado por planicidad de caras del modelo 94. Como puede observarse, los vértices “señalados” que en el modelo sin refinado incumplen la condición de planicidad en cuanto a las caras de su directriz, en el modelo refinado por planicidad de caras son obligados a ser coplanarios con el resto de vértices que definen la cara.

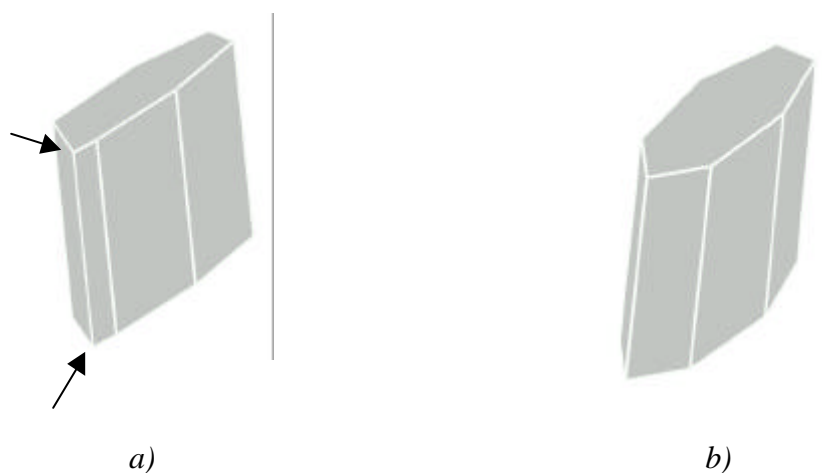


Figura 148. Reconstrucción de un modelo prismático
a) Sin refinado por planicidad de caras b) Con refinado por planicidad de caras

También han sido reconstruidos con éxito modelos cóncavos de tipología prismática si bien todos ellos precisan del refinado por planicidad de caras dada la inexistencia de simetría y correlatividad de tipologías de los vértices de su directriz. En la tabla 149 se muestran los modelos que han sido ensayados en este trabajo.

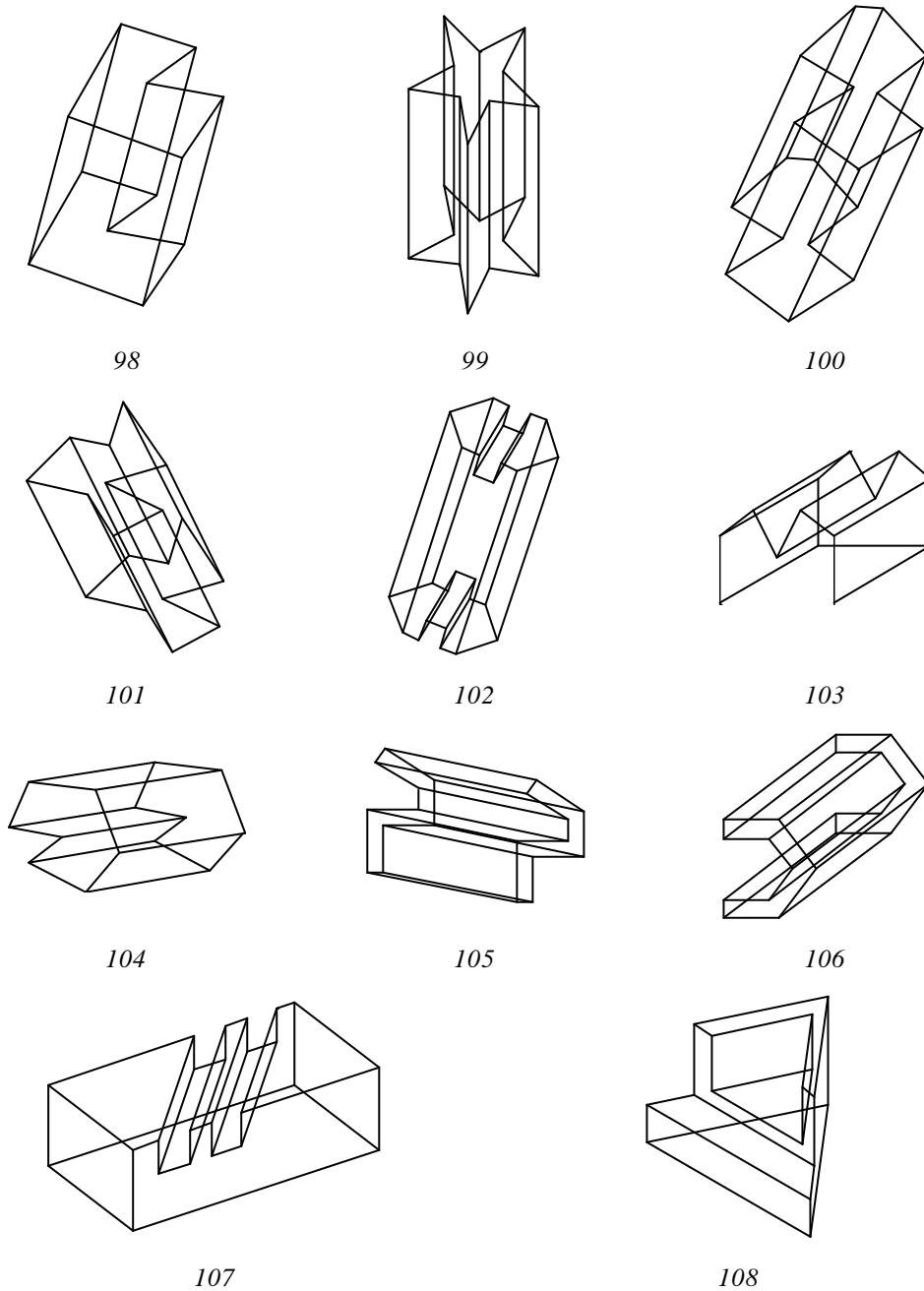


Tabla 149. Modelos prismáticos cóncavos propuestos en esta tesis

Sin embargo, la reconstrucción de modelos de tipología prismática presenta una limitación que nuevamente proviene de la propia definición de la tipología: se considera que una imagen es de tipología prismática cuando el número de aristas orientadas según una dirección predominante suponen como mínimo el 50% del número de nodos en la imagen.

De acuerdo con esa definición los modelos representados en la tabla 150 serían clasificados como modelos de tipología prismática, si bien la reconstrucción de los mismos no puede ser realizada con éxito a partir de los ajustes de los algoritmos realizados para este tipo de imágenes.

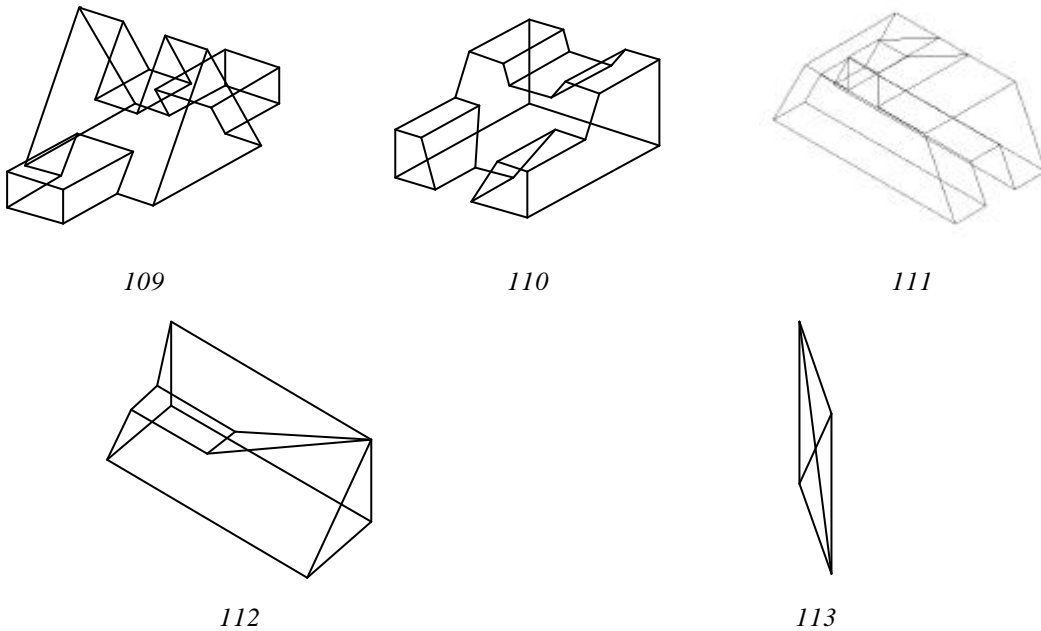
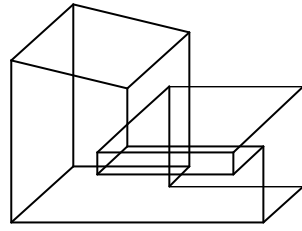


Tabla 150. Modelos definidos erróneamente de tipología prismática

Además de los modelos de tipología prismática, el inflado por niveles es aplicado en aquellos modelos que hemos denominado “indefinidos”, es decir, en aquellos modelos que no pueden ser clasificados en ninguna de las tres tipologías definidas. La figura 151 y las tablas 152, 153 y 154 muestran los 21 ejemplos que han sido detectados de tipología indefinida.



114

Figura 151. Modelo de naturaleza indefinida propuesto por Lamb y Bandopadhyay

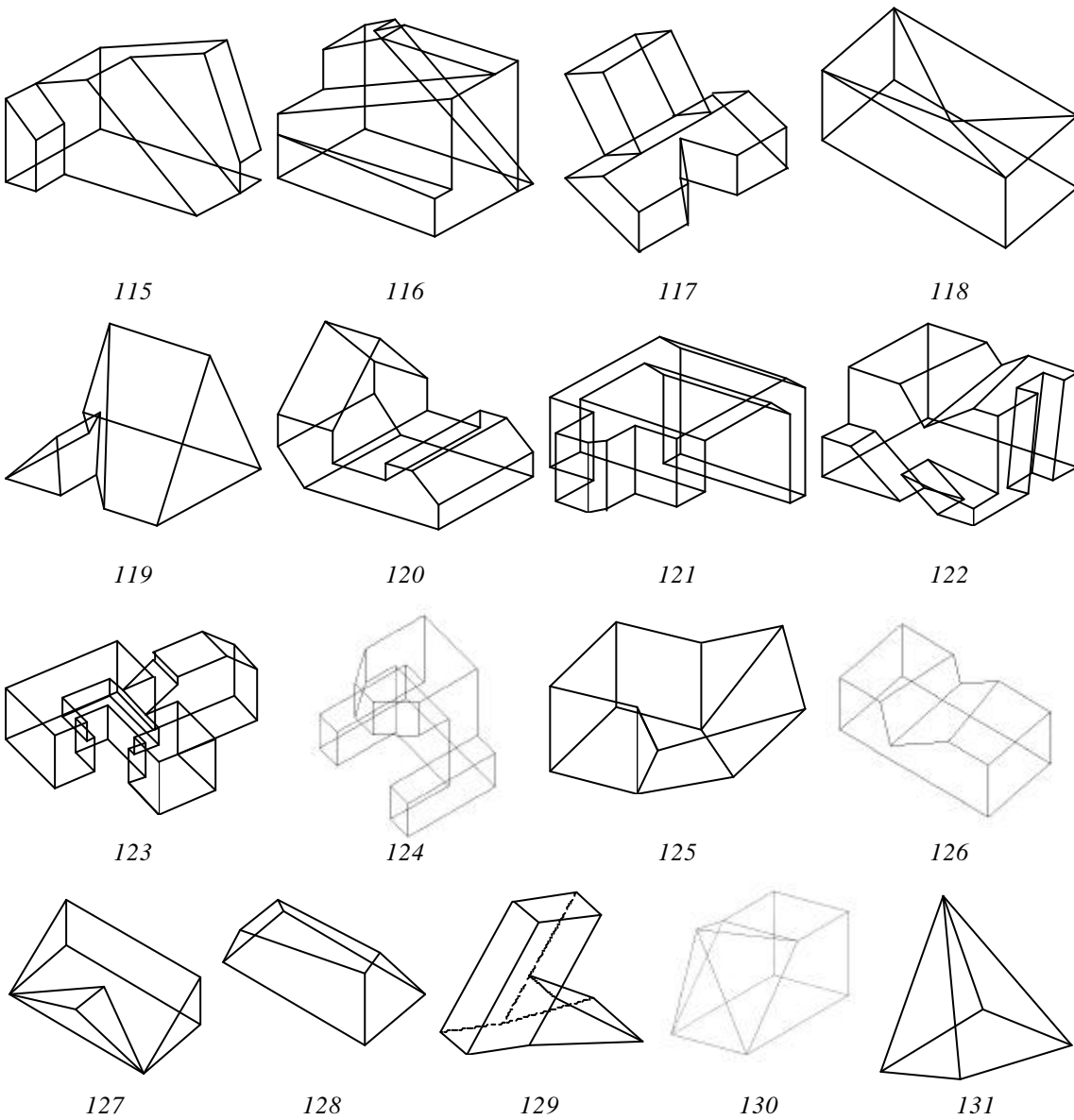
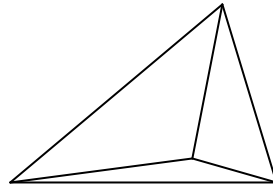
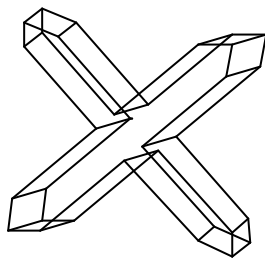


Tabla 152. Modelos que tipología indefinida propuestos en esta tesis

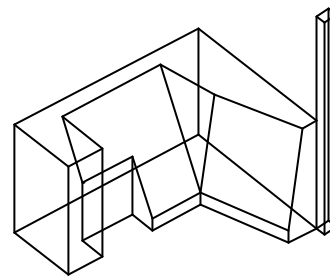


132

Tabla 153. Modelos de naturaleza indefinida propuestos por Marill



133



134

Tabla 154. Modelos de naturaleza indefinida propuestos por Lipson y Shpitalni

El comportamiento particular de cada uno de estos modelos será discutido en la siguiente sección.

7.5 Refinado por planicidad de caras

La estrategia que proponemos consiste en establecer un sistema de ecuaciones en función de las coordenadas “z” de cada vértice que, realizando las mínimas modificaciones posibles, asegure la planicidad de cada una de las caras del prisma. Sin embargo este problema no es de fácil solución en tanto que el sistema propuesto no presenta carácter lineal y posee múltiples soluciones que deben ser analizadas. Por ello, en este trabajo se ha ensayado un método aproximado que permite explorar algunas de las posibilidades de esta estrategia. Esta aproximación consiste en calcular el plano de mejor ajuste para cada una de las caras y obligar a que los vértices que definen la cara se encuentre sobre dicho plano.

El proceso se inicia en las caras que contiene mayor número de vértices y finaliza en las caras con menor número de vértices, de forma que cuando la coordenada “z” de un vértice es modificada para obligar a verificar la condición planicidad en una cara, dicho vértice permanece inalterado durante el resto del proceso.

Evidentemente el proceso así descrito no asegura la planicidad de todas caras del modelo, pero sí la de aquellas caras que tengan mayor número de vértices; mejorando la topología del modelo.

7.6 Selección de regularidades.

El método de Reconstrucción Geométrica de Sólidos propuesto se basa en la optimización de una función objetivo definida como la suma de una serie de términos denominados regularidades de la imagen, que representan reglas de la percepción visual humana y un conjunto de convencionalismos. Sin embargo, para asegurar la consecución del óptimo global de la imagen se precisa una muy cuidada manipulación de dichas regularidades y del peso asignado a las mismas dentro de la función objetivo.

Los métodos ensayados hasta la fecha que tratan la Reconstrucción Geométrica de Sólidos mediante métodos de optimización presenta como inconveniente fundamental la necesidad de una alta interactividad con el usuario que debe seleccionar el conjunto de regularidades que formarán parte de la función objetivo. Pero además, a medida que aumenta la complejidad de la imagen a reconstruir la selección de dichas regularidades se hace más difícil.

Nuestro objetivo es el de crear un sistema de reconstrucción que realice tal selección de regularidades de manera automática en función de la tipología de la imagen así como la elección del método de inflado previo más adecuado.

A partir de la experiencia adquirida de la ejecución de los 138 ensayos realizados presentamos las configuraciones propuestas para las cuatro tipologías definidas.

7.6.1 Modelos de tipología normalón y cuasi-normalón.

La Reconstrucción de modelos de tipología normalón y cuasi-normalón puede obtenerse de manera directa a partir del método de inflado axonométrico sin necesidad

de procesos de optimización. De hecho y como ya se ha comentado con anterioridad, la aplicación de tales procesos después de varios ensayos con una variada combinación de regularidades, muestran que el proceso finaliza sin provocar importantes modificaciones en el modelo final.

7.6.2 Modelos de tipología piramidal.

En las imágenes de tipología piramidal, nuevamente el inflado previo propuesto mediante plano oblicuo permite obtener modelos iniciales acordes con nuestra percepción. No obstante, las propiedades geométricas que en él se asumen representan al mismo tiempo una ventaja y un inconveniente.

Por un lado la planicidad de cara de su directriz, que además resulta ser aquella cara que contiene un mayor número de aristas, permite partir de un modelo que verifica la regularidad que debería exigirse con probabilidad 1. Por otro lado, el asumir que tal pirámide tiene su directriz apoyada en el plano con 45° de oblicuidad, provoca una distorsión en la proporcionalidad de las aristas de la pirámide.

Como consecuencia, la reconstrucción de pirámides mediante plano oblicuo precisa venir precedida de un proceso de optimización que mantenga la planicidad de su directriz y corrija la distorsión en la proporcionalidad de sus aristas.

La planicidad de su directriz puede asegurarse mediante la regularidad de planicidad de caras. Dada la expresión que evalúa dicha regularidad resulta evidente que el coste parcial de cada una de las caras del modelo se verá incrementado con el número de vértices que componen la cara, por lo que la directriz del modelo (cara que contiene mayor número de vértices) será componente principal de esta regularidad. Además el resto de caras de la pirámide están formadas por 3 vértices, por lo que siempre definirán un plano y el coste parcial de dichas caras en la regularidad será nulo. Por otro lado, la regularidad isometría asegura la proporcionalidad de sus aristas distorsionada en el inflado previo.

Además de dichas regularidades, que aseguran las relaciones geométricas entre el modelo y la imagen, es preciso resaltar la necesidad de incluir la regularidad de MSDA con el fin de asegurar que la solución no tienda al óptimo local de la imagen.

Sin embargo la MSDA puede provocar el incumplimiento de las reglas geométricas expuestas con antelación por lo que el peso concedido a dicha regularidad

debe ser inferior a las regularidades de planicidad e isometría. El coeficiente aplicable a dicha regularidad para conseguir un equilibrio apropiado entre todas las regularidades ha sido determinado experimentalmente, quedando así definida la función objetivo para modelos de tipología piramidal en la forma:

$$F = 0,1 R_{\text{MSDA}} + R_{\text{ISOMETRIA}} + R_{\text{PLANICIDAD}} \quad (69)$$

7.6.3 Modelos de tipología prismática.

La reconstrucción de imágenes de tipología prismática propuesta consiste en la aplicación del método de inflado previo por niveles y un posterior proceso de optimización con una elección adecuada de regularidades.

Dos directrices planas y un conjunto de aristas paralelas que unen ambas directrices son las características fundamentales que definen un modelo prismático. Por tanto, ambas características han sido seleccionadas como primordiales para la reconstrucción de imágenes de tipología prismática.

