



**UNIVERSIDAD JAUME I DE CASTELLÓN**  
**DEPARTAMENTO DE TECNOLOGÍA**

**"PERCEPCIÓN ARTIFICIAL DE  
DIBUJOS LINEALES"**

**Ana Piquer Vicent**  
**Septiembre de 2003**



**TESIS DOCTORAL**

**PERCEPCIÓN ARTIFICIAL DE DIBUJOS  
LINEALES**

**DEPARTAMENTO DE TECNOLOGÍA  
DE LA UNIVERSIDAD JAUME I**

por  
**ANA PIQUER VICENT**

para la obtención del  
**GRADO DE DOCTOR INGENIERO INDUSTRIAL**  
**European PhD in Industrial Engineering**

Director de tesis:  
**PEDRO P. COMPANY CALLEJA**

Castellón, Septiembre de 2003



## **AUTORIZACIÓN DEL DIRECTOR DE LA TESIS PARA SU PRESENTACIÓN**

**Dr. Pedro P. Company Calleja** como **Director de la Tesis Doctoral**

**PERCEPCIÓN ARTIFICIAL DE DIBUJOS LINEALES**

realizada en el Departamento de Tecnología de la Universidad Jaume I de Castellón por el doctorando **Doña Ana Piquer Vicent**, autorizo la presentación de la citada Tesis Doctoral, dado que reúne las condiciones necesarias para su defensa.

En Castellón, a .....

EL DIRECTOR DE LA TESIS



## **AUTORIZACIÓN DEL DEPARTAMENTO**

**Dr. Pedro P. Company Calleja** como **Director del Departamento de Tecnología** INFORMA que la Tesis Doctoral titulada

**PERCEPCIÓN ARTIFICIAL DE DIBUJOS LINEALES**

Ha sido realizada por el doctorando **Doña Ana Piquer Vicent**, bajo la dirección y supervisión de **Don Pedro P. Company Calleja** y que el Departamento ha dado su conformidad para que sea presentada ante la Comisión General de Doctorado.

En Castellón, a .....

EL DIRECTOR DEL DEPARTAMENTO





# Agradecimientos

---

En los últimos años he tenido la suerte de trabajar con mucha gente, que de uno u otro modo me han ayudado en la realización de esta tesis. Es por ello, que quiero dedicar unos breves párrafos en agradecimiento a todos ellos.

En primer lugar quiero agradecer el apoyo y dedicación en la dirección de esta tesis ofrecido por el Dr. Pedro Company. Sus sugerencias e ideas han sido de gran ayuda para mi formación como investigadora, así como para el desarrollo y buen fin de este documento.

También me gustaría agradecer los años de colaboración a los compañeros del Departamento de Expresión Gráfica en la Ingeniería de la Universidad Politécnica de Valencia y de la Universidad Politécnica de Cartagena, y el estupendo trato que tuvieron al acogerme en las estancias realizadas en sendas Universidades. En especial quiero dar las gracias a Julián Conesa por los ratos de programación, la paciencia y el buen humor que siempre ha tenido.

Asimismo, quiero agradecer a todos los miembros del Departamento de Computer Science de la Universidad de Cardiff, por ofrecerme una experiencia muy enriquecedora durante mi estancia en su Universidad.

No sería justo no dar mi agradecimiento a toda mi familia, que me ha apoyado en toda mi carrera de estudios. A mi madre, por iniciarme en el estudio desde mi infancia, y a mis hermanos por estar siempre a mi lado.

Por último, aunque no por ello menos importante, agradecer el apoyo moral que me han dado Ricardo y Ricardo Jr. Con ellos espero recuperar todos los momentos perdidos durante estos últimos tiempos.

Gracias a todos.



A Ricardo



# Índice

---

<b>Agradecimientos .....</b>	<b>i</b>
<b>Índice .....</b>	<b>v</b>
<b>Índice de Tablas y Figuras .....</b>	<b>xi</b>
<b>Glosario de Símbolos .....</b>	<b>xvii</b>
<b>Resumen .....</b>	<b>1</b>
<b>Abstract .....</b>	<b>5</b>
<b>Capítulo 1. Introducción.....</b>	<b>9</b>
1.1. EL PROCESO DE DISEÑO .....	9
1.2. LA RECONSTRUCCIÓN TRIDIMENSIONAL .....	10
1.2.1. Datos de partida: Vectorización y Bocetado on-line .....	12
1.2.2. Reconstrucción 2D.....	14
1.2.3. Reconstrucción geométrica 3D.....	17
1.3. OBJETIVOS DE LA TESIS .....	17
1.4. ESTRUCTURA DE LA TESIS .....	18

1.5. EL PROGRAMA REFER.....19

## **Capítulo 2. Antecedentes de la Reconstrucción 3D..... 21**

---

2.1. INTRODUCCIÓN .....21

2.2. CLASIFICACIÓN GENERAL DE LOS ALGORITMOS DE RECONSTRUCCIÓN 3D  
.....22

    2.2.1. Vista Única – Vistas múltiples.....23

2.3. RECONSTRUCCIÓN DE MODELOS 3D DESDE UNA ÚNICA VISTA .....24

    2.3.1. Enfoque del Etiquetado de Líneas .....26

    2.3.2. Método del Espacio del Gradiente .....30

    2.3.3. Enfoque basado en la programación lineal.....34

    2.3.4. Enfoque Progresivo .....38

    2.3.5. Enfoque de la Identificación de Primitivas .....41

    2.3.6. Enfoque de las Regularidades.....42

2.4. RECONSTRUCCIÓN TRIDIMENSIONAL DE MODELOS DESDE MÚLTIPLES  
VISTAS .....45

    2.4.1. Representación Alámbrica / B-Rep.....47

    2.4.2. CSG - Geometría Constructiva de Sólidos .....53

2.5. RESUMEN DE LA CLASIFICACIÓN .....55

    2.5.1. Tipos de superficies .....55

    2.5.2. Interacción .....55

    2.5.3. Representación de líneas ocultas en 2D .....56

    2.5.4. Datos de entrada .....56

2.6. CONCLUSIONES.....58

## **Capítulo 3. La Reconstrucción 3D y las Regularidades .... 61**

---

3.1. INTRODUCCIÓN .....61

3.2. RECONSTRUCCIÓN 2D .....62

3.3. LA OPTIMIZACIÓN .....64

---

3.3.1. El Inflado y la Función Objetivo .....	64
3.3.2. Técnicas de Optimización Matemática “local” .....	66
3.3.3. Sintonizar y Normalizar la Optimización .....	67
3.3.4. Técnicas de Optimización Matemática “global” .....	68
3.4. MODELOS TENTATIVOS.....	69
3.5. REGULARIDADES.....	71
3.5.1. Regularidades falsas y verdaderas.....	71
3.5.2. Planicidad De Caras .....	72
3.5.3. Paralelismo De Líneas.....	73
3.5.4. MSDA .....	74
3.5.5. Verticalidad De Líneas.....	75
3.5.6. Isometría .....	75
3.5.7. Ortogonalidad De Esquinas.....	76
3.5.8. Ortogonalidad De Líneas.....	77
3.5.9. Perpendicularidad De Caras.....	77
3.5.10. Colinealidad de Líneas .....	77
3.5.11. Simetría Facial Oblicua.....	79
3.6. CONCLUSIONES DEL ESTUDIO DE REGULARIDADES .....	79

## **Capítulo 4. Detección de Caras de Poliedros en Representaciones 2D. .... 81**

---

4.1. INTRODUCCIÓN .....	81
4.1.1. Figuras vistas como grafos .....	82
4.1.2. Sobre los poliedros y sus caras .....	83
4.1.3. Sobre la conectividad de los grafos .....	86
4.1.4. Sobre la planaridad de los grafos. ....	87
4.2. ALGORITMOS DE DETECCIÓN DE CARAS ESTUDIADOS.....	88
4.2.1. Roberts .....	89
4.2.2. Leclerc y Fishler .....	90

## Percepción Artificial de Dibujos Lineales

---

4.2.3. Lipson y Shpitalni.....	91
4.2.4. Courter y Brewer.....	93
4.3. MEJORAS APORTADAS.....	99
4.3.1. Detección de los vértices de corte.....	100
4.3.2. Elección de la posible combinación de caras.....	102
4.3.3. Resultados.....	105
4.3.4. Limitaciones.....	106

## Capítulo 5. Simetría Facial Oblicua..... 109

---

5.1. INTRODUCCIÓN.....	109
5.1.1. Simetría.....	110
5.1.2. Simetría Oblicua.....	111
5.2. ANTECEDENTES.....	113
5.3. MÉTODOS ESTUDIADOS DE DETECCIÓN DE SIMETRÍA OBLICUA EN RECONSTRUCCIÓN.....	116
5.3.1. Kanade.....	116
5.3.2. Lipson y Shpitalni.....	117
5.4. MÉTODO DE DETECCIÓN DE EJES DE SIMETRÍA OBLICUA UTILIZADO.....	120
5.4.1. Formulación de la función inversa a la oblicuidad.....	121
5.4.2. Aplicación del método.....	124
5.5. LA SIMETRÍA FACIAL OBLICUA COMO REGULARIDAD.....	128
5.5.1. Cambio de Sistema de Referencia.....	128
5.5.2. Cálculo del Coste de la Regularidad.....	131
5.6. CONCLUSIONES.....	132

## Capítulo 6. Simetría de Modelos 3D detectados en el Espacio 2D..... 135

---

6.1. INTRODUCCIÓN.....	135
6.2. DETECCIÓN DE PLANOS DE SIMETRÍA EN POLIEDROS.....	137



---

6.2.1. Reglas fundamentales para el cálculo de los planos de simetría. ....	140
6.3. LIMITACIONES DEL MÉTODO DE DETECCIÓN DE PLANOS DE SIMETRÍA EN REPRESENTACIONES 2D DE POLIEDROS.....	145
6.4. PRIMERAS APROXIMACIONES A LA UTILIZACIÓN DE LOS PLANOS DE SIMETRÍA .....	148
6.4.1. Planicidad del plano de simetría .....	148
6.4.2. Distancia de los vértices del modelo al plano de simetría .....	149
6.4.3. Ortogonalidad respecto al plano de simetría .....	150
6.4.4. Aplicación. ....	150
6.5. GENERALIDADES DE LOS NUEVOS MÉTODOS. ....	151
6.6. SISTEMAS DE SIMETRÍA .....	153
6.7. CONDICIÓN DE SIMETRÍA.....	154
6.8. EMPAREJAMIENTO DE VÉRTICES SIMÉTRICOS.....	155
6.8.1. Identificación de los dos lados de un modelo simétrico. ....	155
6.8.2. Emparejamiento de vértices simétricos. ....	156
6.9. LA SIMETRÍA DEL MODELO COMO REGULARIDAD .....	159
6.10. RECONSTRUCCIÓN DEL MODELO PARTIENDO DE LA MITAD SIMÉTRICA..	161
6.10.1. La función de simetría 3D. ....	162
6.11. RESULTADOS.....	164
6.12. CONCLUSIONES.....	170

## **Capítulo 7. Conclusiones y Desarrollos Futuros..... 173**

---

7.1. CONCLUSIONES.....	173
7.1.1. Reconstrucción tridimensional .....	174
7.1.2. Detección de Caras .....	174
7.1.3. Regularidades. Simetría facial y Simetría bilateral del modelo .....	175
7.1.4. Aplicación Docente .....	176
7.2. DESARROLLOS FUTUROS .....	176
7.2.1. Modelos representados mediante grafos no-conexos .....	177

**Percepción Artificial de Dibujos Lineales**

---

7.2.2. Modelos con Caras Curvas.....178  
7.2.3. PSO .....180  
7.2.4. Motores Geométricos.....180  
7.3. PUBLICACIONES relacionadas con esta tesis .....181

**Glosario ..... 183**

**Bibliografía..... 195**

**Anexo I: Aplicación Docente de la Reconstrucción..... 215**

---

I.1. INTRODUCCIÓN .....215  
I.2. BOCETADOR/MODELADOR DOCENTE .....217  
I.3. DISCUSIÓN .....220  
I.4. CONCLUSIONES Y TRABAJOS FUTUROS .....222

# Índice de Tablas y Figuras

---

Tabla 2-1. Clasificación General de los algoritmos de reconstrucción. ....	24
Tabla 2-2. Métodos de reconstrucción partiendo de una única vista.....	57
Tabla 2-3. Métodos de reconstrucción partiendo de vistas múltiples .....	58
Tabla 6-1. Ejemplos simétricos reconstruidos con la regularidad de simetría bilateral. ....	167
Figura 1-1. Ámbito de la Reconstrucción Tridimensional. ....	11
Figura 1-2. Objetivo principal de la Reconstrucción 3D.....	12
Figura 1-3 Fases de la reconstrucción de dibujos de ingeniería .....	15
Figura 1-4. Cuatro estados en el modelado mediante bocetado. ....	16
Figura 2-1. Modelo de Roberts, 1963.....	25
Figura 2-2. Diccionario de los tipos de uniones y sus posible etiquetado para un vértice triédrico. ....	26
Figura 2-3. Figura de superficies planas etiquetada.....	27
Figura 2-4. Figura con superficies curvas etiquetada.....	27
Figura 2-5. Gráfico resumen del método de etiquetado. ....	28
Figura 2-6. Modelo con etiquetado lógico.....	29
Figura 2-7. Tipos de uniones triédricas y tetraédricas.....	30
Figura 2-8. Proyección de un punto dual I en el espacio del gradiente.....	31