Por un lado, al igual que ocurría en los modelos de tipología piramidal, la planicidad de las directrices del prisma viene garantizada por la regularidad de planicidad dado que dichas directrices contienen un mayor número de vértices, y por tanto el coste de esta regularidad depende fundamentalmente de la planicidad de estas.

Por otro lado, el paralelismo de sus aristas se garantiza mediante la regularidad de paralelismo, cuyo coste viene en este caso dado fundamentalmente por las aristas no pertenecientes a la directriz por superar en número la condición de paralelismo en la imagen de partida.

Pero para conseguir una reconstrucción acorde con la percepción visual humana es preciso considerar dos nuevas reglas. Como consecuencia del inflado previo por niveles, es preciso paliar los defectos de proporcionalidad que se producen en el modelo de partida por lo que se precisa introducir la regularidad isometría. Pero además, y como se ha comentado con anterioridad, para asegurar que la solución no tienda al óptimo local de la imagen se precisa introducir la MSDA como regularidad ponderada en la función objetivo a optimizar.

La ponderación de dicha regularidad para conseguir un modelo acorde ha sido nuevamente establecida experimentalmente quedando la función objetivo definida para imágenes de naturaleza prismática en la forma:

$$F = 0,1 R_{\text{MSDA}} + R_{\text{ISOMETRIA}} + R_{\text{PLANICIDAD}} + R_{\text{PARALELISMO}} \quad (70)$$

7.6.4 Modelos de tipología indefinida.

El comportamiento de los modelos de tipología indefinida, es decir, de modelos que no pueden ser clasificados en ninguna de las tipologías anteriores, difiere mucho en función de las características geométricas de las imágenes que los representa.

A partir de los modelos que hemos ensayados puede adivinarse un doble comportamiento de estos en función de las particularidades de los mismos:

1. Aquellos modelos que presentan una mayoría de aristas paralelas a tres direcciones principales que serían interpretadas por el vidente como direcciones perpendiculares entre sí, es decir, como direcciones coincidentes con la proyección de los ejes del sistema de coordenadas adoptado, evolucionan favorablemente permitiendo su reconstrucción mediante la optimización de una función objetivo definida en la forma:

$$F = 0,1 R_{\text{MSDA}} + R_{\text{PARALELISMO}} + R_{\text{PLANICIDAD}} + R_{\text{ISOMETRIA}} + R_{\text{COLINEALIDAD}} \quad (71) \\ + R_{\text{ORTOGONALIDAD ESQUINAS}}$$

No obstante, el modelo obtenido puede encontrar algunas deficiencias topológicas con respecto a la interpretación del ser humano que pueden ser paliadas si, tras finalizar el proceso de optimización anteriormente propuesto, se reinicia éste siendo la planicidad y la perpendicularidad de caras los únicos componentes de la función objetivo. Un posterior refinado por planicidad de caras se hace preciso en algunos de los modelos Reconstruidos.

Mediante la aplicación de los procesos de optimización descritos se ha Reconstruido con éxito los modelos 110, 111, 113, 114, 116, 117, 120, 121, 122, 123, 126, 129, 133, 134.

La figura 155 ilustra los resultados obtenidos para uno de dichos modelos. Como puede observarse el modelo de la figura 155b muestra claramente la topología del modelo pero presenta un claro incumplimiento de la planicidad de caras que puede ser eliminado tras un segundo proceso de optimización (figura 155c).

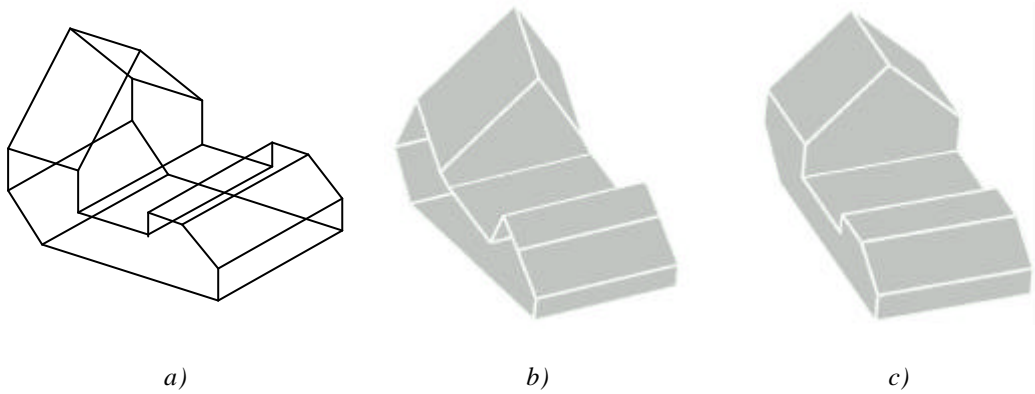


Figura 155. Reconstrucción de modelos de tipología indefinida con predominio de aristas paralelas a tres direcciones principales
a) Imagen de partida b) Modelo obtenido por primera optimización
c) Modelo obtenido por segunda optimización

2. El resto de modelos de tipología indefinida, es decir, aquellos que no verifican la condición de predominio de aristas paralelas a tres direcciones perpendiculares entre sí, permiten su reconstrucción mediante la optimización de la función objetivo

$$F = 0,1 R_{MSDA} + R_{PARALELISMO} + R_{PLANICIDAD} + R_{ISOMETRIA} + R_{COLINEALIDAD} \quad (72)$$

No obstante, y al igual que ocurría anteriormente, una vez finalizado el proceso de optimización pueden paliarse ciertas deficiencias del modelo si se reinicia este siendo la planicidad de caras el único componente de la función objetivo.

Esta estrategia de reiniciar el proceso de optimización es equivalente al refinado por planicidad de caras propuesto para modelos de tipología piramidal y que además suponía un menor coste computacional, sin embargo existe una clara diferencia.

Como se ha comentado con anterioridad, el proceso de refinado por planicidad de caras se inicia en aquellas caras del modelo con un mayor número de vértices y que en el caso de modelos de tipología prismática coincide con las directrices del modelo lo que representa una importante mejora. Por el contrario, en los modelos

de tipología indefinida donde es frecuente encontrar varias caras con idéntico número de vértices, la aproximación del refinado por planicidad de caras provoca en muchas ocasiones intersecciones no deseadas entre las caras del modelo. Por el contrario, el proceso de optimización realiza una mejora global de la planicidad de todas las caras, lo que mejora la topología general del modelo.

La aplicación de los procesos de optimización descritos permite Reconstruir con éxito los modelos 119, 125, 127, 128, 131, 132. En la figura 156 se ilustra la reconstrucción de un modelo mediante el método propuesto.

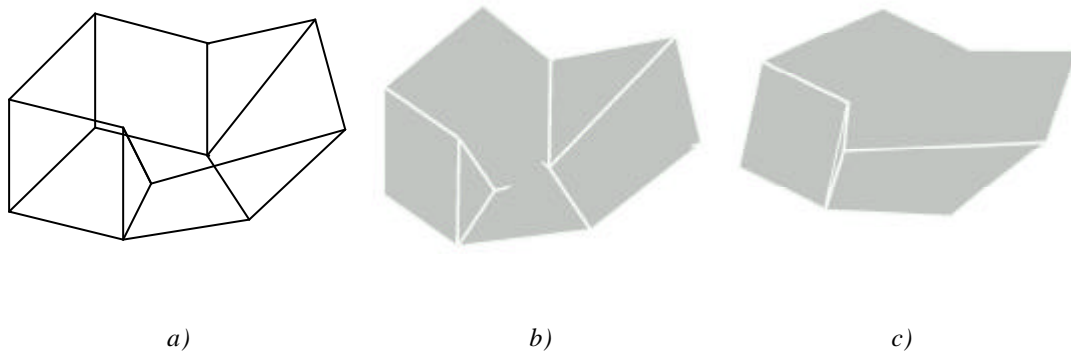


Figura 156. Reconstrucción de modelos de tipología indefinida sin predominio de aristas paralelas a tres direcciones principales
a) Imagen de partida b) Modelos obtenidos por primera optimización
d) Modelo obtenido por segunda optimización

Los procesos descritos permiten la reconstrucción de los modelos de tipología indefinida ensayados. El factor que nos permite diferenciar entre ambos tipos de modelos de naturaleza indefinida ha sido determinado experimentalmente a partir de los ensayos realizados. Consideramos un modelo indefinido del tipo descrito en 1, si al menos el 65% de las aristas son paralelas a tres direcciones principales. En caso contrario el modelo es considerado del tipo descrito en 2.

Un caso especial de modelos clasificados de tipología indefinida lo constituyen los poliedros regulares. En la tabla 157 se muestran cuatro poliedros regulares que han sido ensayados en REFER.

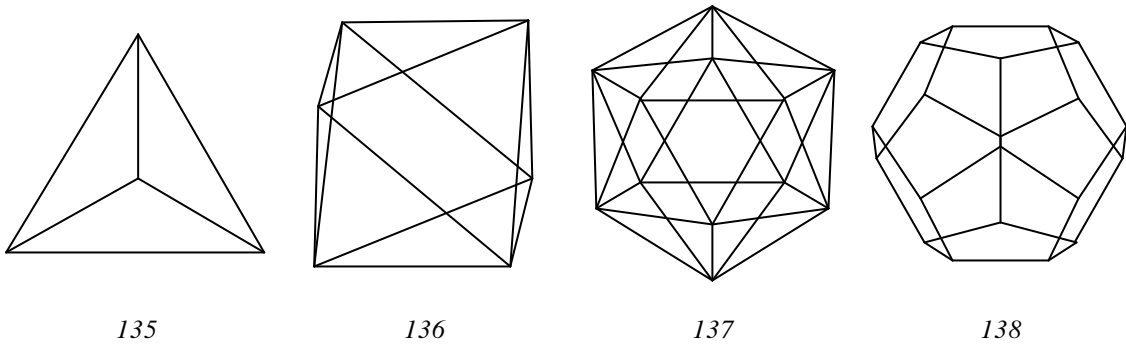


Tabla 157. Modelos poliédricos regulares

Dada su particularidad de tener todos los ángulos iguales la mayoría de estos modelos permiten la reconstrucción mediante una función objetivo con un único componente: la MSDA. Sin embargo, la reconstrucción de estos modelos pone de manifiesto las distorsiones que pueden producirse en el modelo como consecuencia del inflado previo por niveles que deben ser paliadas durante el proceso de optimización. Un ejemplo ilustrativo de estos efectos se muestra en la figura 158b donde se ha reconstruido la imagen del dodecaedro mediante inflado previo por niveles y posterior proceso de optimización, siendo la MSDA el único componente de la función objetivo.

Como puede observarse los vértices 1 y 2 de la imagen dada en 155a son clasificados como tipologías de nivel 5 y 0 respectivamente, lo que provoca un inflado inverso de dichos vértices que impide que la reconstrucción se realice con éxito. Para evitar estos efectos es necesario introducir en la función objetivo la regularidad de planicidad de caras que forzará a salir dichos vértices del interior del modelo. El equilibrio entre ambas regularidades ha sido determinado experimentalmente quedando así definida la función objetivo para poliedros regulares en la forma:

$$F = 0,5 R_{MSDA} + \lambda R_{PLANICIDAD} \quad (73)$$

donde λ es un parámetro que varía constantemente desde 1 hasta 0, y se parte de un inflado previo por niveles una vez detectadas aristas ocultas.

Como consecuencia de la función objetivo propuesta, en las fases iniciales del proceso de optimización la regularidad de planicidad de caras corrige las posibles distorsiones como consecuencia del etiquetado de vértices. En las fases finales del

proceso de optimización (cuando $\lambda = 0$), la regularidad MSDA asegura la característica de regularidad del modelo.

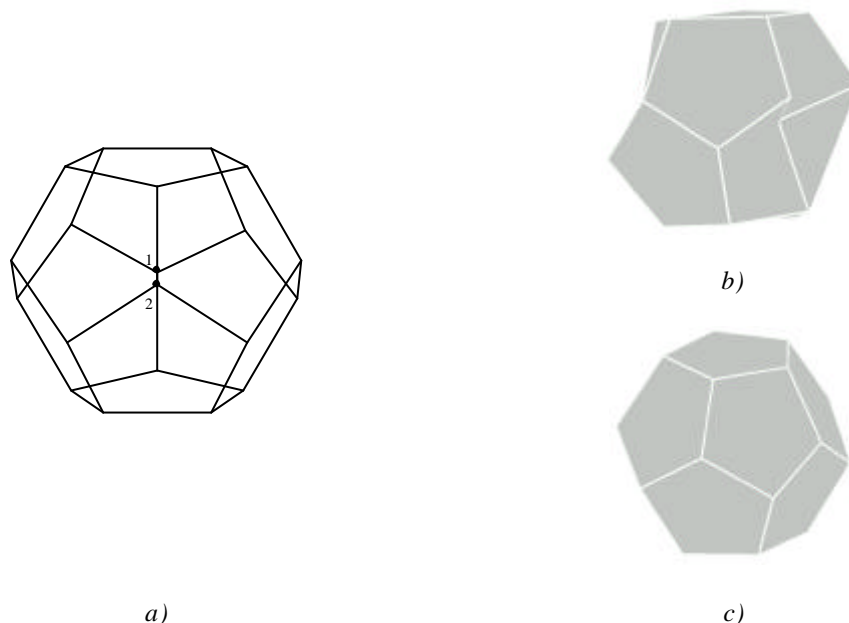


Figura 158. Reconstrucción de modelos poliédricos regulares.

Mediante la aplicación de la función objetivo dada en (73) se logra la reconstrucción de todos los poliedros regulares ensayados, aunque no hemos podido desarrollar ninguna herramienta que permita detectar a partir de la imagen de partida esta característica de regularidad, lo que impide una configuración automática de los parámetros del algoritmo de optimización.

7.7 Particularidades de la regularidad planicidad de caras.

Como se ha comentado con anterioridad, la regularidad de planicidad de caras presenta una característica que le hace diferir del resto de regularidades de la imagen. El hecho es que no debería ser tratada como regularidad en tanto que realmente se trata de una propiedad que deben verificar todos los poliedros: las caras definidas en ellos deben ser planas. Por tanto, tal propiedad debería ser tratada como una regularidad que es verificada con probabilidad 1, es decir, debería ser formulada en el problema de optimización como una restricción.

Sin embargo, en el espacio definido en el que las coordenadas “Z” de cada vértice son obligadas a mantenerse según rectas perpendiculares al plano que contiene la imagen, obligaría a la obtención de la solución trivial de la imagen, en tanto que cualquier variación de la coordenada $Z = 0$ de la imagen incumpliría la restricción propuesta. Y es por ello que la planicidad de caras ha sido introducida en la función objetivo como una regularidad. Pero además, esta regularidad presenta otro inconveniente: el incremento del coste de esta regularidad es bajo para pequeñas modificaciones de coordenadas Z.

Dado que la optimización es un proceso dirigido hacia el menor coste de la función objetivo, aquellos términos que supongan un mayor peso en el coste global de la función, influirán de manera más directa en el proceso de optimización que aquellos otros términos que dependan de regularidades que por su naturaleza supongan pesos inferiores. En otras palabras, el proceso de optimización tenderá a minimizar en primer lugar los términos de mayor valor absoluto, mientras que aquellos términos que tengan menor valor absoluto, serán despreciados. Para paliar este problema se han aplicado sin éxito los siguientes métodos:

- Normalización de las regularidades: donde cada regularidad no es representada por su valor absoluto sino por un porcentaje de variación en las distintas evaluaciones de la función objetivo.
- Implementación de un algoritmo de optimización SA multicriterio. El algoritmo Simulated Annealing Muticriterio al que a partir de ahora nos referiremos como SAM, es una técnica particular del SA utilizada cuando no interesa únicamente la optimización de la función global, sino que además se pretende buscar soluciones que verifiquen la optimización de ciertos sumandos parciales de la función objetivo. Esto es, dada una función: $F(z) = f_1(z) + f_2(z) + \dots + f_N(z)$ donde z es el vector $(z_0, z_1, \dots, z_{n-1})$ que contiene las variables de reconstrucción, la optimización del SAM podría definirse como:

$$\text{SAM } [F(z)] = \text{SA } [f_1(z)] + \text{SA } [f_2(z)] + \dots + \text{SA } [f_N(z)] \quad (74)$$

7.8 Tipologías de los modelos

Tal y como han sido definidos los criterios para la definición de tipologías, resulta fácil comprender que un determinado modelo puede pertenecer a dos tipologías

diferentes. Basta con observar el modelo representado en la figura 139 para entender que responde tanto a un modelo definido de tipología normalón como a un modelo de tipología prismática. Por tanto se precisa establecer algún criterio para decidir a que tipología asociamos estos modelos.

En nuestro caso, el criterio que hemos establecido esta basado exclusivamente en la simplicidad del método de inflado previo que tiene asociado cada tipología, y en el número de éxitos que hemos obtenido tras los sucesivos ensayos. De esta forma hemos establecido un orden de prioridad para analizar la tipología de cada modelo. Cuando un modelo obedece a una determinada tipología finaliza el proceso de análisis, sin comprobar si puede o no pertenecer a otra tipología distinta.

El orden de prioridad establecido analiza en primer lugar si el modelo verifica las propiedades de una tipología normalón. Posteriormente evalúa la tipología prismática y por último la tipología piramidal. Si un modelo no verifica ninguna de las tipologías propuestas de declarado como modelo indefinido.

7.9 Identificación de caras en un grafo.

En este apartado se describen los resultados obtenidos de los ensayos realizados en REFER para la detección de caras en una imagen. A partir de ellos puede deducirse que el algoritmo propuesto funciona bien para la mayoría de los ejemplos propuesto, pero presenta una seria limitación que proviene del elevado número de circuitos que pueden ser generados a partir de los circuitos fundamentales del grafo.

Como ya se ha planteado, el número de circuitos fundamentales de una imagen viene definido por su número ciclomático, que viene dado por la expresión $m - n + 1$, donde “m” representa el número de segmentos y “n” el número de vértices de la imagen. Por consiguiente, el número de caras potenciales obtenidas por combinación de los circuitos fundamentales de un modelo vendría dado por la expresión:

$$\sum_{i=2}^{i=m-n} C_{m-n+1}^i + 1 \quad (75)$$

Mediante la aplicación del teorema de adyacencia, se reduce el tamaño del espacio de circuitos acelerando el proceso de detección. Pero para determinados

modelos, tal reducción puede no resultar suficiente y el algoritmo resulta excesivamente costoso. Por ello se ha previsto una modificación del algoritmo propuesto en [Shpi96].

7.9.1 Reducción de caras potenciales propuesta.

La modificación que nosotros planteamos, consiste en forzar la eliminación de caras a pesar de que algún circuito no encierre a ningún otro. Es decir, no exige que los dos circuitos estén encerrados por otro circuito, pero sí que compartan una arista adyacente entre ellos. Esto implica que somos menos restrictivos a la hora de eliminar y por consiguiente se pueden eliminar circuitos válidos, que son una cara real del objeto 3D. En la figura 159 se presenta un ejemplo de este hecho.

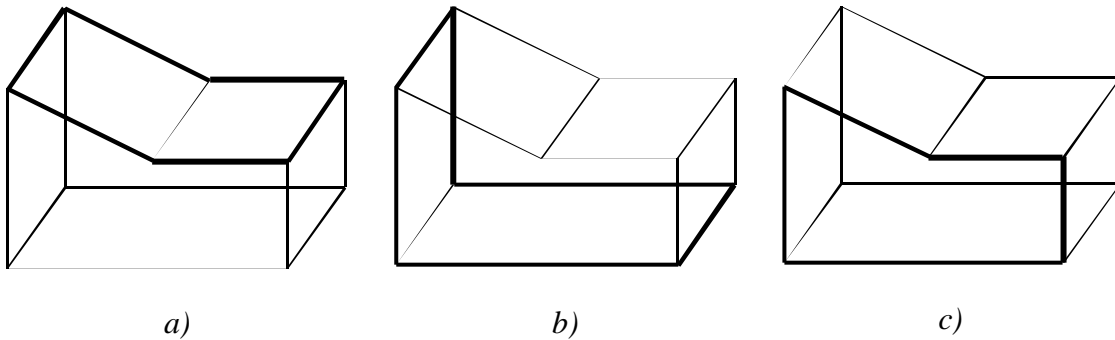


Figura 159. Modificación para la eliminación de circuitos no plausibles

Las figuras 159a y 159b comparten una arista y a su vez ambas comparten más de una arista con la figura 159c. Por tanto, de acuerdo con la modificación propuesta la figura 159c sería eliminada.

Para paliar este defecto, es preciso realizar una ordenación previa del espacio de circuitos comparando inicialmente aquellos que presenten un menor número de aristas. De esta manera, y dado que el circuito de las figuras 159c tiene menor número de aristas, sería analizado previamente a los circuitos de la figura 159a y 159b. Así se evita que las caras con mayor número de aristas eliminen a caras con menor número.

Cabría pensar que si la cara de la figura 159c hubiese tenido un mayor número de aristas que las otras dos, cosa que hubiera sido posible si expandiésemos un poco el modelo hacia la derecha, sí estaría evaluada por los otros dos circuitos y por tanto eliminada. Pero esto no es así porque los circuitos de las figuras 159a y 159b hubieran

sido eliminados previamente por circuitos de menor tamaño como los que se muestran en la figura 160.

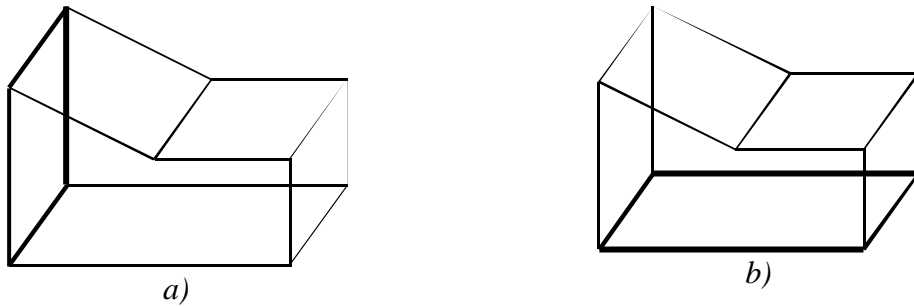


Figura 160. Dos circuitos de menor número de aristas

Las modificaciones propuestas incrementan sensiblemente el número de circuitos eliminados y por tanto, reduce los tiempos de ejecución del programa.

No obstante, en ciertos casos particulares puede ocurrir que la comparación de caras con idéntico número de aristas provoque la eliminación de circuitos que definen caras reales del modelo. El modelo 126 es uno de estos casos donde, por la combinación de los circuitos representados en 156a y 156b, se elimina el circuito de 156c. Como puede observarse, los circuitos de 156a y 156b comparten una sola aristas y a su vez ambos circuitos comparten más de una arista con el circuito 156c.

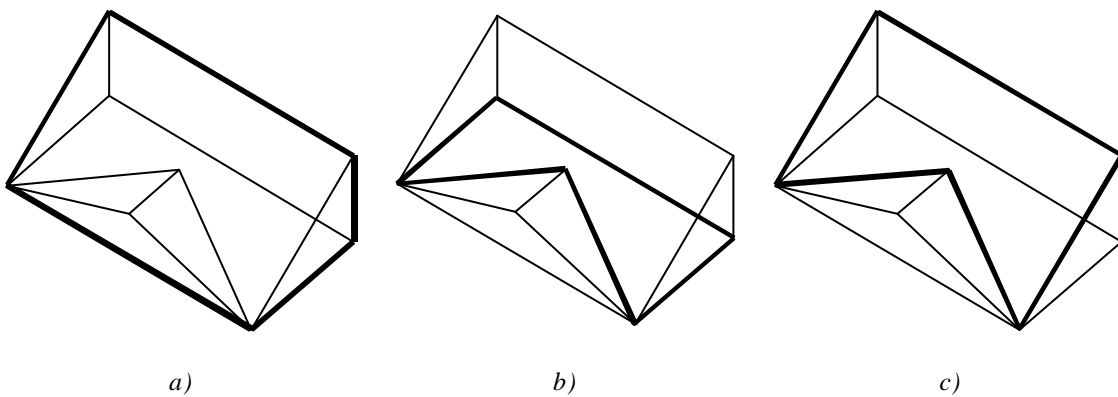


Figura 161. Error en la detección de caras en una imagen

Como consecuencia, las caras detectadas para el modelo 126 quedarían definidas como se muestran en la figura 162



Figura 162. Caras detectadas en el modelo 126

7.9.2 Refinado del número caras definidas en una imagen.

El algoritmo planteado precisa de un refinado final en ciertos modelos, en tanto que el número de caras definido, no es acorde con la percepción visual humana. Muestra de ello es el resultado del análisis realizado en modelos como el que se muestra en la figura 163, donde como resultado se obtienen caras como la definida por los vértices A, B, C y D.

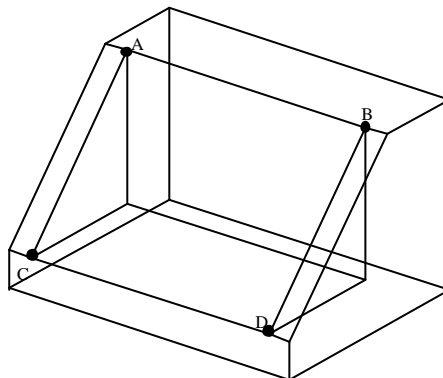


Figura 163. Detección de caras en un modelo sólido

Para eliminar dicho problema se establece el siguiente teorema:

“Cada arista del modelo deberá pertenecer como máximo a dos caras”

Por consiguiente las aristas definidas por los vértices AB, BD, DC y AC no verifica dicho teorema, y cualquier cara que contenga como mínimo a dos de dichas aristas será eliminada.

En las tablas 164, 165, 166 y 167 se muestra el análisis de tiempos de ejecución del algoritmo para algunos modelos poliédricos. El programa ha sido ejecutado en un ordenador Pentium II 300 MHz.

- **Identificación de circuitos fundamentales.**

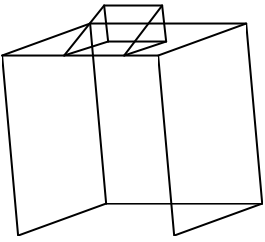
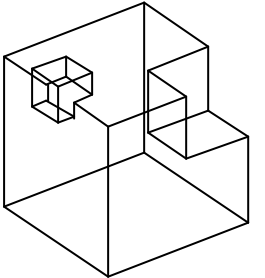
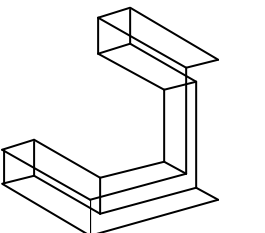
Figura	Características	Resultados ejecución	Tiempo ejecución
	14 vértices 22 aristas	9 circuitos Fundamentales	2 milésimas de segundo
	26 vértices 39 aristas	14 circuitos fundamentales	55 milésimas de segundo
	20 vértices 30 aristas	11 circuitos fundamentales	3 milésimas de segundo

Tabla 164. Análisis de la identificación de circuitos fundamentales

- Generar el resto de circuitos del grafo.

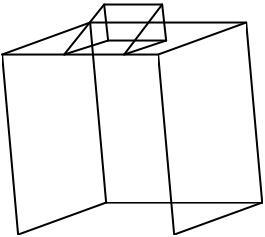
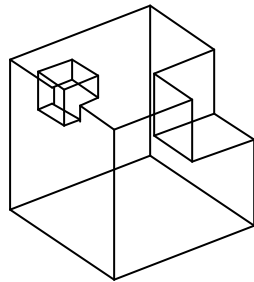
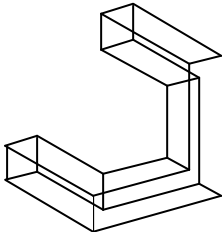
Figura	Características	Resultados ejecución	Tiempo ejecución
	14 vértices 22 aristas 9 circuitos Fundamentales	511 circuitos posibles 45 válidos	41 milésimas de segundo
	26 vértices 39 aristas 14 circuitos fundamentales	16383 circuitos posibles 176 válidos	981 milésimas de segundo
	20 vértices 30 aristas 11 circuitos fundamentales	2047 circuitos posibles 185 válidos	110 milésimas de segundo

Tabla 165. Análisis del resto de circuitos

• **Eliminación de circuitos no plausibles.**

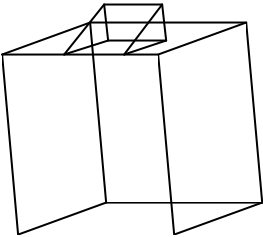
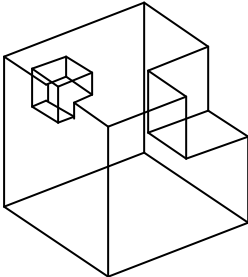
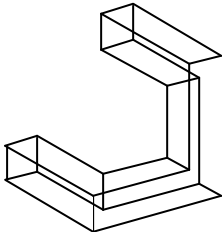
Figura	Características	Resultados ejecución	Tiempo ejecución
	<p>14 vértices 22 aristas 9 circuitos Fundamentales 45 válidos</p>	<p>15 circuitos no discriminados</p>	<p>1 milésimas de segundo</p>
	<p>26 vértices 39 aristas 14 circuitos fundamentales 175 circuitos válidos</p>	<p>52 circuitos no discriminados</p>	<p>23 milésimas de segundo</p>
	<p>20 vértices 30 aristas 11 circuitos fundamentales 185 válidos</p>	<p>23 circuitos no discriminados</p>	<p>96 milésimas de segundo</p>

Tabla 166. Análisis de la eliminación de circuitos no plausibles

- Selección de las caras del modelo.

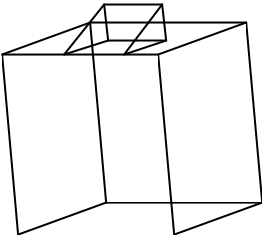
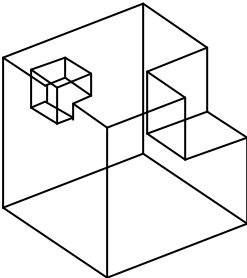
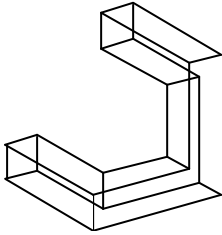
Figura	Características	Tiempo ejecución
	<p>14 vértices 22 aristas 9 circuitos Fundamentales 45 válidos 15 circuitos no discriminados 10 caras válidas</p>	<p>358 milésimas de segundo</p>
	<p>26 vértices 39 aristas 14 circuitos fundamentales 175 circuitos válidos 52 circuitos no discriminados 15 caras válidas</p>	<p>8 segundos y 523 milésimas</p>
	<p>20 vértices 30 aristas 11 circuitos fundamentales 185 válidos 23 circuitos no discriminados 12 caras válidas</p>	<p>2 segundos y 159 milésimas</p>

Tabla 167. Análisis de la selección de caras

A pesar de la reducción del espacio de circuitos el algoritmo de detección de caras sigue siendo inviable para modelos con un elevado número de aristas. Uno de estos modelos es el 120, que precisa más de 45 minutos para la detección de caras.

7.10 Detección de aristas ocultas.

El algoritmo de aristas ocultas implementado funciona bien en la mayoría de los casos que han sido ensayados, sin embargo presenta cuatro limitaciones fundamentales.

- La primera de estas limitaciones esta asociada al tamaño del problema. Como ya se ha dicho el número de configuraciones posibles a partir de una imagen crece exponencialmente en la forma 2^N donde N representa el número de aristas de la imagen. Se han empleado dos técnicas que permiten reducir el tamaño de estas configuraciones, pero a pesar de esta reducción, en algunas ocasiones sigue siendo excesivamente grande, por lo que la evaluación del método supone un alto coste computacional.
- El método presentado analiza todas las posibles configuraciones y elimina aquellas que son incompatibles con un conjunto de reglas. Sin embargo, y dado que el método esta basado en reglas heurísticas es posible obtener configuraciones que a pesar de verificar dichas reglas no representen configuraciones válidas. En la figura 168 se muestran las configuraciones válidas obtenidas para una imagen.

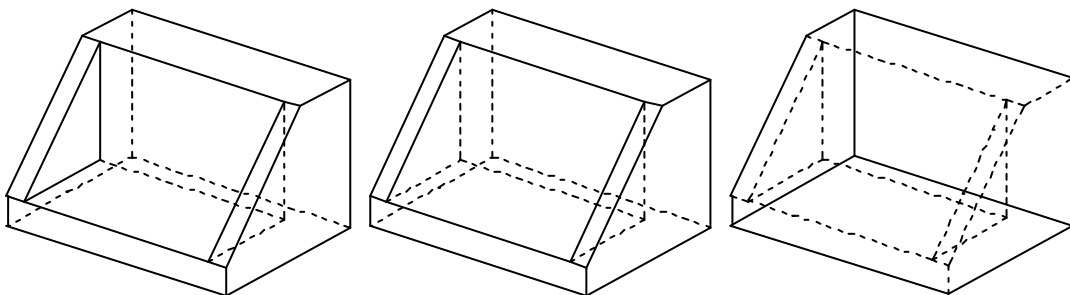


Figura 168. Configuraciones validas en la detección de aristas ocultas

- Como consecuencia del método de tratamiento agrupado de subconjuntos de aristas utilizado para reducir el número de configuraciones, algunas configuraciones válidas pueden no ser analizadas. En la figura 169 se muestra un ejemplo de este hecho, donde para obtener la imagen ahí definida como consecuencia de la detección de aristas parcialmente ocultas deberían verificarse las siguientes condiciones:

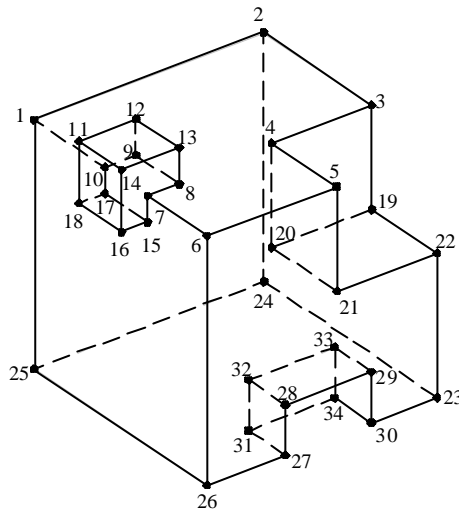


Figura 169. Configuración no analizada en la detección de aristas parcialmente ocultas

1. Para que la arista definida por los vértices 2 y 24 se configure como oculta, se precisa que las caras definidas por 2, 3, 19, 22, 23, 24 y 1, 2, 24, 25 sean ocultas.
2. Para que la arista definida por los vértices 8 y 9 se configure como oculta, se precisa que las caras definidas por 8, 9, 12, 13 y 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 sean ocultas.

Sin embargo, si ambas condiciones se dan, las aristas 1,2 y 2,3 del contorno aparente de la imagen también serían ocultas y por tanto la configuración sería rechazada.

- La obtención de distintas configuraciones válidas en el proceso de detección de aristas ocultas precisa que el usuario seleccione una de estas configuraciones, lo que resulta un inconveniente para la automatización del proceso de reconstrucción.

Por último, se debe notar que siempre existirán al menos dos configuraciones válidas correspondientes a las inversas de Necker.

La conclusión final a la que hemos llegado tras la ejecución de este algoritmo también es válida para el método de inflado previo por niveles incrementales. El objetivo perseguido por ambos métodos es el de reducir las distorsiones que se producían como consecuencia del inflado previo por niveles tal y como se mostró en la figura 101a. Sin embargo, tales distorsiones son eliminadas durante el proceso de optimización que sigue al inflado previo. Es decir, el punto inicial obtenido mediante el inflado previo por niveles fijos y sin detección de aristas ocultas resulta lo suficientemente válido como para conseguir una reconstrucción con éxito. Es por ello que creemos prescindir, de momento, de este algoritmo; que además de suponer en ocasiones un alto coste computacional precisa interactuar con el usuario.

7.11 Simplificación del espacio de soluciones.

Mediante el análisis de las caras de un grafo se puede reducir y simplificar significativamente el proceso de reconstrucción, ya que se reducen el número de grados de libertad de las variables. A modo de ejemplo, en el caso de un cubo, mediante el reconocimiento de sus seis caras, se permite establecer seis relaciones de planicidad (una para cada cara del cubo) reduciendo así el número de grados de libertad de ocho a dos.

Además, debido a que la profundidad de un vértice puede ser establecida arbitrariamente, el número de grados de libertad se reduce a uno. Por consiguiente y como se ha puesto de manifiesto, la identificación de caras reduce la complejidad del problema de reconstrucción.

No obstante y como ya se ha comentado con anterioridad, el problema de reconstrucción se encuentra aún en fases experimentales y los modelos ensayados presentan una complejidad limitada. Es decir que el tamaño del problema es abordable sin la necesidad de emplear técnica de reducción, por lo que en esta tesis no se ha hecho referencia a esta posibilidad de reducir la complejidad del problema por ninguno de los métodos expuestos.

CONCLUSIONES Y DESARROLLOS FUTUROS

Reconstruir poliedros es viable, y hacerlo por medio de optimización es eficiente. En esta tesis se ha desarrollado, implementado y probado una aplicación informática que demuestra la afirmación anterior. El método es susceptible de mejora, pero responde a las expectativas iniciales.

No obstante, para que la reconstrucción tenga utilidad práctica hay que tratar con objetos más complejos (superficies curvas), tratar con mezcla de información gráfica, textual y simbólica (símbolos normalizados), y trabajar a partir de interfases más amigables para el usuario (interfases de tipo caligráfico [Naya01]).

La generación automática de modelos sólidos a partir de dibujos de ingeniería es la forma más eficiente de establecer una comunicación fluida entre diseñadores y sistemas CAD. Este es reto de la Reconstrucción Geométrica de Sólidos.

Nuestro objetivo actual es generar un sistema automático para reconstruir objetos, si bien un paso previo lo constituye la conversión de dibujos de ingeniería en un lenguaje de entrada válido para los sistemas de diseño. Y este es aún un reto pendiente de resolver. Además, se precisa de la identificación de formatos de texto y de simbología presente en los dibujos técnicos para una posterior etapa de refinado que defina un modelo acorde a las especificaciones técnicas.

8.1 Conclusiones finales.

La reconstrucción comenzó planteándose como un problema geométrico. Se enfocó como el problema inverso al estudiado por la geometría proyectiva (a saber: rescatar el modelo a partir del cual se había “proyectado” la imagen).

El enfoque perceptivo siempre ha existido, pero predominaba la idea de que tenía un papel subordinado dentro de un planteamiento geométrico del problema.