## Percepción Artificial de Dibujos Lineales

---

Figura 2-9. a) A y B son dos regiones en proyección unidas por una línea. La representación de las dos caras en el espacio dual, b) la arista que las une es convexa, c) la arista es cóncava.....	32
Figura 2-10. Gráfico resumen del método basado en el espacio del gradiente.....	33
Figura 2-11. a) Proyección que cualquier ser humano reconocería como un tronco de pirámide de base triangular. b) Figura matemáticamente reconstruible. c) figura matemáticamente imposible de reconstruir .....	35
Figura 2-12. Gráfico resumen del método basado en la programación lineal. ....	37
Figura 2-13. Gráfico resumen del enfoque progresivo. ....	39
Figura 2-14. Reconstrucción mediante identificación de primitivas (a) Los estados durante el proceso; (b) el árbol CSG final.....	40
Figura 2-15. Gráfico resumen del método de identificación de primitivas. ....	42
Figura 2-16. Gráfico resumen del método de las regularidades.....	45
Figura 2-17. Modelo celular y modelos sólidos posibles (extraído del artículo de Masuda).51	
Figura 2-18. Datos de partida de un ejemplo de Tanaka.....	52
Figura 2-19. Una de las cinco soluciones que el algoritmo de Tanaka ofrece.....	52
Figura 2-20. Resumen cronológico de los autores de algoritmos de reconstrucción partiendo de vistas múltiples.....	54
Figura 2-21. Evolución de los métodos que parten de una Única Vista. ....	59
Figura 3-1. Boceto inicial y dibujo lineal de salida de una forma poliédrica.....	63
Figura 3-2. a) Dibujo lineal con un falso paralelismo de líneas, b) la solución perceptualmente mejor. ....	64
Figura 3-3. Proceso de inflado. Algunos modelos de la extensión ortográfica intermedios, junto con el correcto.....	65
Figura 3-4. Grandes pasos en Hill-Climbing dejan modelos no proporcionales a) muestra el dibujo original, b) es el modelo psicológicamente correcto (obtenido con $s_1 = 0.2 \max(\Delta x, \Delta y)$ ), c) modelo no proporcionado (obtenido con $s_1 = 2.0 \max(\Delta x, \Delta y)$ ). ....	67
Figura 3-5. Cabeceo del sistema de referencia. ....	75
Figura 3-6. Posibles proyecciones de esquinas ortogonales.....	76
Figura 3-7. Comprobación de colinealidad de dos aristas $L_1$ y $L_2$ .....	78
Figura 3-8. Las dos aristas comparadas ( $L_1$ y $L_2$ ) y las cuatro mediciones ( $L_3, L_4, L_5$ y $L_6$ ) ..	79

Figura 4-1. Figuras de Kopfermann.....	83
Figura 4-2.. a) Poliedro 2-variedad, b) Politopo sin variedad, c) Poliedro que no es una 2-variedad.....	84
Figura 4-3. Esfera $S_1$ con curva $C_1$ que divide su superficie en dos regiones (izquierda), Esfera $S_2$ con asa $H_1$ y dos curvas $C_1$ y $C_2$ que dividen su superficie en dos regiones (derecha) .....	85
Figura 4-4. a) Poliedro de género cero y representado por un grafo conexo, b) poliedro de género cero representado por tres componentes conexas, c) poliedro de género 1 y representado por dos componentes conexas, d) poliedro de género 1 representado por un grafo conexo.....	85
Figura 4-5. Grafo conexo de grado mínimo igual a tres y con una conectividad de aristas de dos (izquierda), dos bloques no conexos resultantes de eliminar las dos aristas del conjunto de corte (derecha).....	86
Figura 4-6. Grafos $K_{3,3}$ y $K_5$ .....	87
Figura 4-7. a) Grafo original no plano (izquierda), b) y c) dos formas diferentes de sumergirlo en un plano.....	88
Figura 4-8. Detección de caras en una representación sin aristas ocultas .....	90
Figura 4-9. Grafo conexo.....	94
Figura 4-10. Árbol BFS y Matriz de Incidencia .....	95
Figura 4-11. Posibles circuitos fundamentales de la Figura 4-9 .....	96
Figura 4-12. Caras detectadas en el dibujo 2D .....	97
Figura 4-13. Cálculo de caras: 57 aristas, 37 vértices y 22 caras .....	98
Figura 4-14. Cálculo de caras: 55 aristas, 35 vértices y 22 caras .....	98
Figura 4-15. Cálculo de caras: 85 aristas, 56 vértices y 31 caras .....	98
Figura 4-16. Objeto 2-conexo.....	99
Figura 4-17. Dos posibles configuraciones planas de un modelo planar.....	100
Figura 4-18. Caras calculadas antes de realizar la comprobación de cruces.....	102
Figura 4-19. Grafo sobre el que se ha calculado una cara falsa.....	103
Figura 4-20. Caras detectadas después de la aplicación del algoritmo de resolución de caras cruzadas.....	104
Figura 4-21. Calculo de Caras antes y después de la aplicación del método de resolución de cruces.....	105

## Percepción Artificial de Dibujos Lineales

---

Figura 4-22.. Ejemplo con 16 caras con 4 posibles parejas de caras cruzadas .....	106
Figura 4-23. Otro caso donde se requiere la aplicación del método de resolución de cruces. .....	106
Figura 4-24. Modelo con posible cara falsa .....	107
Figura 4-25. Modelo con agujero posiblemente pasante.....	107
Figura 5-1. Representación de un poliedro con caras transparentes .....	110
Figura 5-2. Reflexión en E de un punto p. ....	111
Figura 5-3. Figura plana con simetría especular. ....	112
Figura 5-4. Eje de simetría oblicua y su conjugado de la proyección de una cara simple..	112
Figura 5-5. Modelo 2D con una de sus caras con simetría facial oblicua destacada. ....	114
Figura 5-6. Hipérbola utilizada por Kanade .....	116
Figura 5-7. Contorno con simetría facial oblicua en la que se representan los vectores unitarios de las direcciones del eje de simetría oblicua y de su conjugado .....	118
Figura 5-8. a) Objeto poliédrico b) Una cara simétrica con eje de simetría y eje conjugado c) Simetría oblicua obtenida por proyección cilíndrica de la cara .....	121
Figura 5-9. Coordenadas ortogonales y oblicuas de un punto p. ....	122
Figura 5-10. Ejes oblicuos en un contorno con simetría oblicua. ....	123
Figura 5-11. Ejes tentativos de simetría .....	125
Figura 5-12. Modelos proyectados con todos sus ejes de simetría facial oblicua representados.....	127
Figura 5-13. Cara inflada y su proyección.....	128
Figura 5-14. Cambio de sistema de referencia.....	130
Figura 5-15. Ejemplos sin simetría cuya suma de coordenadas $Y''$ de sus vértices es nula. .....	132
Figura 6-1. Boceto 2D y Modelo 3D reconstruido sin tener en cuenta la simetría.....	135
Figura 6-2.. Del boceto al modelo 3D.....	136
Figura 6-3. Plano de simetría de un modelo representado por un circuito 2D.....	138
Figura 6-4. Modelos proyectados con todos sus ejes de simetría facial oblicua representados.....	139
Figura 6-5. Reglas para la reducción de líneas candidatas a formar parte de los planos de simetría del modelo.....	141

Figura 6-6 Grafo con sus ejes de simetría oblicua identificados .....	142
Figura 6-7. Cubo con todos sus ejes de simetría facial detectados .....	143
Figura 6-8. Proceso de introducción de datos y reconocimiento de los planos de simetría del modelo. ....	143
Figura 6-9. Modelo sobre el que se ha detectado un plano de simetría global y otro parcial. ....	144
Figura 6-10. Modelo de caras triangulares con todos los planos de simetría representados. ....	144
Figura 6-11. Cubo con sus tres planos de simetría principales detectados.....	146
Figura 6-12. Modelos proyectados con todos los planos de simetría que detecta el método propuesto.....	147
Figura 6-13. Modelo que contiene planos de simetría cuyo polígono de simetría está formado exclusivamente por aristas del modelo.....	148
Figura 6-14. Modelo reconstruido inaceptable de un bloque prismático. ....	151
Figura 6-15. Determinación de subgrupos de vértices simétricos. ....	152
Figura 6-16. Sistemas de referencia utilizados en la aplicación de los planos de simetría en la Reconstrucción. ....	152
Figura 6-17. Transformación al sistema de referencia de simetría.....	153
Figura 6-18. Vértices etiquetados en función de un plano de simetría.....	156
Figura 6-19 .....	157
Figura 6-20. Caras atravesadas por el plano de simetría.....	158
Figura 6-21. Caras Simétricas cuya relación de simetría de sus vértices se establece en sentido inverso.....	158
Figura 6-22. Caras Simétricas cuya relación de simetría de sus vértices se establece en el mismo sentido.....	159
Figura 6-23. Generación de la mitad simétrica y posterior reconstrucción de la misma.....	161
Figura 6-24. Sistema de Referencia para la transformación de Simetría .....	162
Figura 6-25. a)Boceto inicial, b)dibujo lineal, c)polígono de simetría detectado.....	164
Figura 6-26. Modelo obtenido utilizando únicamente la condición de planicidad del plano de simetría como regularidad. ....	164
Figura 6-27. Modelo obtenido utilizando únicamente la condición de ortogonalidad de líneas que cruzan el plano de simetría con el propio plano como regularidad. ....	165

## Percepción Artificial de Dibujos Lineales

---

Figura 6-28. Modelo obtenido utilizando simultáneamente la planicidad del plano de simetría y la ortogonalidad entre las aristas que cruzan el plano de simetría y el propio plano de simetría como regularidad.....	165
Figura 6-29. Modelo obtenido utilizando la regularidad de simetría bilateral como único método de inflado. ....	166
Figura 6-30. Modelo con tres planos de simetría.....	168
Figura 6-31. Figura con dos planos de simetría Tiempo de reconstrucción partiendo de la imagen a) 3" y de la imagen b) menos de 1" .....	169
Figura 6-32 Figura con tres planos de simetría Tiempo de reconstrucción partiendo de la imagen a) 6" y de la imagen b) menos de 1" .....	169
Figura 6-33. Detección de planos de simetría en figuras correspondientes a modelos extruidos. ....	170
Figura 6-34. Modelos topológicamente iguales antes y después de eliminación de la mitad simétrica. ....	170
Figura 7-1. Modelo con un agujero pasante. ....	178
Figura 7-2 Modelo formado por dos bloques.....	178
Figura 7-3. Modelo representado mediante dos grafos no-conexos .....	178
Figura 7-4. a) Modelo con superficies curvas representado mediante aristas y contornos; b) el mismo modelo con las curvas necesarias para delimitar caras y c) representación de las caras determinadas.....	179



# Glosario de Símbolos

---

$\beta$  : ángulo de oblicuidad.

$\alpha$  : ángulo de rotación.

$\gamma$  : ángulo entre el eje de oblicuidad y el eje  $x$ .

$\sigma$  : desviación típica.

$\sigma^2$  : varianza.

$C (c_x, c_y, c_z)$ : centroide.

$(U V)$ : sistema de referencia oblicuo.

$(X Y)$ : sistema de referencia ortogonal.

$[C]$ : Matriz de Circuitos.

$[M]$ : Matriz Estándar de Momentos de Inercia.

$[MS]$ : Matriz de Parejas Simétricas.

$[N]$ : Matriz de Momentos de Inercia de la figura oblicua.

$[X]$ : Matriz de Incidencia.

$[Z]$ : Matriz de Interacciones.

$A$ : área.

$E$ : conjunto de aristas.

$F$  : Conjunto de caras.

$k_E$  :conectividad de aristas.

$k_V$  :conectividad de vértices.

## Percepción Artificial de Dibujos Lineales

---

$m_i$ : puntos medios de aristas.

$P = [x, y]$ : coordenadas del punto cuando está proyectado.

$Q = [u, v]$ : coordenadas del punto sin oblicuidad.

$S$ : transformación de Simetría 3D.

$T$ : función de oblicuidad.

$u_i$ : vectores unitarios.

$V$ : Conjunto de vértices.

$v_j$ : vértices.

$w_k$ : coeficiente de ponderación.

$X, Y, Z$ : ejes coordenados.

# Resumen

---

Los modeladores geométricos basados en percepción artificial de bocetos se han convertido durante la última década en una de las principales líneas de trabajo para obtener interfaces de diseño asistido por ordenador fáciles de usar y que potencien la creatividad del diseñador.

En el capítulo de antecedentes de esta tesis se presenta un análisis amplio de la evolución de la percepción artificial de modelos tridimensionales contenidos en dibujos bidimensionales, también denominada reconstrucción tridimensional. Pero hay que puntualizar que esta tesis se limita a la reconstrucción tridimensional *desde una única vista*; puesto que es generalmente aceptado que la reconstrucción 3D desde múltiples vistas ortográficas requiere métodos radicalmente diferentes.

Las técnicas de optimización matemática, que han sido las elegidas para la presente tesis, han sido utilizadas anteriormente con cierto éxito en los núcleos de reconstrucción 3D de algunos modeladores-a-partir-de-bocetos. La naturaleza de las formas que pueden ser reconstruidas con un enfoque basado en la optimización se extiende a poliedros cerrados, objetos abiertos de caras planas (planchas de metal doblado), modelos alámbricos (no-variedad) y superficies curvas más o menos simples (principalmente de revolución). Pero uno de los principales retos de la utilización de técnicas de optimización en la fase de reconstrucción tridimensional del modelo a partir del boceto sigue siendo la formulación matemática de las señales perceptuales. En efecto, las señales perceptuales, -también llamadas artefactos o regularidades, - son aquellas configuraciones o signos de una figura bidimensional que permiten interpretarla, reconocerla o imaginarla como la imagen de un objeto tridimensional. Hasta el momento muy pocas investigaciones han formulado matemáticamente estas señales perceptuales con rigor. En consecuencia, los trabajos conocidos han dejado definidas unas funciones objetivo un tanto pobres, cuyos ratios de éxito en la reconstrucción de modelos 3D a partir de dibujos 2D están por debajo de las

expectativas de los potenciales usuarios. Otro inconveniente de esta técnica es que los procesos de optimización matemática pueden llegar a ser muy lentos si no están bien diseñados e implementados; lo que los invalida para su uso en cualquier sesión interactiva; en particular en un sistema de diseño *asistido*, en el que las pausas largas bloquean el proceso creativo del diseñador. En otras palabras, el usuario no siempre obtiene el modelo 3D que él ha imaginado mientras bocetaba, y cuando lo obtiene no siempre es con la rapidez imprescindible para no perder la concentración requerida en un proceso de diseño.