Actualmente, con trabajos como esta misma tesis, está ganando fuerza el planteamiento de que se trata de un problema *perceptivo* con aspectos geométricos subordinados. Es decir, que los seres humanos construimos mundos tridimensionales a

partir de lo que “vemos” fisiológicamente hablando. Por tanto, igual que la fisiología del ojo es algo coyuntural y complementario en el proceso de visión “natural”, se tiende a aceptar que la geometría proyectiva es también algo coyuntural en la visión artificial.

Nuestras principales aportaciones se pueden resumir en:

- Ajuste de los algoritmos de optimización para conseguir un funcionamiento automático.

Se han definido los distintos parámetros de optimización en función de las propiedades que pueden deducirse de la imagen. Lo que supone un alto grado de automatización en el proceso de reconstrucción.

No obstante, y a pesar de la automatización realizada en la elección de los parámetros que controlan los algoritmos de optimización, dada la complejidad del espacio de soluciones no se ha podido encontrar ningún algoritmo de optimización que asegure la obtención de mínimos globales en la Reconstrucción Geométrica, por lo que la utilización exclusiva de algoritmos de optimización a partir de la imagen, es decir, sin estrategias para elaborar puntos iniciales válidos, es un camino erróneo para solucionar el problema de la Reconstrucción Geométrica.

- Elaboración de estrategias de inflado previo, para evitar mínimos locales.

Los métodos de inflado por optimización han sido presentados como una estrategia adecuada en la Reconstrucción Geométrica de sólidos a partir de figuras axonométricas para conseguir obtener modelos acordes con la percepción visual humana. Sin embargo esta estrategia presenta importantes problemas dada la existencia de mínimos locales que corresponden a modelos inconsistentes para el vidente.

Para evitar dicho problema se ha planteado la necesidad de elaborar estrategias que permitan generar puntos de diseño iniciales a partir de la información contenida en la imagen, tan próximos como sea posible al óptimo global.

Las estrategias ensayadas permiten generar puntos de diseño iniciales que permiten a los procesos de optimización llegar al óptimo global incluso utilizando algoritmos basados en criterios simples de descenso. Por consiguiente, se aconseja la utilización de algoritmos simples como Hill-Climbing que suponen un menor coste computacional.

- Elaboración de clasificación por tipologías y detección automática de las mismas.

Se ha justificado que, dado no existe una estrategia única que permita obtener puntos iniciales válidos. En otras palabras puede decirse que la estrategia a adoptar para generar puntos de iniciales depende de la tipología del modelo a reconstruir. Por tanto, es necesario detectar de manera previa y automáticamente la tipología del objeto a reconstruir. Se han propuesto algunos criterios para detectar tipologías.

En definitiva, se ha justificado que las estrategias de inflado se basan en la tipología de los objetos a reconstruir, por lo que la capacidad de detectar automáticamente dicha naturaleza, a partir de propiedades observadas en la figura de partida, es fundamental para conseguir aplicaciones capaces de reconstruir automáticamente todo tipo de objetos.

Se han presentado tres estrategias que han sido elaboradas para generar modelos tridimensionales previos al proceso de optimización, se han descrito sus fundamentos y se han detallado los métodos que permiten aplicarlas para reconstruir automáticamente modelos poliédricos de diferentes tipologías. Las estrategias están basadas en algoritmos simples, automáticos y de rápida ejecución.

8.2 Desarrollos futuros.

Este trabajo se ha limitado a la reconstrucción automática de modelos geométricos a partir de la información contenida en una única imagen vectorial y geoméricamente consistente de un objeto poliédrico.

Por tanto, los ámbitos restantes son objeto de posibles desarrollos futuros. De entre ellos los más importantes a resaltar son:

- Tratar con modelos técnicos. Es decir, modelos que trascienden la simple información geométrica y añaden información de diseño. Lo que también se ha dado en llamar “semántica” del producto.
- Tratar con información geométrica procedente de otro tipo de imágenes. Si bien el grupo tiene un desarrollo basado en la reconstrucción a partir de las tres vistas ortográficas principales (alzado, planta y perfil), en esta tesis, no se han incorporado dichos desarrollos.

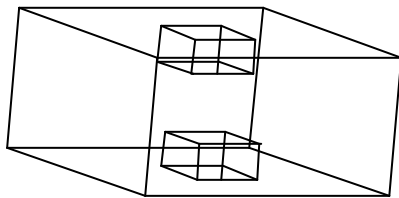
- La capacidad para trabajar con imágenes en mapa de bits, además de las imágenes vectoriales que actualmente se pueden procesar, es fundamental para abarcar la reconstrucción de dibujos hechos a mano.
- Por último, la capacidad de filtrar automáticamente un gran número de imperfecciones también es clave para aumentar la comodidad del usuario y la eficacia de la aplicación.

8.2.1 Desarrollos abordables a corto plazo.

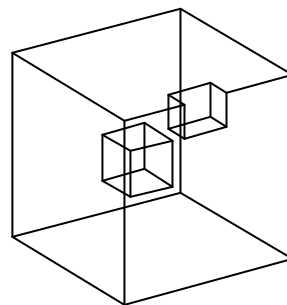
El primero de los objetivos que nos planteamos es el desarrollar una clasificación más depurada de tipologías de modelos que permita mejorar la reconstrucción automática. Debemos recordar que existen distintos comportamientos en los modelos que hasta ahora hemos clasificado como indefinidos que incluyen la necesidad de un doble proceso de optimización y en ocasiones realizar un refinado por caras.

Dicha clasificación además, debe incluir algunas familias de modelos que aún no han sido consideradas. De entre estas familias debemos destacar:

- Modelos no eulerianos, es decir modelos que viene representados por más de un grafo conexo (tabla 170). Con los desarrollos que hemos realizado hasta el momento dichos grafos pueden ser reconstruidos como objetos independientes, pero se precisa definir la posición relativa de los mismos en el espacio.



144



145

Tabla 170. Modelos no eulerianos

Para ello tenemos previsto la aplicación de nuevas reglas de percepción tales como la que define la ley de la proximidad (regla 4 de Hoffman):

“Tendemos a ver las cosas próximas entre sí como relacionadas”

De acuerdo con dicha ley, si los elementos de una imagen están cerca, en 3D también lo estarán. En la figura 171a se representa el cubo de Necker con burbujas diseñado por el psicólogo Marc Albert.

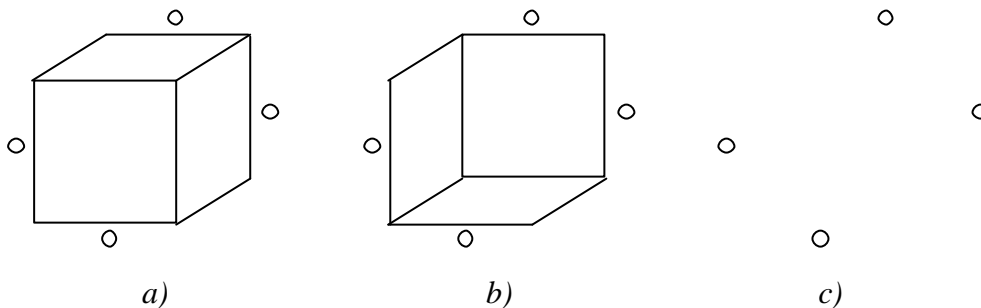


Figura 171. Ley de la proximidad

En él, parece que las burbujas tienen profundidad distinta, dos están más adelantadas y dos están más atrás. La profundidad de cada burbuja viene a ser la misma que la profundidad que tiene el canto del cubo más cercano a ella. Cuando el cubo invierte su profundidad (figura 171b) también lo hacen las burbujas. Pero además, si eliminamos el cubo de la imagen (figura 171c) todas las burbujas parecen coplanarias. Por consiguiente, las burbujas heredan la profundidad del cubo y la regla de herencia es la proximidad: cada burbuja hereda su profundidad de la porción del cubo que en la figura esté más cercana a ella.

- Modelos alámbricos (en el caso de su tipología no sea normalón), es decir, modelos que vienen representados por un conjunto de aristas que no definen contornos de caras. La particularidad de estos modelos hace que las regularidades de planicidad y perpendicularidad de caras no puedan ser utilizadas en el proceso de reconstrucción,

por lo que el problema resulta más complejo. En la tabla 172 se representan algunos de estos ejemplos.

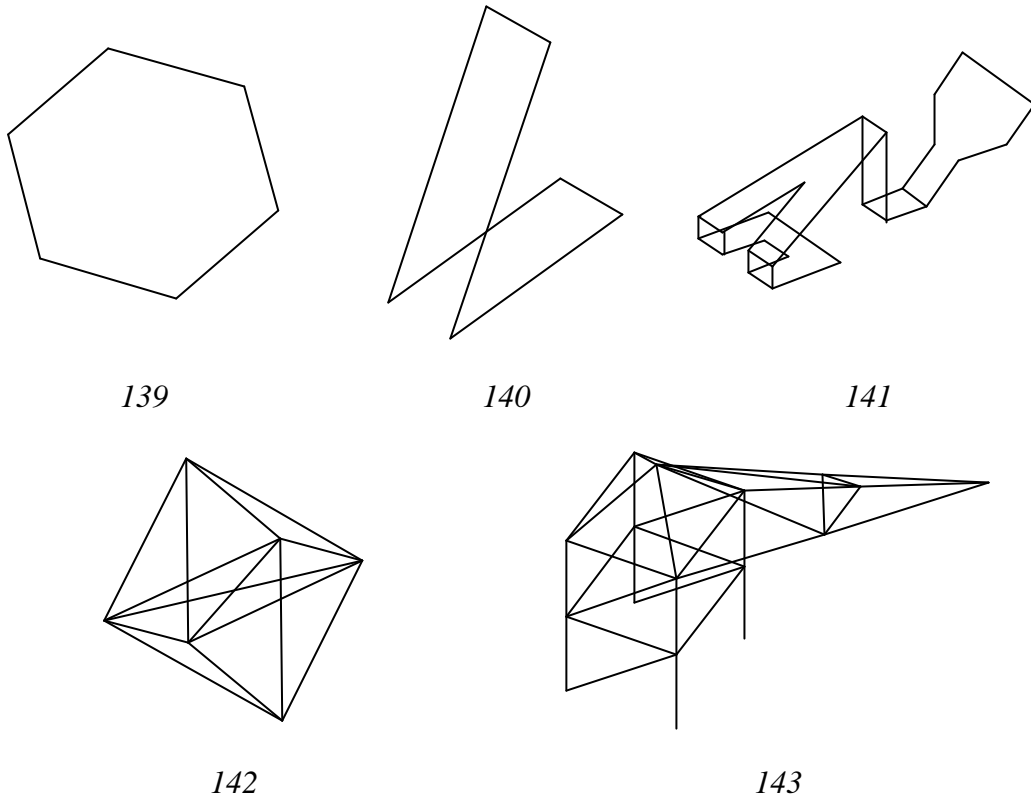


Tabla 172. Modelos alámbricos

El desarrollo de nuevas técnicas de inflado previo constituye otro de los trabajos que pueden ser abordados a corto plazo. Algunas de estas técnicas se proponen como combinación de las existentes. A modo de ejemplo, en la figura 173 se representa una imagen que puede ser reconstruida por un inflado previo axonométrico, a excepción del vértice “A”. Se plantea la posibilidad de realizar este tipo de inflado en el resto de los vértices y posteriormente asignar una coordenada “z” al vértice “A”, comparando el nivel de este vértice con el nivel de los vértices conectados a él.

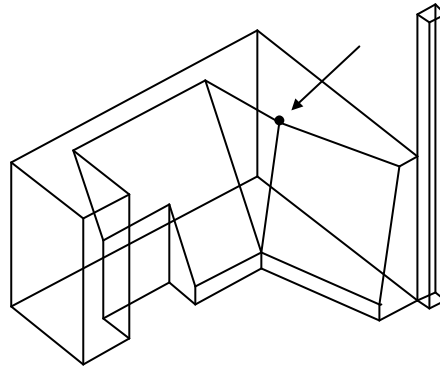


Figura 173. Imagen que permite combinación de métodos de inflado previo

8.2.2 Desarrollos no abordables a corto plazo.

De entre los objetivos que nos hemos planteado desarrollar, pero que no son abordables a corto plazo podemos destacar:

1. Incorporar la Reconstrucción de modelos con superficies cuadráticas y de revolución.
2. Integrar en REFER algoritmos que permitan la obtención de figuras delineadas en 2D a partir de bocetos reales. Este objetivo se plantea en dos fases:
 - Implementar los algoritmos de vectorización que han sido desarrollados en el grupo.
 - Implementar una aplicación que permita introducir las figuras dibujando sobre tabletas digitalizadoras. Para ello se requiere comprobar las capacidad de interacción de las nuevas tabletas LCD que permiten visualizar interactivamente sobre la misma superficie en la que se dibuja.
3. Implementar un módulo postprocesador, que sea capaz de filtrar los errores e imperfecciones de las figuras 2D vectorizadas. Es decir, un módulo de reconstrucción de la imagen que facilite los datos que requieren los procesos habituales de reconstrucción. Este objetivo persigue acercar definitivamente la reconstrucción a los diseñadores, porque pretende tratar el problema de la síntesis en el proceso de diseño de un entorno más amigable y cercano a los diseñadores, comenzando a eliminar las trabas de los sistemas actuales de diseño.

REFERENCIAS

-
- [Alde83] Aldefeld B. "On Automatic Recognition of 3D Structures from 2D Representations". Computer Aided Design, vol.15, No. 2, pp. 59-64, (1983).
- [Alde84] Aldefeld B. and Richter H. "Semiautomatic Three-Dimensional Interpretation of Line Drawings". Computer & Graphics, vol. 8, No. 4, pp. 371-380, (1984).
- [And00] <http://www.tec.uji.es/regeo/proyectos.htm>
- [Attn69] Attneave, F. and Frost, R. "The determination of perceived tridimensional orientation by minimum criteria". Psychonomic Journals, Perception & Psychophysics, vol 6, pp. 391-396, (1969) Texas.
- [Bair91] Baird, L. C., & Wang, P. S. "3D object recognition using gradient descent and the universal 3D ray grammar". Proceedings of SPIE Conference on Intelligent Robots and Computer Vision X: Algorithms and Techniques, Boston, MA, vol. 1607 (1991).
- [Chen92] Chen Z., Perng D., Chen C. and Wu C. "Fast reconstruction of 3D mechanical parts from 2D orthographic views with rules". Int. J. Computer Integrated Manufacturing, vol. 5, No. 1, pp. 2-9, (1992).
- [Chen96] Chen C.L.P. and Xie S. "Freehand drawing system using a fuzzy logic concept". Computer Aided Design, vol. 28, No. 2, pp. 77-89, (1996).
- [Clow71] Clowes, M.B. "On seeing things". Artificial Intelligence, vol 2, pp. 19-116 (1971).
- [Comp97] Company P. "Integrating Creative Steps in CAD Process". International Seminar on Principles and Methods of Engineering Design, Napoli, vol. 1, pp. 295-322, (1997).
- [Comp99(1)] Company P., Gomis J.M. y Contero M. "An Optimization-Based Algorithm to Reconstruct 3D Models from Single Line Drawings". Progettazione e fattibilità dei prodotti industriali, Diseno y fabricabilidad de los productos industriales, (a cura di) Lanzotti A. e Patalano S. Giannini Editore (ISBN 88-900081-3-X), Napoli, marzo 1999., pp 3-12.
- [Comp99(2)] Company, P., Gomis, J.M. y Contero, M., "Geometrical Reconstruction from Single Line Drawings Using Optimization-Based Approaches". WSCG'99 Conference proceedings, edited by Vaclav Skala, ISBN 80-7082-490-5, volume II, 1999, pp. 361-368.
- [Cone99] Conesa, J., Company, P. y Gomis, J.M., "Initial modeling strategies for geometrical reconstruction optimization-based approaches". 11th ADM International conference, vol B, pp. 161-171 (1999).

- [Cone00] Conesa, J., Contero, M. y Company, P, “Comportamiento de los algoritmos de optimización en la Reconstrucción Geométrica de Sólidos”. Anales de Ingeniería Mecánica. XIV Congreso Nacional de Ingeniería Mecánica, vol I, pp. 425-430 Madrid (2000).
- [Cugi91] Cugini U. “The problem of user interface in geometric modelling”. Computer in Industry, vol.17, pp. 335-339, (1991).
- [DeBo95] De Bonet, J.S. “Reconstructing rectangular polyhedra from hand-drawn wireframe sketches” (1995).
- [Dori92] Dori D. “Dimensioning Analysis. Toward Automatic Understanding of Engineering Drawings”. Communications of the ACM, vol. 35 No. 10, pp. 92-103, (1992).
- [Dori95] Dori D. and Tombre K. “From engineering drawings to 3D CAD models: are we ready now?”. Computer Aided Desing, vol. 27, No. 4, pp. 243-254, (1995).
- [Fole96] Foley, J.D., Van Dam, A., Feiner, S.K., Hughes, J.F. y Phillips R.L. “Introducción a la graficación por computador”. Addison_Wesley Iberoamericana, S.A. ISBN 0-201-62599-7.
- [Gomi96] Gomis J.M. y Company P. “Reconstrucción geométrica tridimensional”. Anales de Ingeniería Gráfica, vol. 2, pags. 26-34, (1996).
- [Gomi97(1)] Gomis J.M., Company P. y Contero, M. “Reconstrucción de modelos poliédricos a partir de sus vistas normalizadas”. Anales de Ingeniería Mecánica, Año 11, vol. 1, pags. 383-391, (1997).
- [Gomi97(2)] Gomis J.M., Company P. y García J. “Preprocesador para modelado geométrico tridimensional a partir de la delineación 2D de axonometrías”. IX Congreso Internacional de Expresión Gráfica en la Ingeniería, vol. 2, pags. 345-354, (1997).
- [Gomi98] Gomis, J.M., Alcañiz, M., Díaz del Río, C., Herreras, M.A. y Landete, M. “La vectorización y la gestión de documentos en los sistemas CAD”. X Congreso Internacional de Ingeniería Gráfica, vol. I, pp.565-582, (1998).
- [Gomi99(1)] Gomis J.M., Company P. y Gil M.A. “Vectorization in Recovering Engineering Drawings”. Progettazione e fattibilità dei prodotti industriali, Diseno y fabricabilidad de los productos industriales, (a cura di) Lanzotti A. e Patalano S. Giannini Editore (ISBN 88-900081-3-X), Napoli,, pp 3-12, (1999).
- [Gomi99(2)] Gomis J.M. y Capuz, S. “Reconstrucción de Modelos Sólidos 3D a partir de dibujos de ingeniería”. XV Congreso Nacional de Ingeniería de Proyectos. Ed. Asociación Española de Ingeniería de Proyectos, Universidad de León, vol I, pp.595-604 (1999).
- [Grim95] Grimstead I.J. and Martin R.R. “Creating solid models from single 2D sketches”. Proc. Third Symp. On Solid Modeling Applications, ACM Siggraph, pp. 323-337, (1995).
- [Grim96] Grimstead I.J. and Martin R.R. “Incremental Line Labelling for Sketch Input of Solid Models”. Computer Graphics Forum, vol. 15, No. 2, pp. 155-166, (1996).

- [Gu86] Gu K., Tang Z. and Sun J. "Reconstruction of 3D Objects from Orthographic Projections". *Computer Graphics Forum*, vol. 5, pp. 317-324, (1986).
- [Guja89] Gujar U.G. and Nagendra I.V. "Construction of 3D solid objects from ortographic views". *Computer & Graphics*, vol. 13, No. 4, pp. 505-521, (1989).
- [Gusp00] Guspí, F. y Introcaso, B. "Soluciones ralas de sistemas lineales indeterminados". Departamento de Matemática FCEIA y Grupo Geofísica IFIR.
http://www.fceia.unr.edu.ar/fceia1/publicaciones/numero8/articulo1/soluciones_ralas_de_sistemas_lin.htm
- [Guzm] <http://www.cic.ipn.mx/~aguzman/>
- [Hoff00] Hoffman D. "Inteligencia Visual. Cómo creamos lo que vemos". Ed. Paidós Ibérica S.A, ISBN: 84-493-0918-2 (2000).
- [Huff71] Huffman, D. A. "Impossible objects as nonsense sentences" *Machine Intelligence* Edinburgh University Press, pp. 295-323 (1971).
- [Ides73] Idesawa, M. "A system to generate a solid figure from three views", *Bull. JSME*, vol 16, pp 216-225 (1973).
- [Kirk83] Kirkpatrick s., Gelatt C.D. and Vecchi M.P., "Optimization by simulated annealing". *Science*, vol. 220, no. 4598, pp. 671-680, (1983).
- [Kouv92] Kouvelis, P., Chiang W. And Fitzsimmons J. "Simulated annealing for machine layout problems in the presence of zoning constraints". *European Journal of Operational Research*, vol. 57, pp. 203-223, (1992).
- [Kuo98] Kuo M. H. "Reconstruction of quadric surface solids from three-view engineering drawings". *Computer Aided Desing*, vol. 30, No. 7, pp. 517-527, (1998).
- [Kuzo99] Kuzo, P. And Mace, P. "Correction of a 2D sketch for 3D reconstruction" WSCG99 The 7-th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media'99, February 8 - 12, 1999
 University of West Bohemia, Plzen, Czech Republic. Comunicación publicada: WSCG'99 Conference proceedings, edited by Vaclav Skala ISBN 80-7082-490-5, vol. II, 1999, pp. 148-155.
- [Lamb90] Lamb, D. & Bandopadhyay, A. " Interpreting a 3D object from a rough 2D line drawing". In *Proceedings of Visualization'90*, pp.59-66, (1990).
- [Lecl92] Leclerc Y. and Fischler M. "An optimization -based approach to the interpretation of single line drawings as 3D wire frames". *International Journal of Computer Vision*, vol. 9, No. 2, pp. 113-136, (1992).
- [Lips95] Lipson H. and Shpitalni M. "A new interface of conceptual design based on object reconstruction from a single freehand sketch". *Annals of the CIRP*, Vol 44, No. 1, pp. 133-136, (1995).
- [Lips96] Lipson H and Shpitalni M. "Optimization-based reconstruction of a 3D object from a single freehand line drawing". *Computer Aided Design*, vol. 28, No. 8, pp. 651-663, (1996).

- [Mack73] Mackworth, A. K. "Interpreting pictures of polyhedral scenes". *Artificial Intelligence*, vol 4, pp. 121-137, (1973).
- [Mali87] Malik, J. "Interpreting line drawing of curved objects". *International Journal of Computer Vision*. vol 1, pp.73-103 (1987).
- [Mari91] Marill T. "Emulating the Human Interpretation of Line-Drawings as Three-Dimensional Objects". *International Journal of Computer Vision*, vol. 6, No. 2, pp. 147-161, (1991).
- [Mart93] Martí E., Regincós J., López_Krahe J. and Villanueva J.J. "Hand line drawing interpretation as three-dimensional objects". *Signal Processing*, vol. 32, pp. 91-110, (1993).
- [Masu97] Masuda H. and Numao M. "A cell-based approach for generating solid objects from orthographics projections". *Computer Aided Desing*, vol. 29, No. 3, pp. 177-187, (1997).
- [Meer93] Meeran S. and Pratt M.J. "Automated feature recognition from 2D drawings". *Computer Aided Desing*, vol. 25, No. 1, pp. 7-17, (1993).
- [Metro53] Metropolis N., Rosenbluth A.W. Rosenbluth M.N. and Teller A.H., "Equation of state claculations by fast computing machines". *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087-1091, 1953.
- [Nage88] Nagendra I.V. and Gujar U.G.. "3-D Objects From 2-D Orthographic View –A Survey". *Computer & Graphics*, vol. 12, No. 1, pp. 111-114, (1988).
- [Naya01] Naya, F., Contero, M. y Gomis, J.M. "Parámetros de trazado en la ejecución de dibujos a mano alzada en el entorno de sistemas CAD". *XIII Congreso Internacional de Ingeniería Gráfica*", pp. 26 (2001).
- [Open] <http://www.opengl.com>
- [Prei84] Preiss K. "Constructing the solid representation from engineering projections". *Computer & Graphics*, vol. 8, No. 4, pp. 381-389, (1984).
- [Pres88] Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T. "Numerical Recipes in C". Cambridge University Press, 1988 (reprinted 1991).
- [Qin00] Qin, S.F, Wright, D.K, and Jordanov, I.N. "From on-line sketching to 2D and 3D geometry: a system based on fuzzy knowledge". *Computer-Aided Design*, vol 32, pp 851-866, (2000).
- [REGE00] <http://www.tec.uji.es/regeo>, 2000.
- [Requ92] Requicha A.A.G. and Rossignac J.R. "Solid Modeling and Beyond". *IEEE Computer Graphics & Applications*, September, pp. 31-44, (1992).
- [Robe65] Robert, L.G. "Chapter 9: Machine Perception of three-dimensional solids". *Optical and Electro-Optical Information processing*. The MIT Press, Cambridge, Massachusetts and London, England. (1965).
- [Shin98] Shin B.S. and Shin Y.G. "Fast 3D solid model reconstruction from orthographic views". *Computer Aided Desing*, vol. 30, No. 1, pp. 63-76, (1998).

- [Sphi96] Sphitalni M. and Lipson H. "Identification of faces in a 2D line drawing projection of a wireframe object". IEEE transactions on pattern analysis and machine Intelligence, vol. 18, No. 10, pp. 1000-1012, (1996).
- [Sphi97] Sphitalni M. and Lipson H. "Classification of sketch strokes and corner detection using conic sections and adaptive clustering". Journal of Mechanical Desing, vol. 119, No. 1, pp. 131-135, (1997).
- [Sugi] <http://www.simplex.t.u-tokyo.ac.jp/~sugihara/Welcomee.html>
- [Sugi82] Sugihara, K. "Mathematical structures of line drawing of polyhedrons – Towards Man- Machine communication by means of line drawings". IEEE Transactions on Patten Analysis and Machine Intelligence, vol PAMI-4, pp.458-469 (1982).
- [Sugi86] Sugihara, K. "Machine interpretation of line drawing". MIT Press, (1986).
- [Thor83] Thorpe C. and Shafer S "Correspondence in line drawings of multiple views of objects". Proceedings of the Eighth International Joint Conference on Artificial Intelligence, vol. 2, (1983).
- [Varl00(1)] Varley, P and Martin R., "A system for constructing boundary representation solid models from a two-dimensional sketch. Frontal geometry ans sketch categorisation". 1st Korea_UK Joint Workshop on Geometric Modeling and Computer Graphics, (2000)
- [Varl00(2)] Varley, P and Martin R., "TA system for constructing boundary representation solid models from a two-dimensional sketch. Topology of hidden parts". 1st Korea_UK Joint Workshop on Geometric Modeling and Computer Graphics, (2000)
- [Varl00(3)] Varley, P and Martin R., "A system for constructing boundary representation solid models from a two-dimensional sketch. Geometric finish Fronta". 1st Korea_UK Joint Workshop on Geometric Modeling and Computer Graphics, (2000)
- [Walt] <http://www.neci.nj.nec.com/homepages/waltz/>
- [Wang93] Wang W. and Grinstein G. "A Survey of 3D Solid Reconstruction from 2D Projection Line Drawings". Computer Graphics Forum, vol. 12, No. 2, pp. 137-158, (1993).
- [Wesl81] Wesley, M.A. and Markowsky, G. "Fleshing out projections". IBM Journal of Research and Development, vol. 25, No. 6 (1981).
- [Yan94] Yan Q., Chen C.L. and Tang Z. "Efficient algorithm for the reconstruction of 3D objects from orthographic projections". Computer-Aided Design, vol. 26, No. 9, pp. 699-717, (1994).

En este epígrafe se presentan las herramientas de búsqueda bibliográfica utilizadas. En el capítulo 9 de esta tesis se muestran las referencias que hemos podido encontrar y a partir de las cuales se ha redactado el estado del arte.

Nuestro trabajo se inicia a partir de unas pocas referencias de algunos autores que trabajaron en la Reconstrucción Geométrica. A partir de ellos hemos intentado encontrar las referencias que se hacían en sus trabajos consultando inicialmente las hemerotecas de nuestras Universidades y las hemerotecas virtuales de Internet, obteniendo así un abanico más amplio de referencias

Pero sin duda alguna la herramienta fundamental utilizada en nuestra búsqueda ha sido Internet. La razón es obvia si tenemos en cuenta que esta herramienta presenta innumerables ventajas frente a las hemerotecas tradicionales, permitiendo realizar nuestras búsquedas bibliográficas con agilidad y comodidad al tiempo que permite realizar búsquedas tan extensas como sean viables en un tiempo razonable. Además, el espectacular crecimiento que ha experimentado Internet en los últimos años ha animado a muchos investigadores y centros de investigación a publicar información muy variada lo que, junto con el desarrollo de los “buscadores” de información, ha dado a lugar el estableciendo de un nuevo sistema de consulta de documentos que permite difundir y compartir información valiosa y más variada y amplia que la que tradicionalmente se publicaba en las revistas.

DISTRIBUIDORAS DE REVISTAS

La consulta de distribuidoras de revistas a las que estamos suscritos ha sido el punto de partida de nuestra búsqueda. De entre las revistas en soporte papel a las que esta suscrito el grupo de investigación en el que se ha desarrollado la tesis cabe destacar:

- Computer graphics SIGGRAPH / Association for Computing Machinery. New York, N.Y, Association for Computing Machinery, 1987 ISSN 00978930

- Computers and graphics. New York, N.Y, Pergamon Press, 1987 ISSN 00978493
- IEEE computer graphics and applications / Institute of Electrical and Electronics Engineers. Los Alamitos, Califo: Institute of Electrical and Electronics Engineers, 1990 ISSN 02721716
- IEEE Transactions on visualization and computer graphics / Institute of Electrical and Electronics Engineers. New York: Institute of Electrical and Electronics Engineers, 1990 ISSN 10772626
- Journal of Engineering Design. Abingdon: Carfax Publishing Company, 1990 ISSN 09544828
- Research in Engineering Design. London: Springer, 1989. ISSN 09349839

Además existen otras revistas a las que el grupo esta suscrito mediante acceso electrónico. De entre ellas queremos destacar:

- <http://www-sp.ebsco.com/online/disclaimer.asp>
- <http://alidoro.catchword.com>
- <http://www.sciencedirect.com>
- <http://www.idealibrary.com>

A dichas revistas hay que añadir las que tenemos acceso directo a través de la UPV en <http://www.upv.es/revistas> de entre las que queremos destacamos por su vinculación no nuestra disciplina:

- Journal of Engineering desing
- Graphical Models

No obstante, también han sido revisadas otras distribuidoras de revistas a pesar de no estar suscritos a ellas, dado que tales consultas pueden ser interesantes para descubrir referencias que luego se pueden obtener por intercambio bibliotecario. En efecto, a pesar de no estar permitido el acceso al contenido de sus artículos, es posible consultar títulos y “abstract” sobre su contenido, sirviendo por tanto como criterio para solicitar su petición. Algunas de estas distribuidoras de revistas que nos han resultado interesantes son:

- <http://www.emis.de>

- <http://ijdar.cfar.umd.edu/>
- <http://mitpress.mit.edu>
- <http://www.bids.ac.uk>
- <http://www.wkap.nl>
- <http://elib.cs.sfu.ca>

BUSCADORES

La búsqueda de información que ofrece Internet ha sido el segundo paso a seguir en la búsqueda de información bibliográfica.

Como primera conclusión de las búsquedas realizadas hay que resaltar que no todos los buscadores resultan válidos para la bibliografía científica y tecnológica. De hecho, la mayoría de los buscadores conocidos comúnmente a través de la publicidad tienen una orientación más propiamente comercial que académica o científica. La elección de un buscador acorde con nuestro trabajo debe hacerse a partir de la experimentación, es decir, mediante la búsqueda sistemática con cada uno de ellos y la comparación de los resultados obtenidos.

Algunos criterios que hemos seguido para seleccionar los buscadores más apropiados han sido la búsqueda de referencias que ya son conocidas (e incluso la búsqueda de nuestro propio grupo de investigaciones y nuestras publicaciones), y el cruce y comparación, a posteriori, de las referencias encontradas por cada buscador.

Para realizar las pruebas hay que repetir una misma búsqueda en diferentes buscadores y con distintos criterios. Para ello es interesante aprovechar los “multibuscadores”, es decir buscadores cuya ejecución activa varios buscadores simultáneamente. El que se utilizó para realizar búsquedas para esta tesis es:

- <http://www.paisvirtual.com/informatica/navegadores/toniomos/multibuscador.html>

Tras las pruebas realizadas, los principales buscadores de orientación específicamente científica que han permitido documentarnos para la redacción de esta tesis son:

- www.webcrawler.com
- <http://citeseer.nj.nec.com/cs>

Junto a dichos buscadores debemos unir otro que, a pesar de su orientación más generalista, no deja de ser menos importante:

- <http://www.alltheweb.com/>

BUSCADORES DE BUSCADORES

El primer problema que se nos planteo a la hora de utilizar buscadores de Internet fue evidentemente el conocer la dirección web de los propios buscadores. Es por ello que en este apartado resaltamos una dirección web de los que son conocidos como “buscadores de buscadores”, es decir, páginas que permiten localizar las direcciones de distintos buscadores.

En <http://www.servicenet.com.ar/esalom/buscador/paises.htm#Ind-Hispania> se encuentra una web que muestra un listado de buscadores agrupados por países.

OTRAS REFERENCIAS

Otras referencias que han sido consultadas las constituyen las Web creadas por algunas asociaciones de gran prestigio. Conocer las direcciones es un proceso azaroso e intuitivo. Se puede localizar el sitio web de una institución ya conocida (utilizando un buscador) o se puede utilizar el buscador para detectar organismos que trabajan en temas relacionados con las palabras clave utilizadas para buscar. De entre las referencias interesantes encontradas siguiendo estos mecanismos de búsqueda cabe resaltar:

- <http://computer.org/> (IEEE)
- <http://www.acm.org/> (ACM)
- <http://www.ai.mit.edu/publications/pubsDB/pubs-index.html> (MIT)
- <http://www.isinet.com/isi/> (ISI).

Otros páginas que han resultado de interés en estas búsquedas han sido las web de grupos de investigación que trabajan en áreas afines a esta tesis. Entre ellas cabe destacar:

- <http://www-cg-hci-g.informatik.uni-oldenburg.de/home.html>

- <http://www.isata.com/programs/robotic.htm>
- http://www.emn.fr/fran/recherche/dept_info/perso/kuzo/CVMOI.html
- http://www.emn.fr/fran/recherche/dept_info/recherche/equipes/image/Themes/CODIS/CODIS.html<http://www.emn.fr/colin/graphicon/Presentation/Welcome.html>
- <http://www.cic.ipn.mx/~aguzman/>
- <http://www.simplex.t.u-tokyo.ac.jp/~sugihara/Welcomer.html>
- <http://www.neci.nj.nec.com/homepages/waltz/>

PALABRAS CLAVE

Las “palabras claves”, es decir, los términos utilizados para la realización de búsquedas constituyen otro de los conceptos a tener en cuenta para realizar una búsqueda bibliográfica.

Es práctica habitual utilizar como palabras claves términos que describen el problema fundamental de la búsqueda. En nuestro caso Reconstrucción. Sin embargo, los términos genéricos suelen venir asociados a distintos ámbitos, y evidentemente no todos ellos resultan acordes con nuestro tema. Por consiguiente se precisa de una combinación de términos que describan de una manera más independiente el problema que nos ocupa.

Por otro lado, la mayoría de los buscadores de Internet presentan distintas posibilidades de realizar una determinada búsqueda. Así por ejemplo permiten posibilidades de realizar búsquedas por títulos de publicaciones, por “abstract” o por contenido de publicación.

A partir de la experiencia que hemos obtenido de las búsquedas realizadas debemos resaltar que si bien, la búsqueda de palabras claves en títulos de publicaciones resulta la más directa y sencilla para encontrar artículos que resulten interesantes para nuestra investigación, la búsqueda por “abstract” o por contenido de la publicación es mejor. La razón es que, si bien en la mayoría de los casos no nos relaciona directamente como publicaciones concretas de nuestro tema, si permiten seleccionar publicaciones en las que se hace referencia a artículos de interés.

En nuestro caso y tal y como se ha descrito con antelación, nuestra búsqueda inicial fue realizada utilizando Reconstrucción como palabra clave. A partir de dicha

palabra clave es cierto que es posible obtener referencias de interés, pero el número de registros encontrados en las diferentes bases de datos resulta inviable para su verificación. Como consecuencia se ha tratado de realizar un filtrado de dichos registros. A continuación se redactan las combinaciones de palabras claves que han sido utilizadas para filtrar el elevado número de registro encontrados. Cabe destacar que la búsqueda se ha realizado en inglés, asumiendo que la mayor parte de la documentación relacionada con el tema estudiado puede encontrarse en dicho idioma:

- Reconstruction algorithm
- Reconstruction projection
- Reconstruction freehand

Junto a dichas combinaciones que guardan relación con el problema fundamental de nuestro tema de investigación, deben ser analizados otro conjunto de términos que comúnmente son utilizados paralelamente al tema de la Reconstrucción. De entre ellos, se relacionan las combinaciones que han permitido obtener publicaciones válidas:

- Optimization drawing
- Drawing Design
- Interpretation drawing
- Line Drawing
- Recognition 3D
- Interface reconstruction

PUBLICACIONES AFINES AL TEMA DE LA RECONSTRUCCIÓN

A continuación se relaciona el conjunto de publicaciones que han sido obtenidas siguiendo las pautas anteriormente referidas. Observando la relación de las siguientes referencias con el objeto de esta tesis, se puede valorar la importancia de la búsqueda realizada para contribuir a obtener el estado del arte del tema esta tesis.