En la presente tesis se han abordado los problemas descritos arriba. En primer lugar, se realizó un análisis de los métodos de optimización empleados por otros autores y la búsqueda, implementación y prueba de mejoras en el método de reconstrucción mediante optimización matemática. A continuación, y con la motivación de hacer una formulación de la función objetivo más eficiente para mejorar los tiempos de cálculo se comenzó por realizar un estudio de la naturaleza y la implementación numérica de las regularidades propuestas por autores anteriores. Dicho estudio descubrió algunas regularidades que consideramos “falsas”, y permitió detectar algunas imperfecciones o errores en las formulaciones de algunas regularidades.

Después de repasar todas las regularidades propuestas por otros autores, advertimos del gran potencial de una nueva regularidad: la simetría del modelo. En efecto, utilizar la simetría mejora el comportamiento de la optimización y es un concepto fundamental que la percepción visual humana utiliza. Además, muchos objetos fabricados son simétricos, tanto porque hace que los objetos sean más fáciles de interpretar y de fabricar, como por los requerimientos funcionales y estéticos. En otras palabras, la característica de *simetría del modelo*, frecuentemente es una intención explícita de diseño. En consecuencia, como aportación original de la tesis, se presenta un nuevo método para determinar automáticamente los planos de simetría de objetos poliédricos partiendo de una vista general de tipo axonométrico. El proceso se realiza en 2D (sobre el boceto), así que el resultado ofrece *planos oblicuos de simetría*: planos de simetría real vistos desde alguna dirección (desconocida).

En el método propuesto en esta tesis un plano de simetría es el plano que contiene a un circuito que representa la sección que el plano de simetría le produce al modelo. Por tanto, un plano de simetría se obtiene de un conjunto de “líneas de simetría”: una secuencia cerrada de ejes de simetría facial oblicua y aristas que forman un *polígono de simetría*. Para detectar los polígonos de simetría se aplica un conjunto jerarquizado de reglas perceptuales. El método requiere la detección previa de los ejes de simetría facial oblicua,

que, a su vez, precisa del reconocimiento previo de las caras del modelo. Ambas detecciones previas deben realizarse sobre el boceto 2D. Los métodos existentes para detectar ejes de simetría facial oblicua y caras han sido mejorados y adaptados para cubrir las necesidades propias de la detección de polígonos de simetría.

Detectar los planos de simetría no es suficiente para el objetivo de modelar a partir de bocetos. La simetría detectada debe utilizarse posteriormente para reconstruir modelos que resulten simétricos. Por ello, se ha desarrollado e implementado la nueva regularidad para la optimización basada en una nueva formulación de un modelo simétrico. La regularidad de simetría de modelo tiene capacidad para guiar el proceso de reconstrucción mediante optimización *por sí sola*. Numerosos ejemplos demuestran su buen comportamiento.

Una extensión de este método se propone para los bocetos en que se detecta más de un plano de simetría. Consiste en utilizar uno de los planos de simetría para el proceso de optimización y el resto de planos de simetría para simplificar el modelo a reconstruir. En otras palabras, uno de los planos de simetría es utilizado como regularidad y el resto como restricciones. El resultado es que se obtienen modelos más fieles a la intención del diseñador (más simétricos), y con un menor esfuerzo de cálculo.

Como complemento a esta tesis, se ha elaborado una primera versión “docente” de la aplicación de modelado mediante bocetos. Y como desarrollos futuros se presenta la posibilidad de construir los modelos sólidos utilizando directamente las señales perceptivas captadas en los bocetos para controlar motores geométricos con capacidades paramétricas/ variacionales.



# Abstract

---

*During the last decade sketching-based geometric modellers have become in one of the main work lines to obtain easy to use computer aided design interfaces and providing a tool for strengthening designer's creativity.*

*In the chapter of this thesis dedicated to Backgrounds, a wide analysis of the studies' evolution about artificial perception of three-dimensional models held in two-dimensional drawings, also called three-dimensional reconstruction, is presented. But have to be notice that this thesis is aimed at single view reconstruction, since it is generally accepted that 3D reconstruction from multiple orthographic views requires radical different approach.*

*Mathematical optimisation techniques have been chosen for this thesis. Optimisation has been previously used with some success in 3D reconstruction's kernels for sketch-based modelling approach. The nature of shapes that can be reconstructed with this optimisation-based approach extends, at present, to polyhedral blocks (manifolds), origami objects (sheet metal), wire frame models (non-manifold objects) and curve surfaces more o less simple (mainly solids of revolution). Nevertheless, one of the main challenges in optimisation-based 3D reconstruction is the mathematical formulation of perceptual cues. Effectively, perceptual cues, -also called artefacts or regularities, - are those signs or configurations in a two-dimensional figure allow us to interpret, imagine or recognize it as the image of a three-dimensional object. At present, very few researches have formulated mathematically these perceptual cues with precision. Therefore, studied works have lead too poorly defined objective functions, and, hence the success rate, in 3D models' reconstruction from 2D drawing, is below*

*the expectations of potential users. Another threshold of this approach is mathematical optimisation processes can be very slow if they are not well-implemented or designed. That makes the process invalid for an interactive session, especially in a system of aided design, where long pauses obstruct the designer creative process. In other words, the user does not always get the 3D model that he/she thought whilst he/she was sketching. Moreover, when the 3D model is obtained, sometimes is without the required speed for not losing the concentration required in the designing process.*

*In the present thesis, described problems above are considered. In first place, it was made an analysis of optimisation methods used by other authors and the search, implementation and checking of improvements in the reconstruction based optimisation approach. With the aim of improving computational times and for becoming the objective function more efficient, the work went on an intensive study about regularities nature and their numerical implementation already proposed by previous authors. This study let to discover some regularities considered “false” and allowed to detect some imperfections or mistakes in the formulation of some regularities.*

*After a revision of all regularities proposed by other authors, a big potential is detected in the formulation of a new regularity: model’s symmetry. In effect, using symmetry improves optimisation performance, since it is a fundamental concept which human visual perception utilizes. Furthermore, many man-made objects are symmetric, both because this makes them easier to interpret and manufacture, and because of functional and aesthetic requirements. In other words, the feature of model symmetry is often an explicit design intent. Hence, as an original contribution of this thesis, a novel method for determining the planes of symmetry of polyhedral objects is presented, starting with a two-dimensional axonometric-like view. The computation is performed in 2D (in the sketch), so the resulting output comprises skewed planes of symmetry: real planes of symmetry viewed from some (unknown) viewing direction.*

*In the approach described in this thesis, a symmetry plane is the plane containing a lines circuit representing the section that the plane of symmetry makes to the model. Therefore, a symmetry plane is obtained from a set of “symmetry lines”: a closed sequence of axes of skewed facial*



*symmetry and edges that meet each other forming a polygon of symmetry. The detection of polygons of symmetry is carried out applying a hierarchical set of perceptual rules. The method requires the prior detection of axes of skewed facial symmetry, which, in turn, requires edges bounding faces to be previously recognised in the two-dimensional sketch. The current methods for detection of axes of skew facial symmetry and faces have been improved and adapted for the polygons of symmetry detection.*

*However, the detection of planes of symmetry is not enough for the aim of modelling by sketches. Detected symmetry must be used next for model reconstruction, obtaining symmetric models. Subsequently, a new optimisation regularity based on a novel formulation of model symmetry has been developed and implemented. The regularity of model's symmetry is able to guide the optimisation of the reconstruction process by itself. Many examples prove its good performance.*

*An extension of this method is proposed for sketches with more than one plane of symmetry detected. It consists in using one of the planes of symmetry for optimisation process and the rest of planes of symmetry to simplify the model to reconstruct. In other words, one plane of symmetry is applied as regularity and the rest of them as restrictions. As a result, models closer (more symmetric) to designer intentions are obtained and calculation times reduced.*

*As complement of this thesis, a first educational version of the software for modelling by sketches has been developed. And as future work is presented the possibility of building solid models from perceptual cues detected in the sketches to control geometric engines with parametric/variational capabilities.*



# Capítulo 1. Introducción

---

## 1.1. EL PROCESO DE DISEÑO

---

Los ingenieros y diseñadores en las fases conceptuales del diseño necesitan herramientas de diseño muy flexibles, puesto que en ese momento no tratan con modelos exactos ni perfectamente dimensionados. En los primeros borradores de un diseño las formas son vagas y las dimensiones proporcionalmente aproximadas a las que se desea para el diseño final. En otras palabras, 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. Los croquis les permiten expresar formas sin atarse a criterios geométricos estrictos, y los bocetos les permiten expresar ideas parciales o inacabadas (dibujos *incompletos*). Ese es el lenguaje que se utiliza normalmente cuando se dibujan objetos que aún no están totalmente definidos para sintetizar nuevos diseños.

Un boceto realizado a mano alzada o un dibujo lineal es suficiente para que un ser humano sea capaz de reconstruirlo mentalmente en un objeto tridimensional. Barrow y Tenenbaum [BT81] muestran numerosos ejemplos de la capacidad que los seres humanos poseen para ver el mismo modelo 3D cuando se les muestra una imagen 2D. Sin embargo las aplicaciones informáticas actuales no han conseguido imitar esta capacidad.

Es por ello que después de que el diseñador consigue convertir los bocetos iniciales en una geometría totalmente definida y cuantificada, debe dibujarla (creando sus “planos” mediante una herramienta CAD 2D) o modelarla en una de las actuales herramientas gráficas de modelado 3D que permiten al diseñador crear modelos virtuales en 3D. La necesidad de construir y especificar la geometría del objeto después de diseñarlo se debe a que los sistemas CAD actuales son ciegos para ver el croquis; es decir, carecen de la *interface* necesaria para interpretar un croquis ya acabado, o para permitir croquizar directamente en

el ordenador. Son excesivamente rígidos para aceptar bocetos o dibujos ambiguos con formas aproximadas.

La cuestión es que aún no se ha conseguido completar un programa, comercial y fácil de utilizar, que sepa interpretar lo que el ser humano sí puede hacer automáticamente. Es decir, no hay en el mercado ningún programa CAD con la capacidad de reconstruir un modelo tridimensional partiendo de un dibujo lineal o un boceto a mano alzada.

Construir una herramienta informática capaz de interpretar tridimensionalmente y de forma automática bocetos o dibujos lineales es el objetivo global de esta tesis. A lo largo de la tesis se describirá cada uno de los aspectos necesarios para la reconstrucción tridimensional aunque en ella solo se abordarán con detalle las aportaciones propias originales de la autora.

## 1.2. LA RECONSTRUCCIÓN TRIDIMENSIONAL

---

La reconstrucción tridimensional de dibujos lineales estudia los métodos necesarios para convertir un dibujo lineal bidimensional en el mismo modelo tridimensional del objeto que para un ser humano representaría. Así la Reconstrucción 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.

El campo de la Reconstrucción Tridimensional puede enmarcarse dentro del área de la Interpretación de Dibujos Lineales; que debe ser entendida dentro del campo del Análisis Gráfico de Documentos, y a su vez ésta es una sección dentro de un campo más amplio como es la Visión Artificial ó Visión por Computador.

La Reconstrucción así entendida es el último proceso a realizar dentro del conjunto de procesos que conllevan la interpretación de dibujos lineales. Tal y como lo exponen Ablameyko & Pridmore [AP00] en su libro "*Machine Interpretation of Line Drawing Images: Technical Drawings, Maps and Diagrams*", los cinco pasos de la interpretación de dibujos lineales son los siguientes:

- ❑ Escaneado de los dibujos en papel y obtener así una imagen en mapa de pixels (binaria o en niveles de gris)
- ❑ Vectorización del mapa de pixels para obtener una representación en términos de primitivas gráficas simples.

- ❑ Identificación de construcciones de dibujo universales (por ejemplo entidades CAD: líneas, arcos, elipses, texto,...; u objetos cartográficos) de la imagen vectorial.

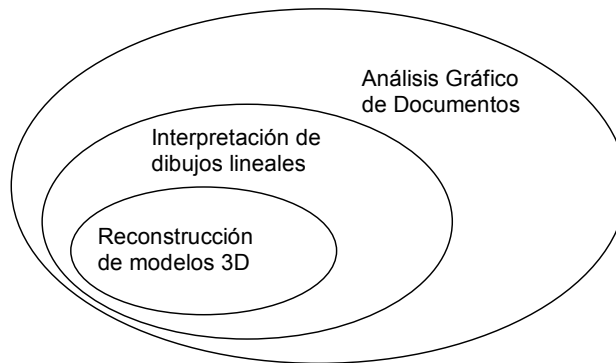


Figura 1-1. Ámbito de la Reconstrucción Tridimensional.

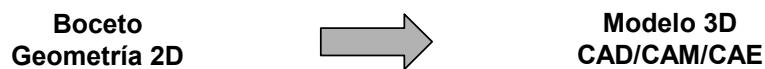
- ❑ Obtención de objetos específicos de los dibujos bidimensionales, como podría ser una librería CAD o GIS en la que se identificaran: dimensiones, símbolos, etc...Creación de “escenas” que hacen de forma explícita la geometría y las relaciones entre los objetos obtenidos.
- ❑ Reconstrucción de una descripción tridimensional de los objetos que se encontraban en el dibujo original. La resolución de este problema implica la determinación geométrica y las relaciones topológicas de todas las partes atómicas que componen un objeto.

El procedimiento dado por Ablameyko y Pridmore para la interpretación de dibujos lineales no es la única forma de introducir datos en el proceso de reconstrucción. Los dibujos lineales reconstruidos en esta tesis son bocetos o proyecciones de objetos desde un punto de vista general, así que la entrada al proceso de reconstrucción puede ser, sin pérdida de generalidad, dibujos vectorizados, bocetos realizados sobre una interfaz de bocetado, o simplemente dibujos ya realizados en formato electrónico (.dwg, .dxf, .dgn,...). Y su salida consiste en modelos tridimensionales tales como los que se utilizan en aplicaciones CAD/CAM/CAE.

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 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 1-3.

A continuación se expone de modo resumido cada una de las tareas principales que forman parte de la interpretación de dibujos lineales.



*Figura 1-2. Objetivo principal de la Reconstrucción 3D*

### 1.2.1. Datos de partida: Vectorización y Bocetado on-line

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. Mientras, las aplicaciones de manipulación de imágenes están orientadas a trabajar con las imágenes almacenadas en formato matricial, o mapa de píxels.