Base de datos CARL

Yan, Q-W, Chen, C.L.P. and Tang, Z, "Efficient algorithm for the reconstruction of 3D objects from orthographic projections". Journal Info Computer aided design. SEP 01 1994 v 26 n 9 <<http://uncweb.carl.org:80/cgi-bin/fullRecord?15707+53+-879278662>>

Shimaya, A. "Interpreting non-3-D line drawing." <<http://uncweb.carl.org:80/cgi-bin/fullRecord?15707+14+-879279582>>

Shimaya, A. "Interpreting non-3-D line drawing". *Jornal Info. Artificial intelligence*. AUG 01 1995 v 77 n 1 <<http://uncweb.carl.org:80/cgi-bin/fullRecord?15707+61+-879279582>>

Chua, Chin Seng and Jarvis, R. "Point Signatures: A New Representation for 3D Object Recognition". *Journal Info. International journal of computer vision*. OCT 01 1997 v 25 n 1 <<http://uncweb.carl.org:80/cgi-bin/resultsScreen?19541+1+879336723>>

McCane, B. Caelli, T. and de Vel, O. "Learning to Recognize 3D Objects Using Sparse Depth and Intensity Information". *Journal Info. International journal of pattern recognition*. SEP 01 1997 v 11 n 6 <<http://uncweb.carl.org:80/cgi-bin/resultsScreen?19541+1+879336723>>

Wang, W. and Grinstein, G. "A Survey of 3D Solid Reconstruction from 2D Projection Line Drawings". *Journal Info. Computer graphics forum: journal of the Europe*. JUN 01 1993 v 12 n 2. <<http://uncweb.carl.org:80/cgi-bin/displayMore?19541+21+31+879337301>>

Marill, T. "Emulating the Human Interpretation of Line-Drawings as Three-Dimensional Objects". *Journal Info. International journal of computer vision*. JUN 01 1991 v 6 n 2 <<http://uncweb.carl.org:80/cgi-bin/resultsScreen?4297+1+879347158>>

Bannour, T. Abda, A. Ben and Jaoua, M. "A semi-explicit algorithm for the reconstruction of 3D planar cracks". *Journal Info. Inverse problems*. AUG 01 1997 v 13 n 4 <<http://uncweb.carl.org:80/cgi-bin/fullRecord?20081+1+-879354955>>

Novitski, B. "Sketch modeling" *Journal Info. Computer graphics world*. MAY 01 1993 v 16 n 5. <<http://uncweb.carl.org:80/cgi-bin/fullRecord?20081+81+-879355354>>

Vosniakos, G. C. "Knowledge-based interpretation of CAD-drawing annotation for mechanical-engineering components". *Journal Info. Computer aided design*. OCT 01 1992 v 24 n 10 <<http://uncweb.carl.org:80/cgi-bin/fullRecord?20081+4+-879355702>>

Base de datos ICV

Lipson, H. and Shpitalni, M. "Optimization-Based Reconstruction of a 3D Object from a Single Freehand Line Drawing, CAD(28), No. 8, August 1996, pp. 651-663. BibRef. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Shpitalni, M. and Lipson, H. "Identification of Faces in a 2D Line Drawing Projection of a Wireframe Object" PAMI(18), No. 10, October 1996, pp. 1000-1012. 9611 Find the edge circuits that correspond to actual object faces. BibRef. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Marill, T., "Emulating the Human Interpretation of Line-Drawings as Three-Dimensional Objects". IJCV(6), No. 2, June 1991, pp. 147-161. BibRef 9106. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Marill, T. "The Three-Dimensional Interpretation of a Class of Simple Line-Drawings". MIT AIMemo 1555, October 1995. For text of paper. BibRef 9510. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Marill, T. "Recognizing Three-Dimensional Objects without the Use of Models". MIT AIMemo 1157, September 1989. BibRef 8909. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Marill, T. "Computer Perception of Three-Dimensional Objects". MIT AIMemo 1136, August 1989. BibRef 8908. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Malik, J. "Interpreting Line Drawings of Curved Objects". IJCV(1), No. 1, 1987, pp. 73-103. BibRef 8700. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Gu, W.K. and Huang, T.S. "Connected Line Drawing Extraction from a Perspective View of a Polyhedron". PAMI(7), No. 4, July 1985, pp. 422-430. BibRef 8507 CAIA84(192-198). <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Heyden, A. "On the Consistency of Line-Drawings, Obtained by Projections of Piecewise Planar Objects" JMIV(6), No. 4, December 1996, pp. 393-412. 9701 BibRef. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Huang, C.L. "Polyhedral Objects Identification Through the Orthographic Projection Views Generation". PRL(10), 1989, pp. 321-327. BibRef 8900.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Fischler, M.A., and Leclerc, Y.G. "Recovering 3-D Wire Frames from Line Drawings". DARPA92(635-653). Seems dated. BibRef 9200.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Thorpe, C.E. and Shafer, S.A. "Topological Correspondence in Line Drawings of Multiple Views of Objects". CMU-CS-TR-83-113, CMU CS Dept., March 1983. BibRef8303
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Draper, S.W. "The Use of Gradient and Dual Space in Line-Drawing Interpretation". AI(17), 1981, pp. 461-508. BibRef 8100.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>

Shafer, S.A, Kanade, T. and Kender, J.R. "Gradient Space under Orthography and Perspective". CVGIP(24), No. 2, November 1983, pp. 182-199. BibRef 8311 CVWS82(26-34).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>

Ishikawa, S. "Reconstructing Faces On A Polyhedron From Apparent Gradients of Edges". CVGIP(28), No. 3, December 1984, pp. 289-302. BibRef 8412.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>

Sugihara, K. "Machine Interpretation of Line Drawings". Cambridge, MA: MIT Press 1986. BibRef 8600 Book.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>

Sugihara, K., "A Necessary and Sufficient Condition for a Picture to Represent a Polyhedral Scene". PAMI(6), No. 5, September 1984, pp. 578-586. BibRef 8409. <<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>

Sugihara, K. "Range Data Analysis Guided by a Junction Dictionary" AI(12), No. 1, 1979, pp. 41-69. BibRef 7900.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>

Sugihara, K. and Shirai, Y. "Range Data Understanding Guided by a Junction Dictionary". IJCAI77(706). BibRef 7700.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>

- Sugihara, K., "Quantitative Analysis of Line Drawings of Polyhedral Scenes". ICPR78(771-773). BibRef 7800.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>
- Sugihara, K. "Automatic Construction of Junction Dictionaries and Their Exploitation of the Analysis of Range Data". IJCAI79(859-864). BibRef 7900.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>
- Gigus, Z., and Malik, J. "Computing the Aspect Graph for Line Drawings of Polyhedral Objects". PAMI(12), No. 2, February 1990, pp. 113-122. BibRef 9002 CVPR88(654-661).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom211.html>>
- Weiss, I. "3-D Shape Representation by Contours" CVGIP(41), No. 1, January 1988, pp. 80-100. BibRef 8801 IJCAI85(969-972).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom212.html>>
- Malik, J. and Binford, T.O. "A Theory of Line Drawing Interpretation" DARPA84(188-194). Related to the preceding papers. BibRef 8400.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom212.html>>
- Kanade, T. "Recovery of the Three-Dimensional Shape of an Object from a Single View". AI(17), 1981, pp. 409-460. BibRef 8100.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom212.html>>
- Kanatani, K.I. "Reconstruction of Consistent Shape from Inconsistent Data: Optimization of 2.5D Sketches". IJCV(3), No. 4, November, 1989, pp. 261-292. BibRef 8911 ICCV88(508-512).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom212.html>>
- DeFloriani, L. "Feature Extraction for Boundary Models of Three-Dimensional Objects" PAMI(11), No. 8, August 1989, pp. 785-798. BibRef 8908.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom212.html>>
- Leclerc, Y.G. and Fischler, M.A. "An Optimization-Based Approach to the Interpretation of Single Line Drawings as 3D Wire Frames". IJCV(9), No. 2, November 1992, pp. 113-136. BibRef 9211.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>
- Shomar, W.J., Young, T.Y., Three-Dimensional Shape Recovery from Line Drawings, HPRIP-CV94(53-100). BibRef 9400.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom210.html#UU7340>>

Ulpinar, F. "Perception of 3-D Shape from 2-D Image of Contours" Ph. D.Thesis (EE-CE), August 1991, BibRef 9108 USC IRISTR-278.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom213.html>>

Ulpinar, F., and Nevatia, R. "Constraints for Interpretation of Line Drawings under Perspective Projection". CVGIP(53), No. 1, January 1991, pp. 88-96. BibRef 9101 DARPA89(882-892).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom213.html>>

Barnard, S.T. "Interpreting Perspective Images" AI(21), No. 4, November 1983, pp. 435-462. BibRef 8311 DARPA82(193-203). (SRI work).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom214.html>>

Barnard, S.T. "Interpreting Perspective Images" AI(21), No. 4, November 1983, pp. 435-462. BibRef 8311 DARPA82(193-203). (SRI work).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom214.html>>

Barnard, S.T. "Pentland, A.P., "Three-Dimensional Shape from Line Drawings" DARPA83(282-284). BibRef 8300 IJCAI83(1062-1064).
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom214.html>>

Lee, S.J., Haralick, R.M., and Zhang, M.C., "Understanding Objects with Curved Surfaces from a Single Perspective View of Boundaries". AI(26), 1985, pp. 145-169. BibRef 8500.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom214.html>>

Haralick, R.M., Chu, Y.H., Watson, L.T. and Shapiro, L.G. "Matching Wire Frame Objects from Their Two Dimensional Perspective Projections" PR(17), 1984, pp. 607-619. BibRef 8400.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom214.html>>

Brillault-O'Mahony, B. "High Level 3D Structures from a Single View". IVC(10), 1992, pp. 508-520. BibRef 9200.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/shapefrom214.html>>

Aldelfeld, B., Automatic 3D Reconstruction from 2D geometric Part Descriptions, CVPR83(66-72). BibRef 8300.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/author/ade.html#Adelfeld, B.>>

Hebert, M., Ponce, J., A New Method for Segmenting 3-D Scenes into Primitives, ICPR82(836-838). BibRef 8200.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/author/heb.html#Hebert, M.>>

Aldefeld, B. "On Automatic Recognition of 3D Structures from 2D Representations" *CAD(15)*, No. 2, 1983, pp. 59-64. BibRef 8300 CVPR83(66-72). <<http://www.icv.ac.il/DataBases/biblio/bibliography/char514.html>>

Marti, E., Regincos, J., Lopez-Krahe, J., and Villanueva, J.J., "Hand Line Drawing Interpretation as Three-Dimensional Object". *SP(32)*, 1993, pp. 91-110. BibRef 9300.
<<http://www.icv.ac.il/DataBases/biblio/bibliography/char514.html>>

El primer autor que utilizó la optimización como estrategia para la Reconstrucción Geométrica de Sólidos fue Marill [Mari91]. El algoritmo propuesto por este autor, “Hill-Climbing”, es un algoritmo de descenso que utiliza la disminución del coste de la función objetivo como único criterio para guiar el proceso de optimización, lo que le hace muy propenso a la obtención de mínimos locales.

Además explora muchas soluciones, porque la elección de la dirección de búsqueda es exhaustiva; por lo que dicho algoritmo es únicamente válido si no se precisan grandes tareas computacionales [Pres88]. Sin embargo cuando aumenta la dificultad de los problemas (esto es, cuando aumenta la complejidad del objeto a reconstruir), la búsqueda de un algoritmo eficiente constituye un objetivo fundamental para la reconstrucción.

Baird y Wang [Bair91] analizaron un nuevo algoritmo para la optimización de la función objetivo propuesta por Marill. Su algoritmo aprovechaba la información de las derivadas, concretamente se basaba en el análisis del gradiente, siendo la disminución de la función objetivo el único criterio que guiaba la optimización. Sin embargo, si bien el nuevo algoritmo conseguía una notable reducción del coste computacional, en tanto que todas las variables de reconstrucción eran modificadas en cada ciclo del algoritmo, seguían manteniéndose problemas de convergencia hacia mínimos globales

Posteriormente, Leclerc y Fischler [Lec192] implementaron y estudiaron el comportamiento del algoritmo del “Gradiente Conjugado”. Se trata de un algoritmo de optimización de búsqueda guiada de primer orden, por lo que precisa de la derivabilidad de la función objetivo.

Pero además, si el gradiente debe obtenerse numéricamente este método puede requerir mucho tiempo de computación, y si en alguna etapa los componentes del vector gradiente de la función objetivo son de magnitud muy distinta puede perderse la convergencia debido a problemas numéricos. Hay que resaltar que se trata de un método que avanza muy rápidamente hacia al solución al principio del proceso de optimización, pero luego tienen una convergencia muy lenta, e incluso en ocasiones no converge.

La función objetivo es definida por estos autores en la forma:

$$E(\lambda) = \lambda \text{SDA}^2 + (1-\lambda) \text{DP} \quad (76)$$

donde el término "DP", que denominan desviación de planitud, mide la desviación de los vértices del modelo obtenido respecto a las caras planas a las que el algoritmo supone que deben pertenecer, SDA hace referencia a la desviación de ángulos y λ es un parámetro que varía desde 0 a 1, puede adivinarse una clara intención en cada uno de sus términos.

El término desviación de ángulos constituye el único componente de la función objetivo en las etapas iniciales de la optimización ($\lambda = 0$), donde el diseño que esta constituido por la imagen de partida viola claramente dicho criterio. Posteriormente conforme avanza el proceso de optimización, el parámetro λ reduce el peso de este término hasta que en las etapas finales de la optimización la función objetivo esta constituida exclusivamente por la desviación de planitud.

De este análisis puede deducirse que el término desviación de ángulos es propuesto como una estrategia para escapar del óptimo trivial de la imagen, mientras que la desviación de planitud constituye la única regularidad real que define el modelo final.

Mediante este algoritmo se reducen en gran medida los inconvenientes de "Steepest Descent" mediante la técnica propuesta por Fletcher y Reeves. En ella las búsquedas se orientan en la forma:

$$d_K = -\nabla F + \frac{[\nabla F_K]^T [\nabla F_K]}{[\nabla F_{K-1}]^T [\nabla F_{K-1}]} d_{K-1} \quad (77)$$

donde como puede observarse d_K es una combinación lineal de $[d_0, d_1, d_2, \dots, d_{K-1}]$.

Esta técnica se conoce con el nombre de gradiente conjugado ya que puede demostrarse que las sucesivas direcciones que se generan de esta manera son conjugadas (figura 174). Por ello la optimización se consigue más rápidamente.

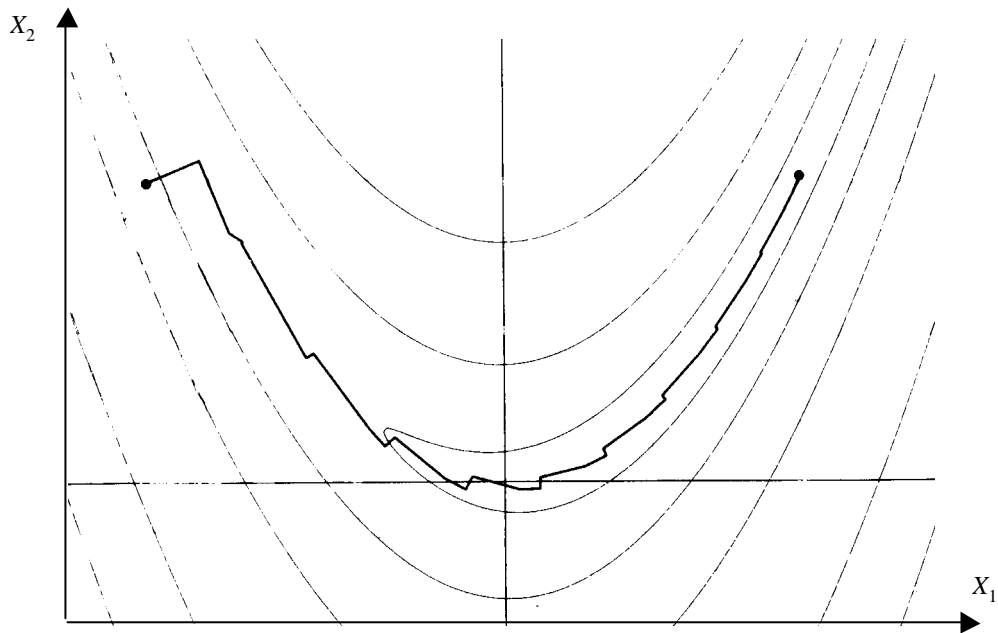


Figura 174. Optimización mediante Gradiente Conjugado

Si en alguna etapa, el método no produce disminución de la función objetivo según lo esperado, debe utilizarse la dirección $-\nabla F$ como dirección de búsqueda y reiniciar el proceso iterativo.

Nuevos algoritmos se han propuesto posteriormente. Lipson y Sphitalni [Lips96] revisaron el algoritmo del “Gradiente Conjugado” llegando a la conclusión de que se trataba de un algoritmo que convergía muy rápidamente pero que en algunas ocasiones no convergía. Además exploraron dos nuevos algoritmos: “la interpolación parabólica y un “Algoritmo Genético”.

La técnica de interpolación parabólica se basa en la determinación de mínimos en parábolas unidimensionales definidas a partir de tres puntos de la función a optimizar. La obtención de dicho mínimo parabólico se obtiene mediante la expresión:

$$X_{\min} = b - \frac{1}{2} + \frac{(b-a)^2 [f(b) - f(c)] - (b-c)^2 [f(b) - f(a)]}{(b-a) [f(b) - f(c)] - (b-c) [f(b) - f(a)]} \quad (78)$$

siendo a, b y c tres abscisas y f(a), f(b) y f(c) las coordenadas respectivas dadas por la función objetivo. En la figura 175 se muestra un ejemplo, donde a partir de los puntos 1,

2 y 3 de la función objetivo (línea continua), se determina la parábola (línea discontinua) que inicia el proceso de optimización.

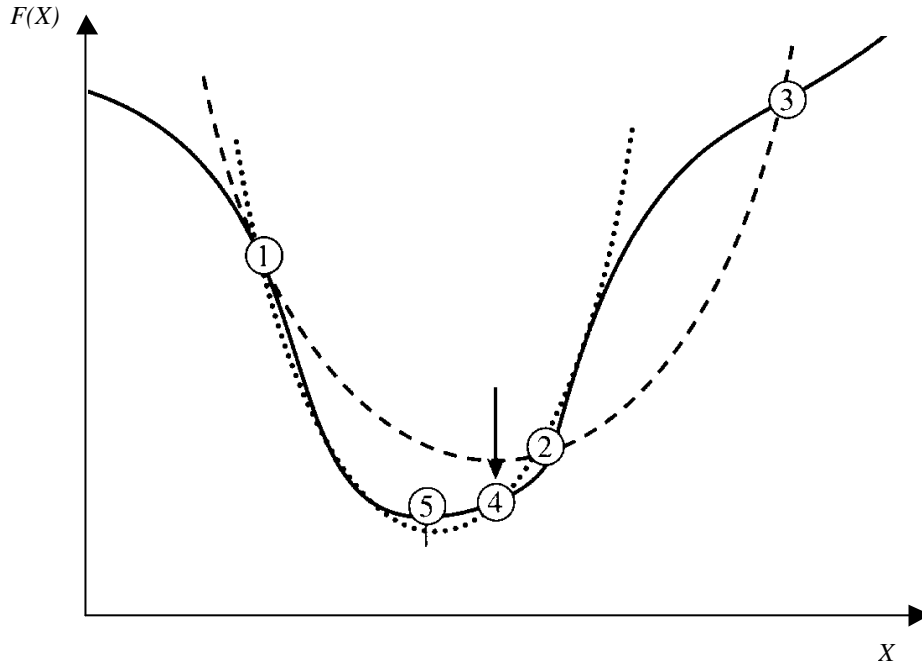


Figura 175. Interpolación parabólica.

Dicha parábola es evaluada de acuerdo con la expresión (78) que determina un mínimo en el punto 4, reemplazando al punto 3. Una nueva parábola es definida a partir de los puntos 1, 2 y 4 (línea de puntos), repitiéndose el proceso de forma iterativa.

El método expuesto es aplicado cíclicamente vértice por vértice hasta que el sistema llega a un equilibrio.