La inmensa mayoría de los periféricos de representación tienen un funcionamiento orientado al almacenamiento en mapa de píxels. Por ejemplo, los escaners y cámaras fotográficas digitales permiten adquirir imágenes utilizando un almacenamiento de la imagen matricial. Por ello, las aplicaciones no pueden utilizar ningún recurso que requiera conocer las propiedades geométricas de la imagen de partida.

La vectorización permite convertir una imagen de mapa de bits en figuras geométricas. Es decir, permite identificar puntos, rectas y curvas en una imagen dada en mapa de píxels. La vectorización de imágenes ha sido estudiada por un gran número de autores como: [CLD96], [JV97], [NL90], [PLC98], [RM95], [VT95], [Dor95a], [Dor95b], [GCG98], [HF94], [WD98]. En algunos casos incluyen la Discriminación de Textos-gráficos [AJM97], [LK94] y detección de símbolos y textos de las cotas para su posterior interpretación: [DL97], [DV98] y [Dor92]. Pero éste 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 interacción notable por parte del usuario [GCG98].

Los dibujos destinados a ser reconstruidos además de poder ser realizados en papel y posteriormente vectorizados, pueden ser realizados con una aplicación comercial CAD en 2D y posteriormente leídos por la aplicación de reconstrucción tridimensional. La vectorización era la vía preferida cuando se entendía que el propósito principal de la reconstrucción tridimensional debía ser trasladar a formato electrónico la información histórica disponible en formato papel.

En la actualidad este problema se está resolviendo por otros medios, y ha dejado de ser un objetivo de la reconstrucción. La opción de usar dibujos delineados por ordenador no es una vía de entrada habitual, porque no es razonable empezar delineando para acabar obteniendo un modelo. Esta opción se usa principalmente como vía rápida de entrada de ejemplos de prueba para comparar aplicaciones de reconstrucción. También es útil para recuperar dibujos bocetados en sesiones anteriores y guardados en formatos compatibles.

El tercer camino, que es el más natural cuando el propósito es disponer de una interfase amigable para diseñar, consiste en la introducción de los datos aplicando metodología de interfaces caligráficas. Este tipo de interfaces están basadas en la utilización de una tableta LCD o un Tablet PC con un lápiz electrónico que permite al usuario realizar bocetos directamente sobre la pantalla. Como solución transitoria hasta la generalización del hardware apropiado, también pueden realizarse los bocetos con un ordenador con pantalla de cualquier tipo y utilizando el ratón como si de un lápiz se tratara.

Bocetar directamente en el ordenador requiere también un software apropiado, que puede ser muy simple o muy sofisticado. Como mínimo se debe disponer del software necesario para convertir las acciones del usuario en dibujos (vectorizados o no) y/o órdenes. Los sistemas de *modelado basados en gestos* se centran en el concepto de utilizar gestos predefinidos y la introducción de comandos mediante el lápiz. Es decir, constan de gestos que se interpretan como comandos para generar sólidos partiendo de las secciones 2D dibujadas de los mismos [ZHH96], [EHB97], [BZ 98], [QWJ00] y [PJB00]. Las técnicas de reconstrucción de modelos a partir de bocetos, -que son las que se han utilizado en el desarrollo del software desarrollado para chequear los resultados de esta tesis,- no utilizan gestos para controlar el flujo del programa, sólo requieren que los trazos se interpretan como parte del dibujo, y los comandos se introducen a través de la metodología WIMP

(Windows, Icons, Menus and Pointing) [QWJ00], [JM92], [IMK97], [SG98], [OK01], [NJC02] y [NCC03].

El sistema de reconstrucción tridimensional elegido tiene como dato de entrada una única vista. La vista de entrada debe ser la representación axonométrica de un objeto o un boceto aproximado del mismo. No se le da al sistema ninguna información de cómo es el objeto 3D, el tipo, o su posición relativa.

El dibujo inicial se asume que está representado en una proyección general del objeto. Es decir, la proyección se ha realizado desde un punto de vista arbitrario. Esto implica que desde el punto de vista desde el cual se proyecta el modelo no deben existir coincidencias accidentales de líneas o vértices.

La aplicación está diseñada para que los bocetos o dibujos de entrada estén representados con todas sus aristas y vértices, como si las caras del objeto fueran transparentes. Es decir, el dibujo de entrada debe ser un modelo alámbrico.

Por lo tanto la entrada al sistema es un simple dibujo de líneas 2D, de forma que cada línea será una arista del objeto 3D y cada unión de éstas sea un vértice del objeto. La similitud con un grafo de entidades conectadas está clara. Es por ello, que en muchos momentos a lo largo de este documento se hablará de la imagen de partida como de un grafo conexo, e incluso se le aplicarán conocimientos provenientes de la teoría de grafos.

### **1.2.2. Reconstrucción 2D.**

En esta fase se enmarcan las técnicas de filtrado de los datos de entrada al sistema de reconstrucción, ya sea 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. Puesto que la forma de introducir los datos es variada, también estas técnicas son amplias. De este modo, la reconstrucción 2D engloba todas las técnicas que se pueden aplicar al documento de partida y lo convierten en un dibujo geométrico de partida consistente para su posterior reconstrucción 3D.



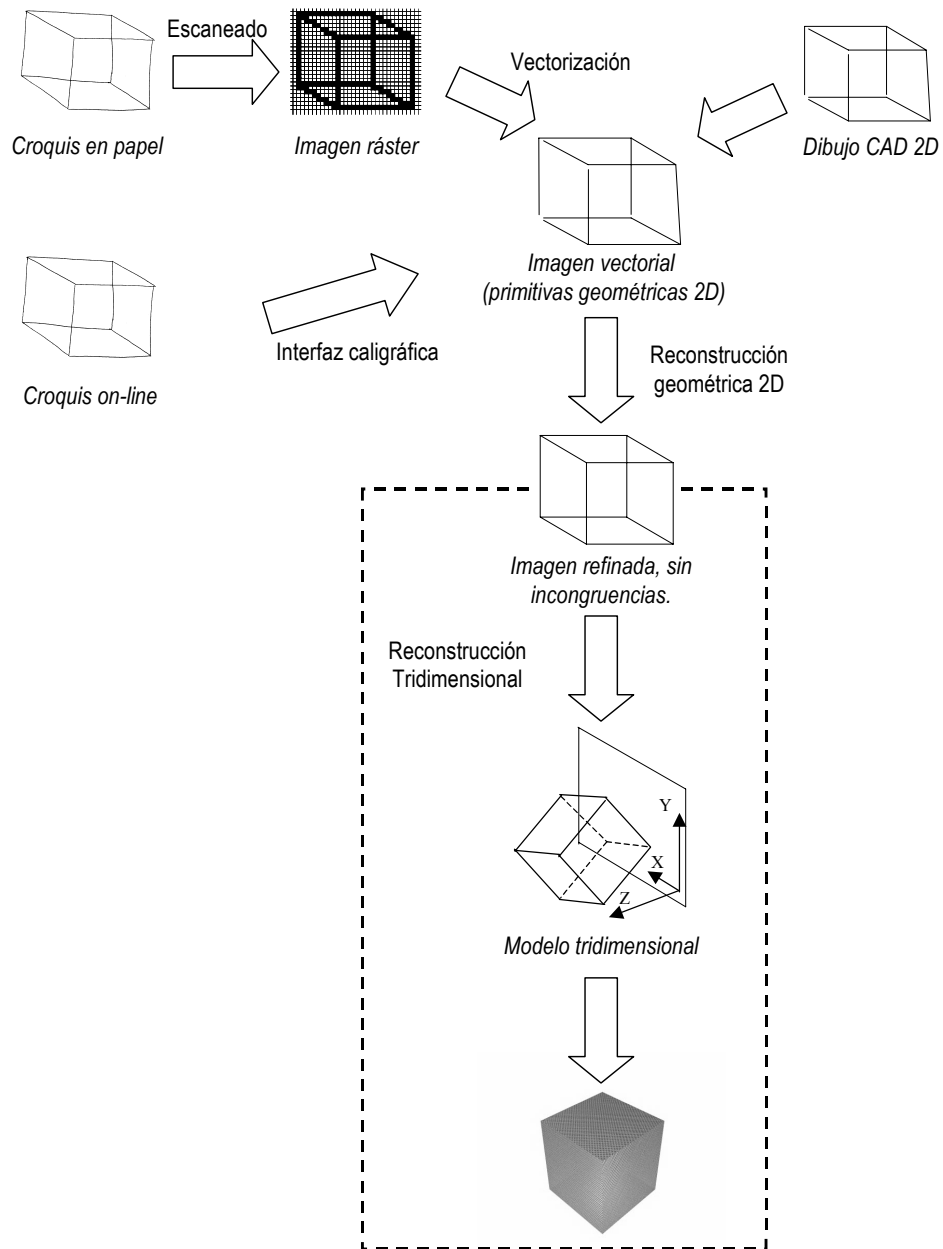


Figura 1-3 Fases de la reconstrucción de dibujos de ingeniería

## Percepción Artificial de Dibujos Lineales

---

Así dentro de las tareas realizadas en la reconstrucción 2D quedan incluidas eliminar las imperfecciones provocadas por el proceso de vectorización (como por ejemplo la detección y eliminación de las líneas provocadas por los pliegues del papel o las sombras causadas por su envejecimiento) o los errores de la delineación por ordenador, como por ejemplo las líneas dobles, las aristas mal conectadas, etc. Pero su tarea principal 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 constata 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].

Si el dibujo proviene de un plano vectorizado, la reconstrucción 2D sólo puede hacerse off-line. Es decir, que la reconstrucción se tiene que hacer sobre el dibujo totalmente acabado, y sin disponer de información sobre las acciones llevadas a cabo para generarlo. Pero, en el caso en el que se posea una interfaz caligráfica donde bocetar el dibujo de partida, la reconstrucción 2D puede realizarse off-line u on-line. Esta última puede aprovechar la información deducida de las acciones realizadas por el usuario, pero suele renunciar a captar la intención global del diseñador, que sólo puede conocerse al acabar el boceto. En ese caso una de las principales funcionalidades que debe tener la interfaz caligráfica es la conversión dinámica de líneas en segmentos rectos (conversión de la Figura 1-4-a a la Figura 1-4-b). También pueden coexistir reconstrucción on-line de algunos aspectos y off-line de otros (paso de Figura 1-4-b a Figura 1-4-c).

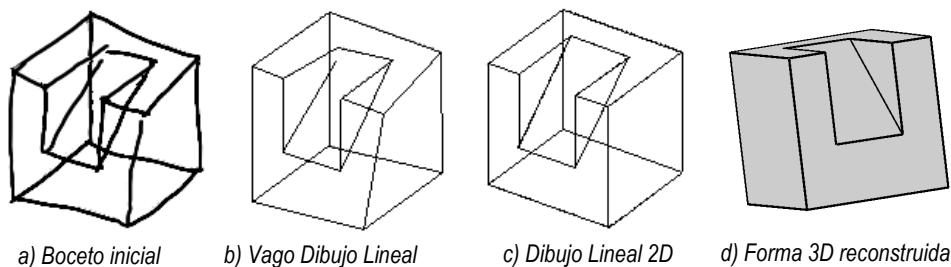


Figura 1-4. Cuatro estados en el modelado mediante bocetado.

En resumen, la reconstrucción on-line suministra retroalimentación inmediata al usuario, porque trabaja mientras el usuario dibuja el boceto y su uso ofrece una mayor integración en la interfaz caligráfica. Mientras la reconstrucción off-line reforma el boceto por completo cuando el usuario ha finalizado el dibujo y puede suministrar un análisis más detallado del boceto del usuario, descubriendo intentos del diseñador que pueden ser difíciles de detectar en un sistema on-line. Ambas estrategias son utilizadas por diferentes sistemas que se están desarrollando en la actualidad.

### **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 habitual 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 comenzó siendo muy heurística, aunque en la actualidad se incrementa el uso de reglas de la percepción visual tomada de la psicología de la percepción.

La reconstrucción 3D suele estar compuesta por dos pasos: un primer paso nos ofrecería una aproximación al modelo 3D (modelo tentativo) y un segundo paso, llamado refinado, que incluye las técnicas precisas para obtener un modelo que sea acorde con todas las especificaciones dadas en la imagen.

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

## **1.3. OBJETIVOS DE LA TESIS**

---

El objetivo principal de este trabajo se orienta a la creación de una herramienta de diseño que permita reconstruir tridimensionalmente objetos a partir de bocetos, permitiendo integrar las fases iniciales de diseño en el entorno de los sistemas CAD/CAM/CAE. Es

decir, se quiere construir una aplicación automática que ofrezca el modelo tridimensional que un ser humano percibiría al observar dicho boceto partiendo de un boceto bidimensional que representa de forma aproximada la proyección de un objeto.

El resultado debe ofrecer una correcta topología en 3D y unas cualidades geométricas aproximadas: paralelismo, perpendicularidad, proporcionalidad, simetría,...

Dada la amplitud del ámbito de trabajo, puesta de manifiesto en §1.2, esta tesis marca sus objetivos específicos en la fase de reconstrucción 3D, y más concretamente en la detección de ciertas características en los dibujos 2D, así como en la detección y formulación de algunas regularidades. Es decir, la tesis se centra en la obtención de manera automática de modelos geométricos, utilizando un algoritmo de optimización cuya formulación depende de ciertas características detectadas en la imagen 2D. Si bien, transcurre en paralelo con otros trabajos de investigación del grupo de investigación REGEO [Reg00] que abordan otros aspectos de nuestro objetivo general.

## 1.4. ESTRUCTURA DE LA TESIS

---

La presente tesis se encuentra estructurada en siete capítulos, según el siguiente esquema:

### **Capítulo 1: Introducción**

En este capítulo se determinan los principales rasgos del campo de actuación (reconstrucción tridimensional), se resume el objetivo general del trabajo y se explica la propia estructura de la tesis.

### **Capítulo 2: Antecedentes de la Reconstrucción 3D**

El estudio de la situación actual de los algoritmos de reconstrucción es el objeto del presente capítulo. Así, a lo largo del mismo se repasa el estado del arte de la Reconstrucción de modelos tridimensionales partiendo de dibujos lineales, nombrando los trabajos de los autores más relevantes dentro de este campo, clasificándolos y resaltando sus semejanzas y diferencias.