La “Interpolación Parabólica” resulta un algoritmo fácil de implementar, sin embargo, presenta un primer inconveniente en la obtención del mínimo parabólico cuando los tres puntos que definen la parábola presentan igual ordenada al evaluar la función objetivo. La técnica representa un método muy robusto pero no puede garantizarse la obtención de mínimos globales.

Por otro lado los algoritmos “Genéticos” (en adelante AG), de acuerdo con Lipson y Sphitalni es el algoritmo que ofrece una mayor garantía para conseguir el óptimo global, sin embargo presenta una gran problemática dado el elevado número de iteraciones que se precisan. Lo que supone un alto coste computacional.

Los algoritmos genéticos presentan cuatro diferencias básicas con relación a los métodos de búsqueda tradicionales:

1. Los A.G. operan codificando un conjunto de parámetros que representan un punto del espacio de soluciones. Esta codificación se realiza por medio de una cadena de caracteres, de tal forma que existe una correspondencia biunívoca entre cada solución y su representación en cadena.
2. Los A.G. buscan sobre un conjunto de puntos (trabajan con una población de soluciones), en lugar de un simple punto del espacio de soluciones.
3. Los A.G. usan una función objetivo que califica directamente la calidad de una solución dada, sin utilizar valores de las derivadas ni cualquier otro tipo de conocimientos o información auxiliar.
4. Los A.G. utilizan reglas de transición probabilísticas, no reglas deterministas.

El mecanismo de actuación de un algoritmo genético se reduce a copias de cadenas de caracteres o intercambio de subcadenas.

Su estructura básica, que se representa en la figura 176, está organizada en cuatro fases:

- Creación de una “población” de cadenas.
- Evaluación de cada cadena.
- Selección de las mejores cadenas.
- Manipulación genética para crear una nueva “población de cadenas”.

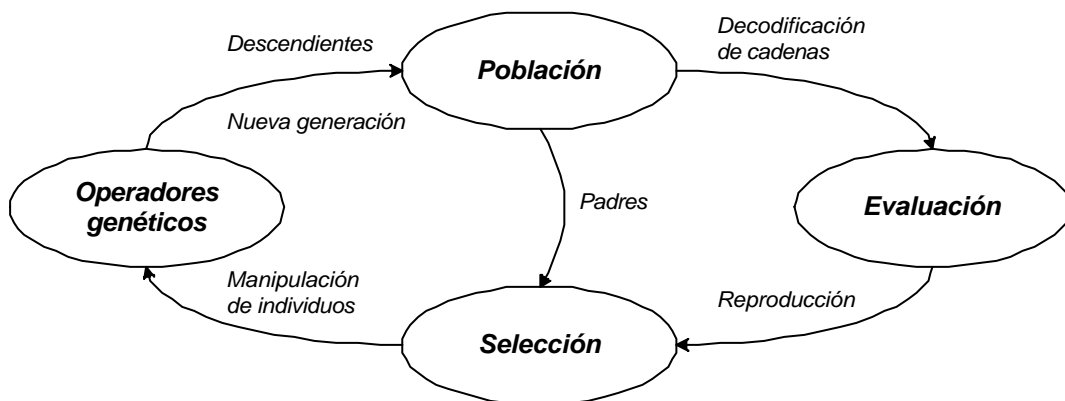
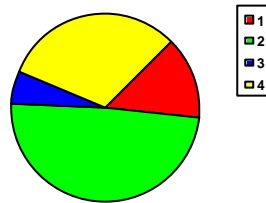


Figura 176. Estructura de un algoritmo genético

La reproducción es el proceso por el cual algunas cadenas se copian, dependiendo de los valores tomados por la función objetivo. Es decir, la evaluación de la función objetivo determina que cadenas tienen una mayor probabilidad de contribuir a la formación de una o más cadenas descendientes para formar parte de la nueva generación.

Son muchas las técnicas existentes para implementar el operador de reproducción, entre ellas destacan:

1. Método de la ruleta: se puede ilustrar como un diagrama sectorial en el que a cada sector se le asigna una amplitud proporcional a la calidad de la cadena asociada. En la figura 177 se representa la ruleta correspondiente al análisis de calidades realizado en la tabla 178.



Individuo	Cadenas	Calidad	% del total
1	0 1 1 0 1	169	14.4
2	1 1 0 0 0	576	49.2
3	0 1 0 0 0	64	5.5
4	1 0 0 1 1	361	30.9
Total		1170	100.0

Figura 177. Ruleta de un algoritmo genético

Tabla 178. Análisis de calidades de un algoritmo genético

La reproducción es simulada haciendo girar la ruleta, y se observa el sector o cadena que resulta premiado o seleccionado, pasando a ser un candidato para realizar la reproducción. De esta forma, repitiendo el giro de la ruleta se van seleccionando tantos candidatos como se desee. Es evidente que en este proceso existirán cadenas que no serán elegidas y posiblemente alguna otra podrá aparecer más de una vez. Una vez elegidos los candidatos, se crea una copia de cada uno. Las copias pasan a formar parte del grupo de candidatos a los que se aplicarán los operadores genéticos para generar la siguiente generación de la población de soluciones.

2. Método del rango: que presenta la ventaja de permitir controlar la derivación hacia la cadena de mayor calidad. Para ello se ordenan las candidatas en orden decreciente de calidad, asignado rango 1 a la cadena de mayor calidad, rango 2 a la que le sigue y así sucesivamente. Una vez establecido el rango, a la de rango 1 se le asigna una probabilidad “p” adecuada de ser seleccionada y a las siguientes se le asignan probabilidades $p_2=p(1-p)$, $p_3=p(1-p_2)$, ..., $p_n=p(1-p_{n-1})$.
3. Método de espacio de rangos: presenta la ventaja de medir la diversidad que se puede considerar como una medida de la diferencia entre distintas nociones o bloques dentro de una cadena. Una forma de medir la diversidad dentro de una generación podría ser calculando la distancia entre cada cadena y las otras ya seleccionadas. Eso no sería difícil si las cadenas están codificadas en un alfabeto binario {1,0}, permitiéndose aplicar la distancia euclídea generalizada a un espacio de tantas dimensiones como caracteres tiene la cadena.

La manipulación de individuos es el proceso que tiene lugar después de reproducción y que se conoce como recombinación. Este proceso genera las cadenas que pasarán a ser miembros de la nueva generación. El proceso se divide en dos fases:

1. Emparejamiento aleatorio de las cadenas candidatas
2. Selección de un punto de corte de estas mediante un número aleatorio del intervalo $[1, k-1]$, siendo “k” la longitud de la cadena.

Como consecuencia, dos nuevas cadenas se crean intercambiando entre sí los caracteres que ocupan las posiciones anteriores y posteriores al punto de corte. Un ejemplo del este mecanismo se muestra en la figura 179.

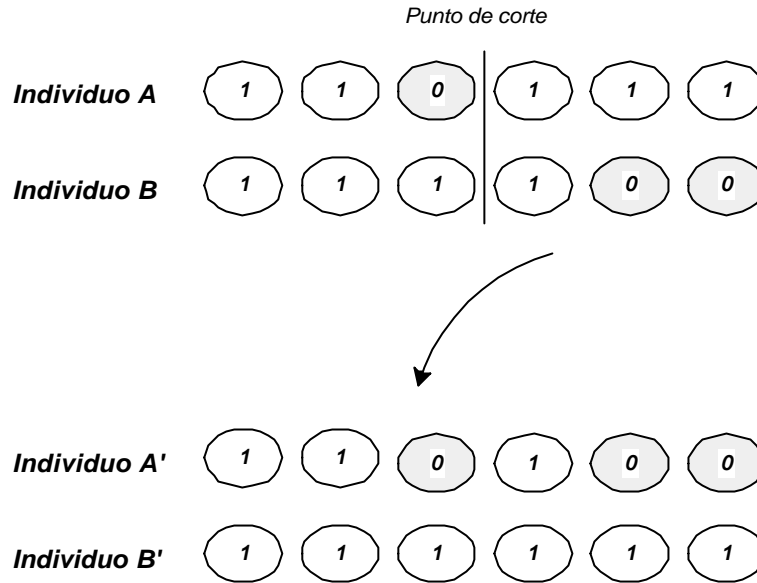


Figura 179. Recombinación de cadenas

El operador de mutación juega un papel secundario, pero no por ello deja de ser necesario. Los mecanismos de reproducción recombinação pueden extraviar parte de información que por su diversidad sería eliminada en los procesos descritos. El operador de mutación protege al algoritmo de tales situaciones a través de la introducción de nuevas estructuras en la población mediante la modificación aleatoria de algún “bloque”. Este mecanismo permite al sistema escapar de mínimos locales, y dirigir la búsqueda hacia nuevas direcciones. El operador de mutación escoge aleatoriamente un elemento de una cadena y lo sustituye por otro carácter del alfabeto empleado en la codificación.

Un ejemplo del mismo sería el mostrado en la figura 180.

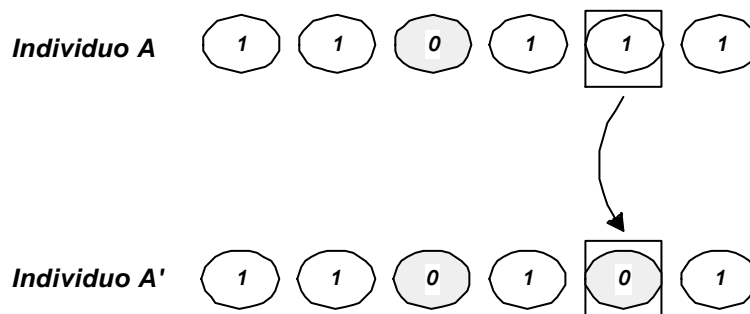


Figura 180. Mutación de cadenas

El concepto de “esquema” o “plantilla de semejanza” hace referencia al procedimiento de búsqueda para descubrir semejanzas entre aquellos individuos que presentan una mejor calidad.

Se podría definir un esquema como una plantilla que representa a un subconjunto de cadenas con semejanzas en ciertas posiciones de la cadena.

Para precisar las ideas, supóngase que las cadenas se generan a partir de un alfabeto binario $\{0,1\}$ y para que el esquema resulte más evidente añadamos un tercer símbolo, el “*” o comodín. Con estos tres símbolos se pueden crear cadenas (esquemas) cuyo significado es claro si se consideran como un patrón de coincidencias. Un esquema dado se presenta en una cadena si las posiciones de los 1 y 0 en el esquema coinciden con las posiciones homólogas en la cadena. El símbolo “*” no interviene en esta comparación. Así pues, la idea de esquema proporciona un mecanismo preciso y compacto para referir las semejanzas entre cadenas proporcionando una nueva y valiosa información, que puede utilizarse para guiar el proceso de búsqueda.

anexo 3

RELACION DE ESCALAS Y ANGULOS AXONOMETRICOS

En el capítulo 5 se introdujo la expresión analítica dada en (45) que nos permite determinar el ángulo que una determinada arista tridimensional forma con el plano del cuadro sobre el que es proyectada. Dicha expresión es una particularidad del bien conocido problema directo de la axonometría.

Es usual en las representaciones axonómicas las determinaciones de los coeficientes de reducción y escalas axonómicas a partir de los ángulos que forman los ejes en el espacio X , Y y Z , con sus respectivas proyecciones X' , Y' y Z' sobre el plano de proyección (figura 181).

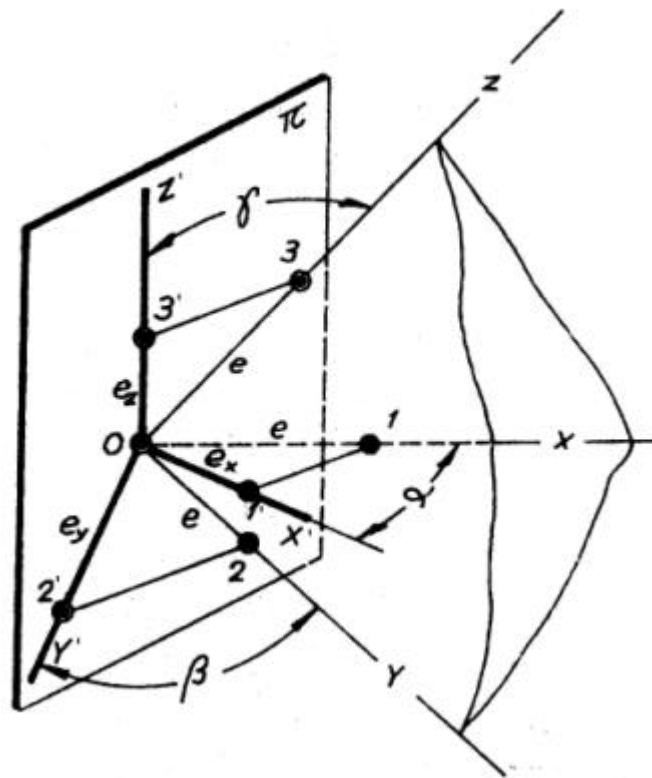


Figura 181. Deducción de los coeficientes de reducción a partir de los ángulos que forman los ejes con sus proyecciones.

Las relaciones existentes entre los segmentos 0-1', 0-2' y 0-3' del eje proyectado, y los correspondientes segmentos en el espacio 01, 02 y 03, se definen como coeficientes de reducción para cada uno de los ejes.

Considerando una misma medida real “e” sobre los ejes (y que se adopta como *escala natural*), las respectivas proyecciones valdrán:

$$\begin{aligned}e_x &= e \cos \alpha \\e_y &= e \cos \beta \\e_z &= e \cos \gamma\end{aligned}\tag{79}$$

Si la magnitud “e” se hace coincidir con la unidad, fácilmente deduciremos que las magnitudes e_x , e_y y e_z reducidas sobre los ejes, equivalen, respectivamente a los cosenos de los ángulos α , β y γ .

Nuestro caso, podría ser expresado como el problema inverso de la proyección axonométrica. Es decir, conocidas los coeficientes de reducción, determinar los ángulos que los ejes forman con el plano de proyección.

La expresión dada (45) ha sido utilizada en el ámbito de la percepción visual desde una publicación de Attneave y Frost [Attn69] en la que se analizaba la forma en la que el “ojo humano” interpretaba las proyecciones axonométricas. No obstante resulta evidente que dicha expresión no es única, es decir, que pueden ser expresada con distintos términos a partir de la formulación del problema inverso de la proyección. De hecho tenemos constancia de que otros autores han utilizado expresiones similares con distintos objetivos ligados a la interpretación de imágenes.

En este anexo se pretende realizar la justificación analítica de dicha expresión. Para ello partiremos de la representación grafica dada en la figura 182.

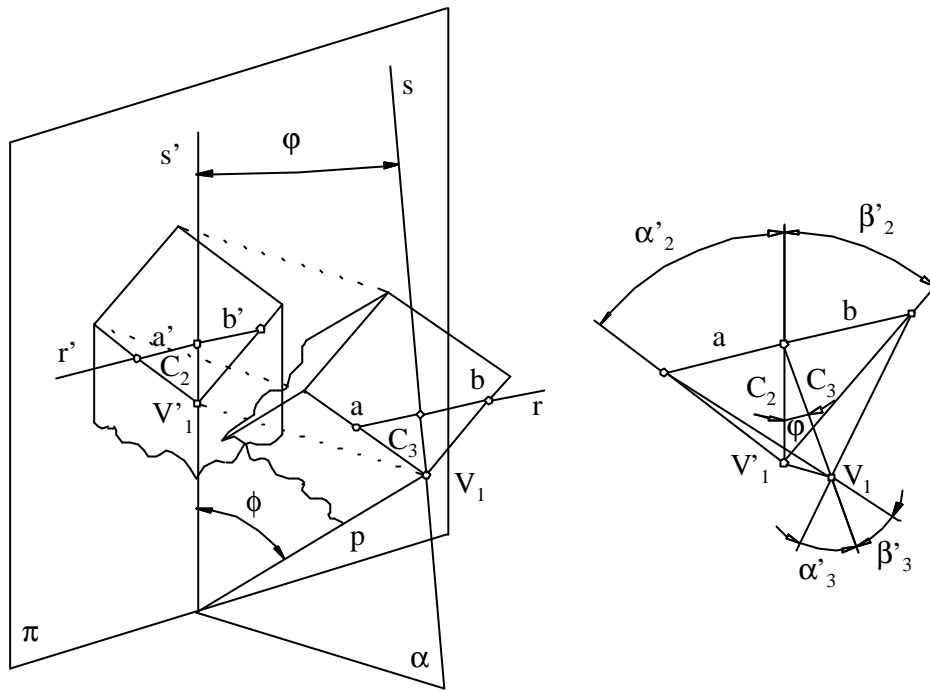


Figura 182. Determinación analítica del ángulo formado por una arista y su correspondiente proyección sobre un plano

En dicha figura se representa un modelo cúbico y su proyección ortogonal sobre el plano π . Tomado un plano α que resulte perpendicular al plano de proyección y pase por el vértice V_1 de la cara del cubo tratada, será posible definir la recta “r”, de forma que resulte ser perpendicular al plano α , y por consiguiente, su proyección r' perpendicular a la trazas de la recta “s” contenida en α . Dicha condición de perpendicularidad del plano α determina que el método sólo resulte válido para proyecciones axonométricas ortogonales.

A partir de las magnitudes representadas podemos establecer las siguientes relaciones:

1. El triángulo $S'-P-S$ resultará ser rectángulo en V_1 (porque las rectas r y s están contenidas en una cara ortogonal a la arista “p” y dado que “v” es el vértice de un triedro trirectángulo).
2. φ y ϕ resultarán ser ángulos complementarios, es decir $\varphi + \phi = 90^\circ$ (porque la dirección de proyección es ortogonal a π).

A partir de dichas relaciones es posible deducir las siguientes relaciones:

$$\begin{aligned} \cotg \alpha'_2 &= \frac{C_2}{a} & \cotg \alpha'_2 &= \frac{C_3 \cos \varphi}{a} & \frac{C_3}{a} &= \frac{\cotg \alpha'_2}{\cos \varphi} & (80) \\ \cotg \beta'_2 &= \frac{C_2}{b} & \text{de donde} & & \text{o bien} & & \\ \cotg \beta'_2 & & \cotg \beta'_2 &= \frac{C_3 \cos \varphi}{b} & \frac{C_3}{b} &= \frac{\cotg \beta'_2}{\cos \varphi} & (81) \\ C_2 &= C_3 \cos \varphi & & & & & \end{aligned}$$

De idéntica manera podrá establecerse para el modelo tridimensional:

$$\begin{aligned} \cotg \alpha'_3 &= \frac{C_3}{a} & \text{Ecuaciones que} & & \cotg \alpha'_3 &= \frac{\cotg \alpha'_2}{\cos \varphi} & (82) \\ & & \text{relacionadas con (47) y (48)} & & & & \\ & & \text{quedarán en la forma} & & \cotg \beta'_3 &= \frac{\cotg \beta'_2}{\cos \varphi} & (83) \\ \cotg \beta'_3 &= \frac{C_3}{b} & & & & & \end{aligned}$$

Ahora bien, dado que $\alpha'_3 + \beta'_3 = 90^\circ$ por constituir una de las esquinas del cubo proyectado, se verificará que:

$$\cotg \alpha'_3 \cotg \beta'_3 = 1 \quad (84)$$

por lo que del producto de las ecuaciones (82) y (83) se obtendrá la expresión:

$$\frac{\cotg \alpha'_2 \cotg \beta'_2}{\cos^2 \varphi} = 1 \quad \text{de donde} \quad \cos \varphi = \sqrt{\cotg \alpha'_2 \cotg \beta'_2} \quad (85)$$

Finalmente y teniendo presente que los ángulos φ y ϕ resultan ser complementarios, podremos establecer la relación:

$$\cos \varphi = \sin \phi \quad (86)$$

obteniendo de esta manera la expresión final dada en (45):

$$\phi = \arcsen \sqrt{\cotg \alpha'_2 \cotg \beta'_2}$$

que nos dará la relación entre el ángulo que forma una arista tridimensional con el plano del cuadro de la axonometría, en función de su proyección.

Arbol CSG	Se trata de una estructura jerárquica que representa la composición de un determinado <i>modelo</i> a partir de una serie de <i>primitivas</i> definidas y un conjunto de transformaciones. Cada rama del árbol representa un <i>primitiva</i> y cada <i>nodo</i> interno representa la operación de transformación a la que es sometido el <i>modelo</i> o <i>modelos</i> conectados al nodo.
Arista	Una arista queda definida por la intersección de dos <i>caras</i> en un <i>modelo</i> .
Arista visible	Se define como arista visible todas las <i>aristas</i> del <i>modelo</i> que en la <i>proyección</i> axonométrica dada en la <i>imagen</i> no quedaría oculta por ninguna <i>cara</i> del <i>modelo</i> .
Arista parcialmente oculta	Se define como arista parcialmente oculta todas las <i>aristas</i> del <i>modelo</i> que en la <i>proyección</i> axonométrica dada en la <i>imagen</i> quedaría oculta, de manera parcial o total por alguna de las <i>caras</i> definidas en el <i>modelo</i> .
Bandera booleana	Es un <i>bit</i> de información que determinar si una determinada condición es verdadera o falsa. Usualmente se asocia el término falso a un bit con valor 0 y el término verdadero a un bit con valor 1.
Base de un espacio de circuitos	Es el conjunto de <i>circuitos fundamentales</i> de un <i>grafo</i> . La combinación de estos <i>circuitos</i> genera todas las <i>caras potenciales</i> del <i>modelo</i> .
Bit	Unidad de información binaria. El bit puede tomar dos valores (0 1), por lo que permite la representación más elemental de un pixel. Un bit indica si la unidad pictórica de un pixel debe representarse o no.
Bocetos	Son dibujos incompletos. No contienen toda la información que es habitual en un plano técnico. Su principal utilidad es que permiten expresar ideas parciales o inacabadas.
Brep	Ver <i>Representación Brep</i> .
Cabeceo	Dado un plano XY sobre el que va a ser proyectado un <i>modelo</i> , se define como cabeceo al ángulo de giro alrededor del eje X al que es sometido el <i>modelo</i> para evitar que las <i>aristas</i> y las <i>caras</i> del <i>modelo</i> queden paralelas al plano de proyección.

Camino	Camino o <i>circuito</i> Es una secuencia conexa de <i>enlaces</i> que producen una ruta en el <i>grafo</i> que comienza en un <i>nodo</i> y termina en otro.
Cara	Es el <i>circuito cíclico</i> de un <i>grafo</i> que define una <i>cara</i> en el <i>modelo</i> que representa.
Caras potenciales	Es el conjunto de <i>circuitos cíclicos</i> de un <i>grafo</i> que son candidatos a ser <i>caras</i> del <i>modelo</i> tridimensional.
Cara visible	Se definen como caras visibles toda cara donde las <i>aristas</i> que definen su contorno son todas <i>aristas visibles</i> .
Cara oculta	Se definen como caras ocultas toda <i>cara</i> donde al menos una de las <i>aristas</i> que definen su contorno resulta ser una <i>arista parcialmente oculta</i> .
Ciclo	Ver <i>circuito cíclico</i> .
Cifra de mérito	Valor cuantitativo que permite valorar, de la manera más objetivamente posible, todos los <i>modelos</i> posibles de acuerdo con un mismo criterio. En otras palabras, la cifra de mérito permite cuantificar la bondad de las distintas <i>soluciones</i> posibles.
Circuito	Ver <i>camino</i> .
Circuito simple	Se define como circuito simple a aquel <i>camino</i> en el que no se repite ningún <i>nodo</i> del <i>grafo</i> .
Circuito cíclico	Es un <i>circuito simple</i> con la características de que el primer y último <i>nodo</i> son el mismo.
Circuitos fundamentales	Conjunto de <i>circuitos cíclicos</i> tal que todos ellos poseen un <i>enlace</i> que son se encuentran en el resto de los <i>circuitos</i> y entre todos recorren todas los <i>enlaces</i> del <i>grafo</i> .
Contorno aparente	Es el conjunto de <i>enlaces</i> y <i>nodos</i> que representa el <i>circuito</i> exterior de un <i>grafo</i> . Desde el punto de vista de la geometría se define como la <i>proyección</i> sobre el plano de la imagen del cilindro o cono circunscrito a un <i>modelo</i> .
Clase de vecindad	Conjunto de <i>soluciones</i> a las que se puede acceder desde una <i>solución</i> dada aplicando el mecanismo de generación.
Croquis	Dibujos que permiten expresar formas geométricas sin atarse a los criterios estrictos de la geometría. Se trata de dibujos “imperfectos” o inconsistentes desde el punto de vista geométrico, porque no se trazan utilizando instrumentos de delineación. También se denominan dibujos “a mano alzada”.

CSG	Ver <i>Representación CSG</i> .
Desviación máxima permisible	Desviación máxima que puede tomar una variable respecto a un <i>valor de referencia</i> para considerar que se verifica cierta propiedad que representa la variable.
Dibujo lineal	Dibujo que expresa de forma estricta la geometría de una <i>imagen</i> .
Direcciones principales	Se consideran como direcciones principales aquellas que se repiten con mayor frecuencia en una <i>imagen</i> .
Eje de una pirámide	Es la <i>arista</i> que une el <i>vértice</i> de la pirámide y el centro geométrico de su <i>polígono directriz</i> .
Enlace	Son las líneas que muestran las conexiones entre dos <i>nodos</i> distintos.
Epoca	Número de transiciones necesario hasta conseguir un número fijo de transiciones aceptadas.
Equiangularidad	Se dice que una forma o un <i>modelo</i> es equiangular cuando todos los ángulos existentes en la forma o el <i>modelo</i> son iguales.
Espacio de circuitos	Es el conjunto de <i>caras potenciales</i> definidas en un <i>grafo</i> .
Espacio de soluciones	Conjunto infinito de vectores n-dimensionales que representan <i>soluciones</i> al problema de reconstrucción.
Evaluar un modelo	Evaluar un <i>modelo</i> es establecer una forma cuantitativa que permita asignar un valor a cada <i>modelo</i> . Dicha evaluación supone usualmente asignar una <i>cifra de mérito</i> a un <i>modelo</i> .
Extensión ortográfica	Conjunto infinito de <i>modelos</i> geoméricamente válidos obtenidos a partir de una <i>imagen</i> . Se considera que un <i>modelo</i> es válido si su <i>proyección</i> coincide con la <i>imagen</i> de partida.
Factor de confianza	Intervalo de valores numérico limitado definido por un <i>valor de referencia</i> y la <i>desviación máxima permisible</i> que marcan el cumplimiento de una determinada propiedad. Conforme nos aproximamos dentro del intervalo al <i>valor de referencia</i> , el factor de confianza se aproxima a 1. Conforme nos acercamos dentro del intervalo a la máxima <i>desviación permisible</i> , el factor de confianza se aproxima a 0.
Figura proyección	Es la figura obtenida como consecuencia de la proyección de un <i>modelo</i> . Usualmente se denomina simplemente proyección.

Figura imagen	Es la imagen obtenida como consecuencia de la sección producida por un plano a la <i>figura proyección</i> . Usualmente se denomina simplemente imagen.
Figuras imposibles	Se dice que una <i>figura imagen</i> corresponde a una figura imposible, cuando el <i>modelo</i> que representa no puede ser construido materialmente.
Grado de un vértice	Es el número de <i>aristas</i> de las que forma parte un <i>vértice</i> . Es decir, el número de <i>aristas</i> que concurren en un <i>vértice</i> .
Grafo	Conjunto de <i>nodos</i> y <i>enlaces</i> . Cada <i>nodo</i> define la intersección de varios <i>enlaces</i> . Cada <i>enlace</i> une dos <i>nodos</i> . Un grafo representa una <i>figura imagen</i> que es <i>proyección</i> de un <i>modelo</i> .
Grafo conexo	Se dice que un <i>grafo</i> es conexo o simple si desde cualquiera de sus <i>nodos</i> se puede llegar a cualquier otro <i>nodo</i> del <i>grafo</i> . Un grafo conexo siempre representa a un <i>poliedro euleriano</i> .
Grafo simple	Ver <i>grafo conexo</i> .
Guiñada	Dado un plano XY sobre el que va a ser proyectado un <i>modelo</i> , se define como guiñada al ángulo de giro alrededor del eje Z (eje perpendicular al plano XY) al que es sometido el <i>modelo</i> para evitar que las <i>aristas</i> y las <i>caras</i> del <i>modelo</i> queden paralelas al plano de proyección.
Imagen	Ver <i>figura imagen</i> .
Imagen geoméricamente consistente	Se define como imagen geoméricamente consistente aquella <i>imagen</i> que verifica todas las exigencias geométricas de una <i>proyección</i> , tales como paralelismo de <i>aristas</i> . Y que además está libre de errores de interpretación tales como que dos <i>aristas</i> no concurren en un mismo <i>vértice</i> , por hacerlo en <i>vértices</i> muy próximos entre si, o que una <i>aristas</i> representa dos <i>aristas</i> solapadas o coincidentes.
Imagen matricial	Almacenamiento de una <i>imagen</i> mediante un mapa de pixels.
Imagen vectorial	Almacenamiento de una <i>imagen</i> mediante elementos primitivos tales como puntos, líneas y curvas.
Método de inflado	Consiste en mantener las coordenadas (X,Y) de todos los <i>vértices</i> del <i>modelo</i> iguales a las coordenadas del correspondiente <i>nodo</i> de la <i>imagen</i> . Las coordenadas Z de los <i>vértices</i> del <i>modelo</i> constituyen las <i>variables de reconstrucción</i> .

Modelo	Representación tridimensional de un objeto. Todos los <i>vértices</i> que definen el <i>modelo</i> tienen asignadas coordenadas x, y z.
Modelo geoméricamente válido	<i>Modelo</i> que al ser proyectado sobre un plano da como resultado un conjunto <i>enlaces</i> y <i>nodos</i> que coinciden con el <i>grafo</i> de partida.
Modelo normalón	Se definen como normalón aquellos <i>modelos</i> en los que todas sus <i>aristas</i> están orientadas según tres direcciones ortogonales.
Modelo origami	<i>Modelo</i> no sólido. Las paredes que definen el contorno del <i>modelo</i> se suponen de espesor despreciable.
Modelo prevalente	Mejor forma de organización de un <i>modelo</i> . La “mejor forma” es evaluada de acuerdo con el cumplimiento de unas determinadas reglas de percepción. Al <i>modelo</i> prevalente se le llama también <i>modelo</i> tentativo.
Modelo psicológicamente aceptable	Se dice que un <i>modelo</i> es psicológicamente aceptable cuando el <i>modelo</i> obtenido a partir de una <i>imagen</i> mediante procesos de reconstrucción concuerda con el <i>modelo</i> interpretado por ser humano cuando visualiza una <i>imagen</i> . En otras palabras, el <i>modelo</i> obtenido automáticamente mediante reconstrucción sería el mismo <i>modelo</i> que un ser humano diseñaría en un sistema CAD cuando visualiza una <i>imagen</i> .
Número ciclomático	Es el número de <i>circuitos fundamentales</i> de un <i>grafo</i> y queda definido por la expresión $E-N+1$; siendo E el número de <i>enlaces</i> del <i>grafo</i> y N el número de <i>nodos</i> en el <i>grafo</i> .
Objeto poliédrico	<i>Modelo</i> geométrico limitado por regiones planas. Cada una de dichas regiones identifica una <i>cara</i> del poliedro. Una <i>arista</i> queda definida por la intersección de dos <i>caras</i> y un <i>vértice</i> queda definido por la intersección de dos o más <i>aristas</i> .
Óptimo global	<i>Punto solución</i> de un espacio n-dimensional donde la función objetivo alcanza su coste mínimo.
Óptimo local	<i>Punto solución</i> de un espacio n-dimensional donde la función objetivo alcanza un coste mínimo en un volumen más o menos extenso alrededor del <i>punto solución</i> .
Pixels	Elemento pictórico. Mínima unidad de representación. Cada pixel lleva asociado características tales como color, luminosidad y contraste.

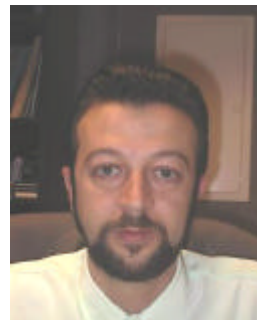
Planta superior de la imagen	Se define como planta superior de la imagen al contorno de <i>enlaces</i> de la <i>imagen</i> que definen la <i>cara</i> del <i>modelo</i> que tendría mayor coordenada “Z” supuesto apoyado el <i>modelo</i> en un plano XY. En el caso de existir más de una <i>cara</i> con igual coordenada “Z” se selecciona aquella que en la <i>imagen</i> presente un <i>vértice</i> con mayor coordenada “Y”.
Poliedros eulerianos	Ver <i>poliedro simple</i> .
Poliedro simple	O <i>poliedro euleriano</i> , es aquel poliedro que puede deformarse para obtener una esfera, es decir, un poliedro que a diferencia de un toro no tiene agujeros. Los poliedros simples satisfacen la ecuación de Euler.
Polígono directriz de una pirámide	Es el conjunto de <i>enlaces</i> que representan la directriz de la superficie reglada.
Polígonos normalones	Se define como polígono normalón aquel polígono que tienen la propiedad de que todos los ángulos entre dos <i>aristas</i> concurrentes son de 90°
Primitivas	Conjunto de entidades simple. Dichas primitivas se corresponden con puntos, rectas y curvas en una <i>Representación Brep</i> y con <i>modelos</i> definidos y parametrizados en <i>Representaciones CSG</i> .
Proyección	Ver <i>figura proyección</i> .
Proyección central	Se dice que la <i>proyección</i> de un <i>modelo</i> sobre un plano es central cuando los rayos proyectantes del <i>modelo</i> pasan un punto fijo (centro de proyección propio).
Proyección oblicua	Se dice que una <i>proyección paralela</i> es oblicua cuando los rayos proyectantes no son perpendiculares al plano que contiene la <i>imagen</i>
Proyección ortogonal	Se dice que una <i>proyección paralela</i> es ortogonal cuando los rayos proyectantes son perpendiculares al plano que contiene la <i>imagen</i> .
Proyección paralela o cilíndrica	Se dice que la <i>proyección</i> de un <i>modelo</i> sobre un plano es paralela o cilíndrica cuando los rayos proyectantes del <i>modelo</i> son paralelos a una dirección dada (centro de proyección impropio).
Punto solución	Ver <i>solución</i> .
Rango de arista	Es el número de <i>caras</i> de las que forma parte una <i>arista</i> .
Rango de vértice	Es el número de <i>caras</i> del que forma parte un <i>vértice</i> .

Reconstrucción Geométrica	Es la determinación de las relaciones topológicas y geométricas de todas las partes atómicas de un objeto a partir de una o varias imágenes
REGEO	Grupo de investigación de Reconstrucción Geométrica de Sólidos.
Reglas de percepción	Conjunto de reglas que el ser humano utiliza intuitivamente en la interpretación de imágenes bidimensionales.
Reglas heurísticas	
Regularidad	Expresión analítica de las reglas de la percepción visual humana que rigen la interpretación de <i>dibujos lineales</i> .
Representación Brep	Representación de un <i>modelo</i> por un conjunto finito de superficies delimitadas por un conjunto de <i>aristas</i> .
Representación CSG	Representación de un <i>modelo</i> a partir de un conjunto de <i>primitivas</i> , una serie de transformaciones y un conjunto de operaciones booleanas.
Restitución	Es la identificación de un objeto mediante algún sistema de acoplamiento de plantillas.
Sensibilidades	Parámetros que miden la influencia relativa de cada una de las variables sobre la función objetivo.
Sistemas CAD	Sistemas de diseño asistido por ordenador.
Solución	O “ <i>punto solución</i> ” es el vector n-dimensional que representa un valor de las <i>variables de reconstrucción</i> .
Solución trivial	Es la <i>solución</i> definida por el vector cuyas componentes son todas nulas (= 0).
Subespacio de soluciones	Se trata de una representación bidimensional de un espacio n-dimensional donde sólo se modifican los valores que toman dos variables. En realidad se trata de representar un corte del espacio n dimensional de <i>soluciones</i> , ya que la representación completa resulta imposible.
Valor de referencia	Valor numérico tomado como referencia para medir las desviación de una determinada magnitud.
Variables de reconstrucción	Conjunto de variables que permiten definir los distintos <i>modelos</i> reconstruidos a partir de una <i>imagen</i> . El método propuesto, las coordenadas “Z” de cada <i>vértice</i> del <i>modelo</i> constituyen las variables de reconstrucción.
Vértice	Es el punto tridimensional definido por la intersección de dos o más <i>aristas</i> .

Visión artificial Ver visión por ordenador.

Visión por ordenador Conjunto de todas las herramientas y estrategias empleadas para hacer “ver” a los ordenadores.

BREVE CURRICULUM



Nacido en 1967 en Cartagena, Provincia de Murcia. Ingeniero Industrial por la Universidad de Murcia desde 1992. Ingresó en la actual Universidad Politécnica de Cartagena (entonces dependiente de la U. de Murcia) como profesor asociado a tiempo parcial en 1995, pasando a profesor asociado con dedicación exclusiva en 1996. Desde 1998 es Profesor Titular de Escuela Universitaria de Expresión Gráfica en la Ingeniería. Ostentó el cargo de Secretario de Departamento de Expresión Gráfica desde 1997 hasta Junio del 2001, siendo cesado para recibir el nombramiento de Coordinador de las pruebas de acceso a la Universidad de la Comunidad de Murcia en la materia de Dibujo Técnico (<http://www.upct.es/~deg/Personales/Jcp/Principal.htm>).

Imparte docencia de Diseño Industrial y Diseño Asistido por Ordenador en la Titulación de Ingeniero Industrial habiendo impartido también docencia de Expresión Gráfica y Diseño Asistido por Ordenador en las titulaciones de Ingeniero Técnico Industrial en las especialidades de Electrónica, Electricidad y Química Industrial, así como de la asignatura de Dibujo Industrial en la titulación de Ingeniero Técnico Industrial en la especialidad Mecánica.

Su área de investigación se centra en el diseño asistido por ordenador y los gráficos por computador. Fruto de los trabajos realizados en este área son las comunicaciones presentadas en distintos congresos, entre las que destacan “Initial Modeling strategies for geometrical reconstruction optimization-based approaches” presentada en el 11th ADM International Conference on Design tools and Methods in Industrial Engineering, y “Comportamiento de los algoritmos de optimización en la Reconstrucción Geométrica de Sólidos” presentada en el XIV Congreso Nacional de Ingeniería Mecánica.

En la actualidad forma parte de los grupos de investigación IDAO (<http://www.upct.es/~eyc>) y REGEO (<http://www.tec.uji.es/regeo>). Su actividad principal se desarrolla en los nuevos métodos de Reconstrucción, labor que desarrolla de manera conjunta con la Universidad Jaume I de Castellón y la Universidad Politécnica de Valencia.