### **Capítulo 3: La Reconstrucción y las Regularidades**

En este capítulo se da una imagen clara de cuáles son los parámetros y los algoritmos generales de reconstrucción utilizados en esta tesis. Del mismo modo se ofrece la

formulación de todas las regularidades utilizadas o testadas en el proceso de reconstrucción.

#### **Capítulo 4: Detección de Caras de Poliedros en Representaciones 2D**

Una de las detecciones más importantes a realizar en los dibujos 2D, antes de la reconstrucción, es la detección de caras. En este capítulo se hace un repaso de los algoritmos más relevantes destinados a la detección de caras en grafos conexos, así como la aplicación de las mismas, mejoras y sus resultados.

#### **Capítulo 5: Simetría Facial Oblicua**

Se presenta un estudio intensivo de la regularidad de la regularidad de Simetría Facial Oblicua. Se ofrece un método para la detección de los todos los ejes de simetría oblicua para cada una de las caras 2D previamente detectadas. Se estudian sus posibilidades en la mejora del proceso de reconstrucción aplicándose como una regularidad, y se aporta una formulación con este objetivo.

#### **Capítulo 6: Simetría de Modelos 3D Detectada en el Espacio 2D**

Se aporta un procedimiento propio para el cálculo de los polígonos de simetría como el conjunto cerrado de líneas que representan un plano de simetría en una imagen proyectada 2D. También se ofrece la formulación de dos enfoques diferentes cuyo objetivo es la obtención de beneficios de la detección de la regularidad de simetría de modelos en el proceso de reconstrucción.

#### **Capítulo 7: Conclusiones y Trabajo Futuro.**

Para finalizar se ofrecen las conclusiones del trabajo realizado en esta tesis, y cuáles son los trabajos futuros que la propia tesis ha hecho viables a corto plazo, a juicio de la autora.

## **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

## Percepción Artificial de Dibujos Lineales

---

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].

Posteriormente se implementó una segunda versión del programa para validar los métodos propuestos en la Tesis “Reconstrucción Geométrica de Sólidos Utilizando Técnicas de Optimización” [Con01] y comparar con otros métodos de planteamiento análogo propuestos por otros autores.

Ahora una tercera versión, en la que se ha integrado una interfaz de bocetado on-line, ha sido necesaria para el desarrollo y el contraste de las propuestas de la presente tesis.

REFER ha sido desarrollado utilizando Microsoft's Developer Studio, para ejecutar en ordenador personal (plataforma PC) y bajo sistema operativo Windows 2000. La programación se ha realizado en C++ y la interfaz gráfica con el usuario se ha realizado mediante las librerías de win32 a través de Visual C++, y para visualizar modelos se eligió Open GL [OpenG].

Todas las pruebas realizadas para la elaboración de esta tesis han sido realizadas en un Pentium III 1000MHz/133Mhz FC, con 256Mb de memoria RAM.

El proceso en REFER comienza con una sesión interactiva donde el diseñador dibuja bocetos 2D de forma on-line ayudado por el ratón como si de un lápiz se tratara. Así se genera vagos dibujos parecidos a grafos que en un primer paso son convertidos en dibujos lineales 2D mejorados (que pueden guardarse en formato DXF o IGES) y posteriormente en formas 3D. El formato de salida es DXF, IGES o STEP que contiene un modelo alámbrico o de superficies.

# **Capítulo 2.**

# **Antecedentes de la**

# **Reconstrucción 3D**

---

## **2.1. INTRODUCCIÓN**

---

A lo largo de este capítulo se repasa el estado del arte de la Reconstrucción de modelos tridimensionales partiendo de dibujos lineales.

Se ha realizado una búsqueda bibliográfica intensiva en el ámbito de la Reconstrucción de dibujos lineales. Se han consultado documentos de temas relacionados, y se ha elaborado una base de datos con todos los documentos que finalmente se han considerado relevantes para conocer el estado de arte del tema estudiado.

Al final de este documento, en la Bibliografía se encuentran referenciados todos los documentos que aquí se citan y en la versión electrónica pinchando sobre el título de cada artículo se tiene acceso a los documentos originales en formato PDF. Además, en la mayoría de los casos el nombre de los autores aparece vinculado a su página personal, a donde se podrá acceder si se está conectado a internet. La mayoría son monografías, pero se incluyen algunos libros enteramente dedicados a la reconstrucción.

Con esta búsqueda bibliográfica se ha pretendido disponer de los trabajos de los autores más relevantes dentro de este campo, permitiendo una consulta ágil y eficiente de los mismos, y al mismo tiempo, tener también la posibilidad de clasificarlos y filtrarlos,

atendiendo a diferentes criterios que permitan resaltar todo tipo de semejanzas y diferencias.

Como conclusión, además de validar varios de los criterios ya plenamente establecidos por otros autores, se propone un nuevo criterio general de clasificación. En efecto, los trabajos previos de clasificación se han considerado correctos en general; salvo diferencias de criterio puntuales que se han recogido en las observaciones. Además, se propone un criterio de clasificación de todos los trabajos de reconstrucción existentes hasta la fecha, resumidos en gráficas en las que se muestran las influencias de unos autores sobre otros. También se ofrece una tabla resumen que engloba los aspectos diferenciadores más destacables de los diferentes autores.

## 2.2. CLASIFICACIÓN GENERAL DE LOS ALGORITMOS DE RECONSTRUCCIÓN 3D

---

La clasificación de la reconstrucción no puede atender a un único criterio. Tampoco es posible encontrar un conjunto de criterios independientes. Por tanto, hasta la fecha, lo habitual es clasificar los métodos atendiendo a tres aspectos diferentes:

- ❑ Naturaleza y forma de presentación de la figura de partida.
- ❑ Naturaleza y forma de presentación del modelo resultante.
- ❑ Estrategias aplicadas para realizar la reconstrucción.

Además, las posibles clasificaciones serán necesariamente interrelacionadas siendo la clasificación jerárquica la más habitual.

Wang y Grinstein [WG93] ofrecieron una buena clasificación del estado del arte hasta ese momento, diferenciando entre reconstrucción partiendo de una única vista y múltiples vistas. En el caso de reconstrucción a partir del sistema multivista, el siguiente nivel de clasificación lo realizan en función de la representación final del modelo sólido: CSG (geometría constructiva de sólidos), y B-Rep (modelos de fronteras). En este punto Wang y Grinstein siguen la clasificación que en su día ofrecieron Nagendra y Gujar [NG88], quienes también hicieron una recopilación bibliográfica del tema; pero únicamente desde el enfoque que parte de vistas múltiples.



En el caso de la reconstrucción partiendo de una única vista, el siguiente nivel de la clasificación hecha por Wang y Grinstein no se basa en el tipo del modelo resultante, sino en el enfoque aplicado para reconstruir. Por ello, distingue entre etiquetado, espacio del gradiente, programación lineal, perceptual e identificación de primitivas. En este trabajo se pretende ampliar esta clasificación, tanto en número de autores como en criterios de clasificación.

### 2.2.1. Vista Única – Vistas múltiples

Desde un primer momento siempre han habido dos vertientes que están bien definidas dependiendo de la representación elegida para los datos de entrada al proceso:

- ❑ una única vista: representación axonométrica o perspectiva cónica del objeto que se quiere representar,
- ❑ vistas múltiples: un dibujo lineal correspondiente a una representación de un objeto en sistema diédrico multivista. Generalmente limitado a vistas principales (ortográficas).

La diferencia no estriba únicamente en el número de vistas, los métodos aplicados en ambos casos varían desde la base. Como se verá al ir estudiando a cada uno de los autores, en el caso de la reconstrucción partiendo de vistas múltiples nos encontramos ante un problema de emparejamiento. Los vértices o contornos proyectados en cada una de las vistas de la representación inicial deben ir emparejándose para identificar cuáles son los que pertenecen a un mismo vértice o superficie del modelo tridimensional. Por su parte la mayoría de los estudios realizados hasta el momento sobre reconstrucción partiendo de una única vista se basan en el estudio del grafo resultado de proyectar un objeto.

Dentro de esta clasificación se han dividido los algoritmos de reconstrucción partiendo de vistas múltiples según el tipo de representación final que ofrezcan, si se trata de Geometría Constructiva de Sólidos (CSG) o si es un objeto limitado por superficies (Boundary-Representation).

La clasificación de los algoritmos de reconstrucción partiendo de una única vista se ha realizado basándose en el enfoque que el autor le ha dado al algoritmo.

Se puede concluir que la primera clasificación que se hace, y la más general, es el tipo de datos de partida, haciendo referencia en este punto a si la información de entrada se

## Percepción Artificial de Dibujos Lineales

---

compone de una única vista o de varias. En efecto, en la Tabla 2-1 se observa que la clasificación en vista única o múltiples vistas influye en otras clasificaciones.

Vista Única	Múltiples Vistas
Etiquetado	
Espacio del Gradiente	
Programación Lineal	CSG
Progresivo	B-Rep
Identificación de Primitivas	
Regularidades	

Tabla 2-1. Clasificación General de los algoritmos de reconstrucción.

## 2.3. RECONSTRUCCIÓN DE MODELOS 3D DESDE UNA ÚNICA VISTA

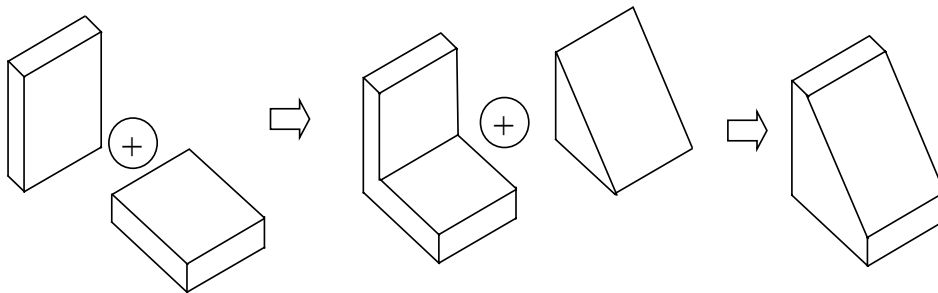
---

En 1963 en el Massachussets Institute of Technology (MIT) se puso en marcha un proyecto muy ambicioso que dio como resultado tres tesis. La primera "Sketchpad" [Sut63] marcó las bases de los primeros sistemas de delineación CAD. Una segunda tesis, "Sketchpad III" [Joh63] que puede ser considerada como el primer programa CAD 3D interactivo. La tercera tesis fue la de Roberts [Rob63]. Posiblemente fue Roberts, [Rob65] el primero que trató de interpretar los dibujos lineales aprovechando los primeros sistemas CAD que por entonces estaban emergiendo, considerándole así como uno de los pioneros en dos campos: los gráficos tridimensionales por computador, y la Visión por Computador.

Roberts tomaba como entrada a su sistema una fotografía, la cual era convertida automáticamente en un dibujo lineal, y a partir de éste obtener el modelo tridimensional. Se asumía que la fotografía estaba realizada en perspectiva y que lo que había representado era un conjunto de objetos que podían ser reconstruidos mediante la unión de bloques poliédricos conocidos. En síntesis, lo que hace Roberts es comparar partes del dibujo lineal con una lista de modelos predefinidos, y luego se ayuda de los mínimos cuadrados para

asegurar la identificación, utilizando la transformación más apropiada. Este modelo es muy similar al que posteriormente se desarrollaría por Wang y Grinstein en 1989, y que se vino a llamar enfoque de la identificación de primitivas, a pesar de que en la época de Roberts aun no existía el concepto de primitiva. En la Figura 2-1 se muestra uno de las escenas más sencillas que Roberts presenta en su tesis, donde se ve cuáles son los bloques que identifica y obtiene así la figura final que es la que la fotografía quería representar.

Guzmán [Guz68] desarrolló un sistema llamado SEE con el objetivo de analizar e interpretar los dibujos lineales de poliedros, pero, a diferencia de Roberts, sin modelos predefinidos. Su método tenía como objetivo descomponer automáticamente en regiones un dibujo lineal representando un conjunto de objetos, de forma que cada región corresponde a un objeto. Utilizó configuraciones de las uniones de líneas y las relaciones entre nodos para establecer una interpretación de una escena de objetos múltiples, donde algunos objetos pueden estar parcialmente ocultos. Sus algoritmos estaban basados puramente en reglas heurísticas, para las cuales no se suministra fundamentos lógicos o geométricos. Pero a pesar de que este sistema era bastante frágil (se le podía engañar con facilidad), y que tendía a tratar los agujeros de los objetos como si se tratara de objetos separados, su trabajo a influenciado en gran medida en la investigación realizada posteriormente. Este trabajo fue el principio de los esquemas de "interpretación" en los que se basa el etiquetado de líneas.



*Figura 2-1. Modelo de Roberts, 1963*

Roberts y Guzmán son los promotores de la reconstrucción de modelos tridimensionales de dibujos lineales partiendo de una única vista. Es por ello por lo que no los he incluido dentro de la clasificación que comienza a continuación, puesto que creo que merecen un puesto especial dentro de toda la bibliografía recogida en este trabajo. Aunque no hay ningún trabajo anterior a éste que los incluya dentro de la clasificación "ordinaria", en los esquemas que en este trabajo se muestran, sí están incluidos. Roberts aportó la idea original a lo que

posteriormente se denominó método de identificación de primitivas (§2.3.5), y Guzmán hizo lo propio con los métodos de etiquetado (§2.3.1). Por tanto, aunque es discutible clasificarlos en dichos métodos atendiendo a criterios puramente cronológicos, entiendo que son sin duda los precursores de dichos métodos a los cuales aportaron la idea original.

### 2.3.1. Enfoque del Etiquetado de Líneas

Los métodos basados en el etiquetado de líneas, más que *reconstruir* son una forma de *interpretar* los dibujos lineales, puesto que se trata de métodos de asignación de significado a las líneas. El procedimiento utilizado por estos métodos se basa fundamentalmente en tener un diccionario de uniones, de forma que se comparan las uniones del dibujo lineal con las del diccionario, y dependiendo del tipo de unión de la que se trate, las líneas que convergen en esta unión serán etiquetadas de un modo u otro.

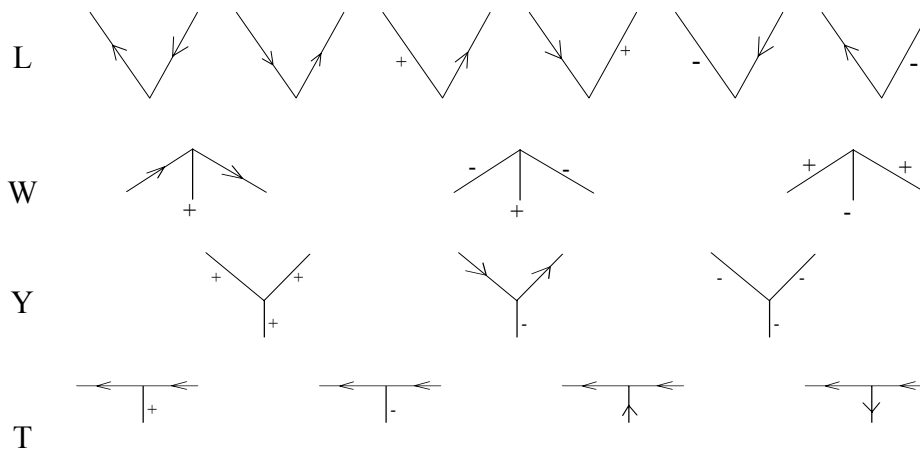


Figura 2-2. Diccionario de los tipos de uniones y sus posible etiquetado para un vértice triédrico.

En 1971, Huffman y Clowes, cada uno por separado [Huf71] [Clo71], exploraron las esquinas de un cubo y clasificaron sus proyecciones considerando todas las posibles esquinas vistas desde todos los posibles puntos de vista. Ellos encontraron que sólo habían cuatro posibles tipos de vértices triédricos (donde convergen tres aristas, o tres caras): Tipo L, tipo T, tipo W (o flecha) y tipo Y (o tenedor). Etiquetando estas cuatro uniones, se dieron

cuenta que sólo un número limitado de etiquetas se podían asignar a los segmentos que se encuentran alrededor de una unión: *cóncavas* (-), *convexas* (+) u *oclusivas* ( $\leftarrow$ ,  $\rightarrow$ ). Cuando un dibujo lineal es etiquetado, a cada línea se le asigna uno de estos tres significados posibles.

Para asignar a un segmento la etiqueta *cóncava* (-) o *convexa* (+) se toma como referencia el ángulo mirando desde fuera al dibujo lineal. Las etiquetas *oclusivas* corresponden a líneas como las de contorno, que comparten una arista y dos caras: una oculta y otra vista (la que oculta), y se representan con una flecha ( $\leftarrow$ ,  $\rightarrow$ ) de modo en que si miramos en la dirección de la flecha, la cara vista queda a la derecha, y la oculta a la izquierda.

Cuando se trabaja con objetos que poseen superficies curvas se añade otro tipo de etiqueta: la doble flecha ( $\rightarrow\rightarrow$ ). La doble flecha se utiliza para designar *ramas*, de modo que si nos movemos en la dirección de las flechas a lo largo de la rama, la superficie queda a la izquierda, la visual utilizada para la proyección es tangente a la superficie en todos los puntos de la rama.

De esta forma, Huffman y Clowes tan sólo por inspección enumeraron todas las configuraciones y las registraron en un diccionario de uniones para objetos con vértices triédricos.

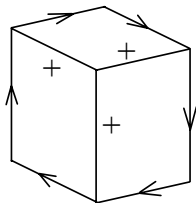


Figura 2-3. Figura de superficies planas etiquetada.

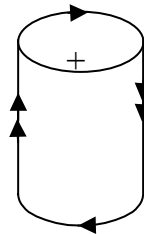


Figura 2-4. Figura con superficies curvas etiquetada.

La aplicación de Clowes, llamada OBSCENE hizo implícito lo que estaba de modo explícito en SEE ([Guz68]).

Malik [Mal87] desarrolló y probó la validez de un esquema más extenso para una clase más general de objetos, incluyendo superficies curvas. Presenta un esquema rigurosamente matemático para etiquetado de dibujos lineales de escenas compuestas por objetos sólidos, regulares, opacos y limitados por superficies sin ningún tipo de marca o textura

## Percepción Artificial de Dibujos Lineales

sobre ellas. Desde un punto de vista general clasifica todas las posibles uniones bajo proyección ortogonal. Y para reducir la multiplicidad de etiquetas de líneas propuso una regla a seguir: en cada vértice se seleccionarían aquellas etiquetas que correspondieran al mínimo número de caras diferentes encontrándose en el vértice.

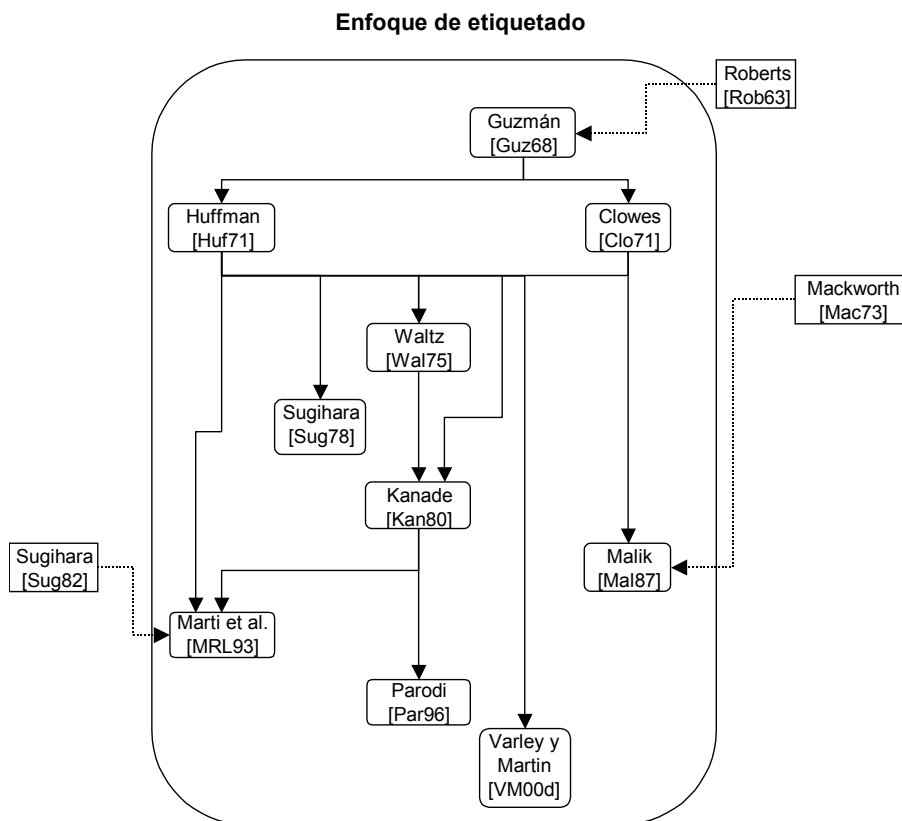


Figura 2-5. Gráfico resumen del método de etiquetado.

La clase de objetos que pueden ser tratados mediante etiquetado es ampliada a dibujos lineales que representan *objetos origami* por Kanade [Kan80]. Los objetos llamados origami, son aquellos no sólidos, cuyas paredes que definen su contorno se suponen de espesor despreciable. Él llama a su mundo de *objetos Origami*, por su parecido con el arte tradicional Japonés de cortar y doblar hojas de papel (papiroflexia), pero su aplicación puede ser referida tanto a objetos realizados con papel u otros de pequeños espesor, como son los ensamblajes de paneles de espesor despreciables como los de chapa metálica.

El trabajo de Kanade es seguido por Parodi [Par96] quien trata y afronta la complejidad del problema del etiquetado para los dibujos lineales de las escenas Origami. Parodi en su búsqueda de un algoritmo “biológicamente correcto”, demuestra que las técnicas de relajación utilizadas por Kanade [Kan 80] son ineficientes y su enfoque aporta nuevas restricciones geométricas que se deberían cumplir para este tipo de objetos.

Otros autores han querido también ampliar este método a otras representaciones como por ejemplo a objetos con sombras por Waltz [Wal75], a dibujos lineales que tienen líneas ocultas representadas (también llamados objetos transparentes) en Sugihara [Sug78], y ampliando el diccionario de etiquetas de Kanade también a aristas ocultas en Martí y otros [MRL93].

Obviamente, un dibujo sólo es correcto cuando tiene un *etiquetado lógico* (+, -, ←, →), un etiquetado que asigne una configuración que esté comprendida en el diccionario, asegurando que las dos posibles uniones de un segmento produzcan la misma etiqueta para ese segmento.

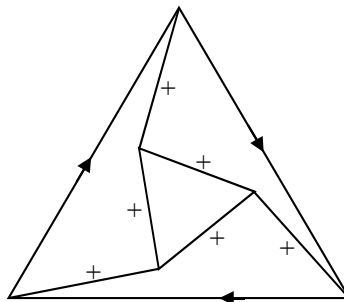


Figura 2-6. Modelo con etiquetado lógico

El principal inconveniente que surge con el esquema del etiquetado es que teniendo un etiquetado lógico, no se está garantizando que el dibujo sea una proyección de un poliedro. Por ejemplo, aunque todos los dibujos correctos tengan un etiquetado lógico y muchos dibujos incorrectos no puedan ser etiquetados, hay algunos dibujos que siendo incorrectos (no corresponden a la proyección de ningún objeto tridimensional), podrían aceptar un etiquetado lógico. Por lo tanto el etiquetado es condición necesaria pero no suficiente. Por ejemplo en la Figura 2-6, la figura representada es totalmente aceptable basándonos en el etiquetado, pero esta figura no representa la proyección de un sólido poliédricos.

Otro inconveniente importante es la necesidad de diccionarios de uniones preprocesados. Es imposible enumerar todas las posibles configuraciones de etiquetas para las uniones, porque en principio, no está limitada la cantidad de caras que pueden concurrir en ellas. Pero incluso si asumimos una cantidad máxima de caras que pueden concurrir en un vértice, el diccionario no es fácil de construir; aplicando técnicas de inspección es difícil listar todas las posibles uniones teniendo un número de caras dado. Un diccionario de este tipo para objetos con uniones tanto triédricas como tetraédricas se construye en [VM 00d], obteniendo 19 tipos de posibles etiquetados básicos. Acepta uniones del tipo L, T, W, Y, K, M, y X, haciendo referencia estos nombres al parecido que tienen la forma de las uniones y la forma de las letras, como se muestra en la Figura 2-7:

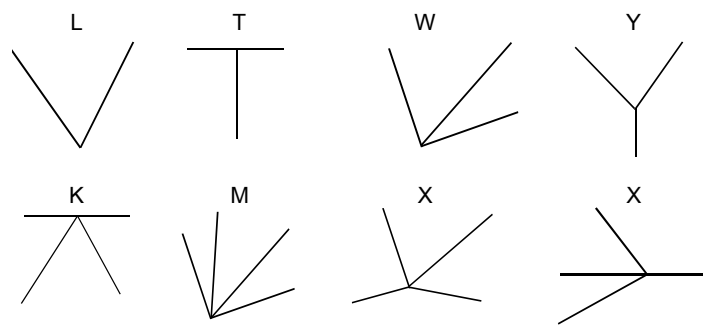


Figura 2-7. Tipos de uniones triédricas y tetraédricas

Los métodos de etiquetado de líneas, a pesar de ser los más estudiados, no han dado una total solución al problema de la reconstrucción. Proporciona una solución geométrica que por sí sola no es capaz de resolver todos los casos incluso cuando trata con objetos poliédricos. Son métodos que pueden ser útiles en ciertos pasos de la reconstrucción, pero que por sí solos son insuficientes.

### 2.3.2. Método del Espacio del Gradiente

Mackworth [Mac73] intenta encontrar un método que se base en reglas de coherencia que en general las superficies y las aristas deban satisfacer, y evitar el uso de interpretaciones predeterminadas de categorías particulares de uniones en el dibujo.



Asume que el objeto que representa el dibujo lineal es sólido y opaco y la proyección es ortogonal, y en la que todos los segmentos dibujados son rectos. Su algoritmo lo implementa en su programa POLY, que lo considera como una continuación del de Clowes (OBSCENE) y del de Guzmán (SEE).

POLY quería mejorar OBSCENE y por otra parte estuvo inspirado en la herramienta del *grafo dual* o *representación recíproca* que presenta Huffman, y que en su momento fue presentado como un mecanismo para obtener la orientación de las superficies que componen un objeto y determinar si se pueden construir.

Mackworth afirma que hay una relación entre el gradiente de una superficie poliédrica y las líneas en su proyección ortográfica. De esta relación define una figura recíproca en el espacio dual, de forma que:

- En el espacio dual un plano que en la realidad es de la forma  $a_x x + a_y y + a_z z + 1 = 0$  se representa como un punto  $(a_x, a_y, a_z)$  de forma que las coordenadas del punto en el espacio dual corresponden a las variables de la ecuación del plano real.
- De la misma forma, el dual de un punto  $(x, y, z)$  es un plano tal que  $(a_x, a_y, a_z)$  están sobre el plano si  $x a_x + y a_y + z a_z + 1 = 0$ .
- Si una línea en el espacio real está formada como la intersección de dos planos reales, su dual será la línea que pasa por los puntos en el espacio dual que representan a esos planos reales.

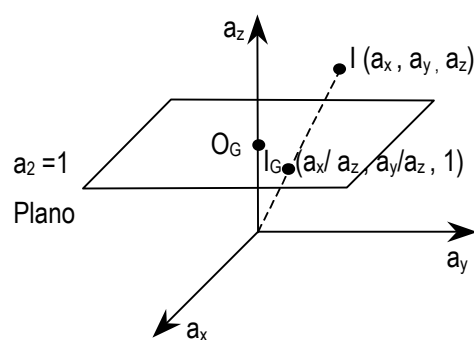


Figura 2-8. Proyección de un punto dual I en el espacio del gradiente.

## Percepción Artificial de Dibujos Lineales

---

En resumen, un grafo dual se obtiene dibujando un vértice por cada cara del grafo original; y por cada línea del grafo original que separa dos caras, en el grafo dual se dibuja una línea uniendo los dos vértices del grafo dual que representan a las dos caras.

El *espacio del gradiente* es un subespacio 2-D del espacio dual, que geoméricamente corresponde con la proyección de los elementos del espacio dual  $(a_x, a_y, a_z)$  sobre un plano  $a_z = 1$ , con el centro de proyección en el origen de coordenadas del espacio dual y utilizando  $(0,0,1)$  como origen del espacio del gradiente (Figura 2-8).

Si se considera una arista visible  $r$  producto de la intersección de dos superficies  $A$  y  $B$  (Figura 2-9), en el espacio del gradiente la configuración corresponde a dos gradientes ( $A_G$  y  $B_G$ ) unidos por una línea  $r_G$ , -que es la representación de la arista  $r$ -. Si se superpone el sistema de coordenadas del espacio del gradiente con el sistema de coordenadas utilizado para el dibujo lineal, Marckwoth comprueba que esta línea  $r_G$  resulta ser perpendicular a la línea del dibujo  $r$ . Además si la orientación en la que la línea dual une los gradientes es la misma en la que sus superficies originales correspondientes, entonces la arista es convexa (Figura 2-9-b). En el caso contrario, la arista se considera cóncava (Figura 2-9-c). Si la línea fuese plana, los dos gradientes debería coincidir.

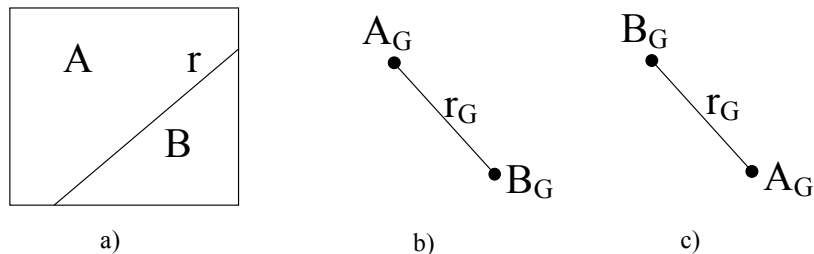


Figura 2-9. a)  $A$  y  $B$  son dos regiones en proyección unidas por una línea. La representación de las dos caras en el espacio dual, b) la arista que las une es convexa, c) la arista es cóncava.

Siguiendo este procedimiento, Mackworth interpreta los dibujos lineales mediante la construcción de las imágenes de cada plano en el espacio del gradiente. En resumen su programa POLY, primero determina el tipo de aristas: conectadas (cóncavas y convexas) y aristas ocultas, y así quiere encontrar la orientación de cada superficie y aristas del objeto.

Mackworth concluye que si un dibujo lineal no admite su figura recíproca, no es reconstruible, es decir, no corresponde con la proyección de un objeto. Sin embargo, ésta

es una condición necesaria, pero no suficiente: algunos dibujos no son realizables y se puede construir su imagen del gradiente.

A pesar de que no disponemos del trabajo original de Wei [Wei87], por estar escrito en chino, se puede encontrar una buena revisión de su artículo en Wang y Grinstein, del cual podemos extraer que extendió este análisis de Mackworth utilizando un espacio dual tridimensional. No sólo utilizó el gradiente de la cara para describir el espacio dual, sino que también utilizó un parámetro de profundidad. Wei dio cuatro restricciones adicionales que los puntos en el espacio dual deberían satisfacer para representar un posible poliedro, que representan las relaciones entre el poliedro y su proyección general. Estas cuatro restricciones adicionales limitan la posible área de planos espaciales, limita los gradientes de caras ocultas, y restringe la colinealidad de ciertas configuraciones cara-arista. Wei probó que estas cuatro restricciones son condiciones necesarias y suficientes para que un dibujo sea realizable, y dio reglas para la reconstrucción recursiva de la escena tridimensional. Su algoritmo sólo admitía objetos poliédricos con cuatro grados de libertad como máximo. Lo que hacía en otros casos, era separar el dibujo y tratar el dibujo original como compuesto de varias partes, es decir, un poliedro se separa en varios pequeños poliedros que no superen los 4 grados de libertad. Aunque esto limita severamente la práctica de este procedimiento, Wei dio una solución general al problema, y también discutió los errores en el posicionamiento de vértices y cómo hacer trabajar sus reglas en esa situación.

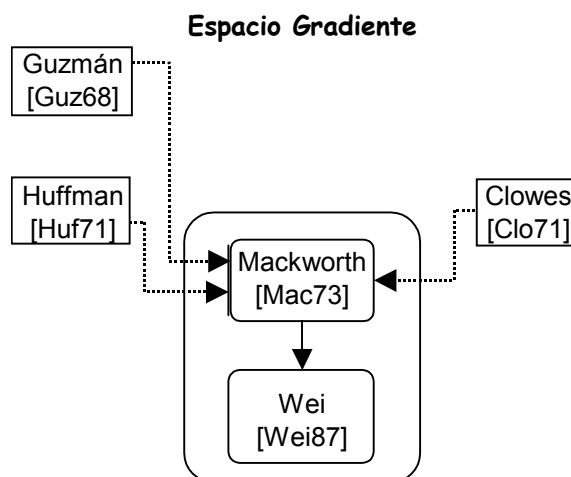


Figura 2-10. Gráfico resumen del método basado en el espacio del gradiente.

Estos dos autores son los únicos que se han clasificado dentro del enfoque del gradiente. Mackworth comenzó el camino y Wei logró llegar a obtener una condición necesaria y suficiente para que el dibujo lineal sea reconstruido. Éste es un enfoque totalmente geométrico, que para nada tiene en cuenta la forma de reconstruir que poseen los humanos, aunque al igual que el enfoque del etiquetado puede ser una herramienta geométrica útil en la Reconstrucción de Modelos 3D.

### 2.3.3. Enfoque basado en la programación lineal.

Sugihara [Sug82] presentó un enfoque diferente para determinar la reconstructibilidad de los dibujos lineales, y consiguió presentar una condición necesaria y suficiente para que un dibujo lineal represente un poliedro en términos de programación lineal. Así es capaz de discriminar entre dibujos lineales correctos e incorrectos.

De las entidades geométricas de la imagen (vértices, aristas, ...) se formulan las condiciones geométricas que las ligan. El conjunto de condiciones da lugar a un sistema de ecuaciones e inecuaciones que se resuelve mediante programación lineal. La resolución del mismo ofrece el modelo tridimensional buscado. Que este conjunto de ecuaciones e inecuaciones se pueda resolver es una condición suficiente para que el objeto sea *reconstruible*.

En una primera fase, el sistema de Sugihara utiliza un algoritmo de etiquetado parecido al utilizado por Huffman y Clowes cuando el dibujo no posee líneas ocultas representadas [Sug82], y el algoritmo de [Sug78] cuando sí que hay líneas ocultas dibujadas. A partir del etiquetado automáticamente se formulan un conjunto de ecuaciones e inecuaciones, de forma que si el problema de programación lineal planteado se puede resolver, es condición necesaria y suficiente para que el dibujo lineal pueda ser reconstruido. De esta forma, se define un problema de programación lineal a través del cual se discriminan los posibles dibujos candidatos incorrectos.

El algoritmo [Sug86] se presenta desde un punto de vista rigurosamente matemático. Es por ello, por lo que pequeños errores en la digitalización provocan un error en la posición de un vértice, que puede hacer que el sistema no acepte un dibujo como reconstruible, cuando para la percepción humana esos errores serían considerados despreciables en la interpretación. Un ejemplo de este tipo de objetos podría ser el que se muestra en la Figura 2-11.

La precisión matemática que se aplica hasta este momento hace que existan ecuaciones redundantes en el sistema, y Sugihara presenta un teorema para detectar las ecuaciones en exceso y eliminarlas. Además, mediante relajación de las coordenadas de los vértices del dibujo, el sistema puede incluso determinar su posición correcta y corregir el dibujo para mejorar su interpretación. De este paso se puede incluso determinar, no simplemente si es un dibujo incorrecto, sino también si un dibujo es corregible y así reconstruible, o si por el contrario se trata de un objeto imposible.

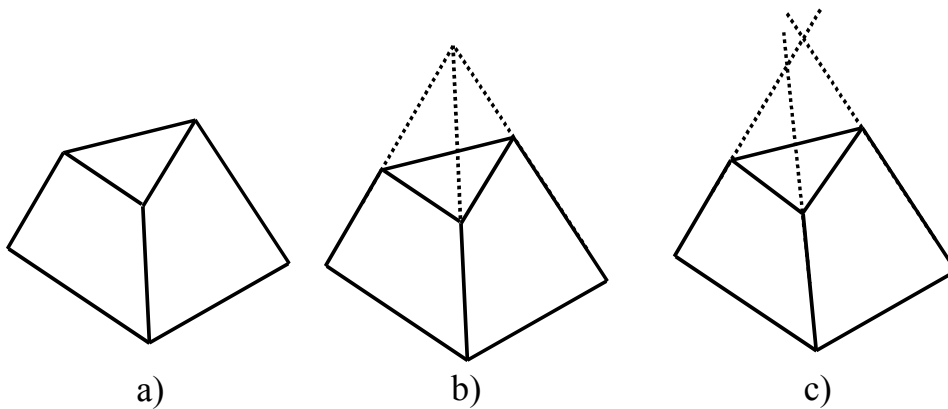


Figura 2-11. a) Proyección que cualquier ser humano reconocería como un tronco de pirámide de base triangular. b) Figura matemáticamente reconstruible. c) figura matemáticamente imposible de reconstruir

El sistema de Sugihara queda limitado a objetos poliédricos, pero tiene como principal característica, con respecto a los autores anteriores a él, que no requiere que la entrada sea perfecta, admitiendo líneas perdidas, líneas superfluas o vértices posicionados incorrectamente.

Shimshoni y Ponce en [SP94] y posteriormente en su tesis [Shi95] continúan los pasos de Sugihara. Shimshoni quiere reconstruir un modelo tridimensional partiendo de una fotografía, y para ello combina el método de Sugihara con las técnicas que utilizan las restricciones en el espacio del gradiente [Kan80], [Mac73], con las técnicas que detectan formas a partir de sus sombras [NIK95], [FZ91] y con los métodos de radiosidad [MRC86].

Shimshoni obtiene un sistema lineal a partir de las restricciones impuestas por las líneas en el espacio del gradiente, y manteniendo un sistema de ecuaciones e inecuaciones parecido

## Percepción Artificial de Dibujos Lineales

---

al que Sugihara proponía. Pero mientras Sugihara resuelve el sistema superrestringido quitando algunas de las restricciones, Shimshoni las mantiene todas e introduce un factor de incertidumbre en ellas. Con ello convierte el problema en un problema de optimización.

Para evitar caer en mínimos locales durante el proceso de optimización, lo que hace es dar un punto de partida tan cercano al óptimo como sea posible. Para ello en un primer paso se utiliza un modelo simplificado del cual se obtienen varios puntos de partida. Como segundo paso se toma la mejor solución de las anteriores como punto de partida para comenzar el proceso de optimización completo.

Grimstead y Martin [GM95] construyen un modelo sólido B-Rep partiendo de una vista única y sin la representación de las aristas ocultas de objetos poliédricos triédricos, sin agujeros o partes conectadas que puedan ser consideradas como objetos independientes. Los pasos que sigue para llevar a cabo su método son:

- ❑ Introducen los datos desde una tableta digitalizadora, por lo que el primer paso debe ser limpiar el boceto 2D y así quitar los posibles errores en la digitalización.
- ❑ Posteriormente utiliza el etiquetado de líneas de Huffman y Clowes para así identificar la topología inicial del objeto y localizar donde se encuentran las caras ocultas.
- ❑ A partir del etiquetado de líneas se obtienen las restricciones y características del dibujo (como por ejemplo las posibles simetrías) involucrando a los coeficientes de caras desconocidas y la profundidad de los puntos. Incluye ecuaciones que computan el paralelismo de líneas y la simetría oblicua.
- ❑ Se encuentra una solución al sistema lineal por medio de mínimos cuadrados y las ecuaciones que son extremadamente incompatibles son rechazadas.
- ❑ Al igual que Sugihara, se da cuenta de que su sistema es sensible a las imprecisiones, y por ello recalcula los vértices como las intersecciones de caras y así asegura que se trata de un sólido reconstruible.
- ❑ Cualquiera de las caras incompletas son entonces completadas con la información que posee el sistema de sus caras vecinas, produciendo así un modelo sólido. Si hay múltiples soluciones es el usuario quien elige la correcta.

Siguiendo con su investigación en [GM96] describen y evalúan una modificación a los métodos de etiquetado para conseguir que éstos puedan trabajar de modo incremental.

Quieren conseguir un programa que a la vez que se va dibujando el boceto, vayan asignándose las etiquetas a las líneas y así ahorrar tiempo y hacer más eficiente la aplicación. Las restricciones en cuanto a clase de posible objeto a reconstruir en este sistema siguen siendo las mismas que en el artículo del 95.

Grimstead y Martin a pesar de seguir el enfoque que comenzó Sugihara, y de ofrecer un método primordialmente geométrico, comienza a observar –tal vez inspirados por el artículo de Leclerc- que existen ciertas “cualidades” que se pueden detectar en el dibujo lineal, como son la simetría, el paralelismo de líneas y las uniones en ángulo recto. Ciertamente, aquí se está considerando de nuevo la percepción humana, y se están aceptando ciertas premisas que aceptaría un diseñador. De la misma forma Varley y Martin, cuyo método se pasa a describir a continuación, aceptan la palabra “regularidades” o “artefactos” como una propiedad que se observa en el grafo y que en el modelo a reconstruir se presentará como tal.

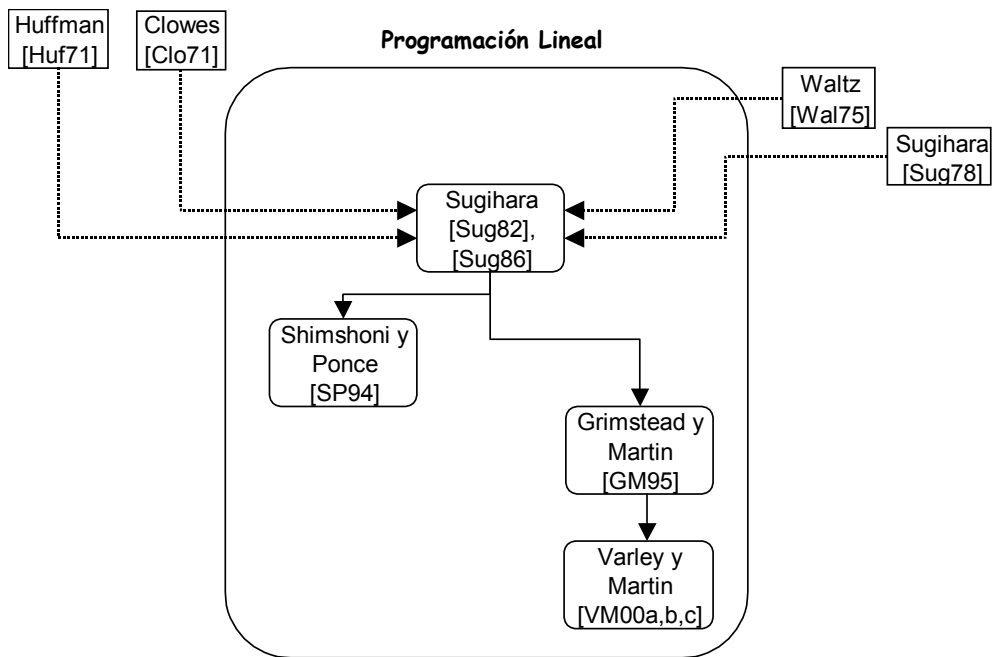


Figura 2-12. Gráfico resumen del método basado en la programación lineal.

El enfoque dado por Grimstead y Martin es seguido directamente por Varley y Martin [VM00a], [VM00b], [VM00c] quienes siguen asumiendo que el objeto que se pretende reconstruir se trata de un volumen cerrado, que todos los vértices del objeto son triédricos, y que no hay agujeros cruzando la pieza. Su aplicación llamada RIBALD sigue fundamentalmente tres pasos:

- El primero es un procesado preliminar [VM00a], que incluye resolver la geometría frontal y la categorización del boceto. En este paso intenta deducir y obtener toda la información posible, sin hacer ningún tipo de cambio en el boceto: las líneas y uniones son etiquetadas, se identifican pares de líneas candidatas a ser paralelas y con todo ello se obtiene la *geometría frontal* (geometría de la parte vista del objeto representado en el boceto). También se identifican potenciales simetrías locales y se intenta catalogar el objeto en un tipo de los que tiene definido, para así simplificar el proceso de reconstrucción.
- La reconstrucción topológica [VM00b]. Este proceso consiste en obtener la topología completa del objeto. Partiendo de toda la información recogida en el paso anterior se realizan diferentes hipótesis sobre la topología de las partes ocultas. Aquí comienza un proceso iterativo que en cada paso elige la mejor hipótesis añadiendo la parte oculta correspondiente hasta completar todas las partes ocultas del objeto.
- El refinado geométrico [VM00c]. Tiene como objetivo producir la mejor interpretación geométrica de la topología utilizando las simetrías y regularidades. La identificación de elementos simétricos en el objeto o regularidades produce una o más restricciones, las cuales limitan el posicionamiento y orientación de las caras. El sistema lineal intenta satisfacer el número máximo de restricciones, y una vez resuelto, las aristas y vértices son recalculados a partir de las caras que los contienen.

### 2.3.4. Enfoque Progresivo

Lamb y Bandopahay [LB90] intentando evitar un método numérico por la susceptibilidad que estos tienen a los errores en la entrada de datos, intentan aplicar algunas reglas de percepción, -por lo que algunos le han llegado a llamar *enfoque perceptual*-. Sin embargo, sus antecedentes declarados son los métodos de Sugihara, Waltz y Huffman.



En el estudio de Lamb y Bandopadhyay la entrada al sistema se realiza mediante tableta digitalizadora, y en un primer paso lo que se hace es procesar las coordenadas de las uniones del dibujo, así como las pendientes y longitudes de las líneas respecto a un sistema de ejes principales asociado al dibujo. Posteriormente se procede a un etiquetado de cada uno de los vértices y líneas de grafo, siguiendo el sistema de etiquetado de Waltz. El método de etiquetado se aplica fundamentalmente para rechazar antes de comenzar el cálculo a aquellos dibujos que representan objetos imposibles y para identificar si las caras son paralelas a las direcciones principales u oblicuas. En función de esta clasificación se le asignan coordenadas a todas las uniones de la región. De modo que se asignan coordenadas a los vértices del siguiente modo:

- ❑ Si la región es paralela a un plano principal, entonces es fácil asignar las coordenadas de sus vértices siendo su distancia al plano principal la longitud de las aristas perpendiculares al mismo.
- ❑ Si la región es oblicua la información de los planos paralelos a los planos principales ya evaluados dan la información necesaria para calcular la ecuación de la pendiente y con ella el plano.
- ❑ Si la región no es paralela a ningún plano conocido, se busca si es posible aplicar alguna regla de simetría para completar las coordenadas de ese plano.
- ❑ Si ninguno de estos supuestos es cierto, el algoritmo le pide al usuario que especifique las coordenadas.

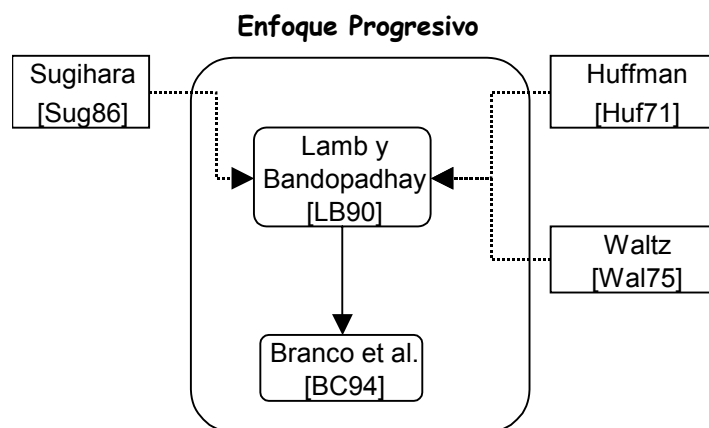


Figura 2-13. Gráfico resumen del enfoque progresivo.

## Percepción Artificial de Dibujos Lineales

y el proceso se repite hasta que a todas las uniones visibles se les ha asignado una coordenada.

En resumen este sistema reconstruye poliedros partiendo del boceto realizado en sistema axonométrico que puede ser tanto isométrico como oblicuo, y para ello utiliza reglas heurísticas y algunas perceptuales para intentar ajustar automáticamente el dibujo y reconstruir su estructura 3D. Cuando estas reglas fallan depende de la interacción del usuario retomar el proceso.

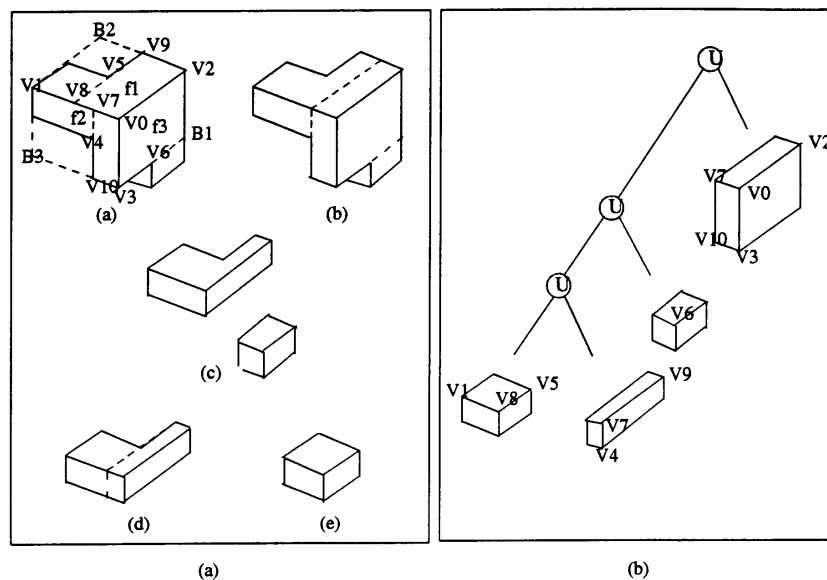


Figura 2-14. Reconstrucción mediante identificación de primitivas (a) Los estados durante el proceso; (b) el árbol CSG final

Branco y otros [BCN94] crean un sistema llamado IDeS, y sigue una línea muy parecida a la de [LB90], pero con la particularidad que combina técnicas de reconstrucción 3D con operadores CSG convencionales. IDeS es un sistema totalmente interactivo. Su aspecto es muy parecido a un programa de dibujo, sobre el que se puede dibujar ayudado por un menú sobre el que hay que pinchar para elegir las herramientas. Acepta tanto dibujos en perspectiva como en sistema axonométrico, y siempre sin aristas ocultas. El usuario dibuja el boceto y es también él quien informa al programa si el dibujo 2D representa a un modelo 3D. El método utilizado para la reconstrucción se divide en dos pasos fundamentalmente:

primero se calculan las caras que son totalmente o parcialmente vistas, y luego las ocultas, y el método básicamente es el mismo que el de Lamb y Bandopadhay.

Algunos autores han llamado *interactivo* a este enfoque, por el objetivo principal que declaraban Lamb y Bandopadhay: construir una herramienta interactiva con la que los diseñadores, y usuarios de sistemas CAD/CAM pudieran utilizar de un modo sencillo, práctico y cuyo resultado, si no era el deseado, fuera fácilmente corregible. Esto es cierto, pero no es este el objetivo exclusivo de este enfoque, puesto que hay autores clasificados en otros enfoques que han conseguido obtener una herramienta considerablemente más interactiva que estos. Además, en este trabajo, el grado de interactividad no se ha considerado un criterio relevante para clasificar los diferentes métodos.

### 2.3.5. Enfoque de la Identificación de Primitivas

Wang y Grinstein [WG89] presentan un algoritmo para extraer *bloques* como primitivas (poliedros rectangulares) automáticamente desde un dibujo. Es decir, trabaja sólo con poliedros normales. Examinan el etiquetado de los dibujos lineales y extraen información de los bloques de ciertos tipos de uniones siguiendo los esquemas de etiquetado de Huffman y Clowes. La orientación de las uniones de líneas únicamente determina la orientación del bloque en el espacio y las líneas etiquetadas de la unión determinan si el bloque es añadido o substraído. El primer bloque identificado es asignado una localización arbitraria en el espacio, y los siguientes bloques son posicionados relativamente a los anteriores de acuerdo con su relación de adyacencia. Se mantienen dos listas de estructuras de datos: una contiene todos los bloques que tienen su posicionamiento en el espacio resueltos, y la otra los que tienen los bloques sin resolver. Si la segunda lista no está vacía cuando el dibujo ha sido "atravesado", se sabe que hay más de un objeto. El mismo proceso puede ser otra vez aplicado para resolver el siguiente objeto.

Wang, en su tesis doctoral [Wan92], extendió este método para dibujos utilizando como primitiva objetos tetraédricos para encontrar una representación CSG para un dibujo lineal de un poliedro general.

Así, la identificación de primitivas reconstruye la escena reconociendo ejemplos o ejemplos

### Identificación de Primitivas

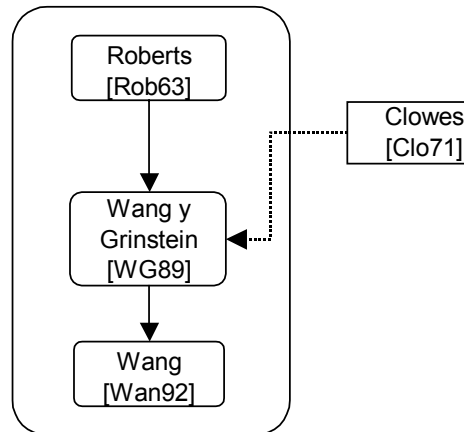


Figura 2-15. Gráfico resumen del método de identificación de primitivas.

parciales de primitivas conocidas, como es un cubo, un bloque, un cilindro, etc. Este enfoque está limitado por la naturaleza de los objetos con los que puede tratar, puesto que asume que el objeto tridimensional representado está completamente compuesto por primitivas conocidas. Una segunda limitación es que asume que el dibujo es totalmente perfecto. Sin embargo tiene el beneficio que la estructura tridimensional final es un modelo de geometría constructiva de sólidos (CSG).

### 2.3.6. Enfoque de las Regularidades

Las *Regularidades* son definidas como un conjunto de propiedades de un modelo las cuales son visibles o detectables en la imagen 2D. Otra definición podría ser: relaciones geométricas espaciales entre entidades individuales o en grupos de entidades, que aparecen en el dibujo 2D de forma no accidental y que corresponden a regularidades geométricas reales, las cuales existen también en el objeto 3D. Se trata de una forma de hacer explícita la percepción humana. A pesar de que el término regularidades ya se había utilizado en otros enfoques con anterioridad, es aquí donde ganan una importancia suprema.

Marill [Mar91] presenta un enfoque basado en una simple y sencilla observación; y centra su estudio en la siguiente consideración:

*La interpretación humana de un dibujo lineal tiende a ser la más sencilla posible.*

Marill aplica esta regla de simplicidad a los ángulos que forman las aristas entre sí en sus uniones que deberán ser lo más sencillos posibles: todos iguales para un mismo objeto. Esto es lo que denominó *Mínima Desviación Estándar de Ángulos (MSDA)* y era la única propiedad que su programa (denominado Construct) le exigía al modelo tridimensional: conservar la igualdad de cada uno de los ángulos entre las aristas medidos dos a dos. Los poliedros regulares y convexos tienen tendencia a aceptar esta suposición, y es fácil adivinar que esta estrategia sería apropiada únicamente para algunos pocos poliedros regulares o para estructuras alámbricas equiangulares.

A pesar de la sencillez de la propuesta, y de lo limitado de su aplicación, es un paso muy interesante puesto que ofrece un nuevo enfoque más cercano a la naturaleza del sistema de percepción humana y es el primero que habla del método de *inflado*.

En efecto, Marill introduce los métodos de reconstrucción mediante inflado que consisten en mantener las coordenadas  $(X, Y)$  de todos los vértices del modelo iguales a las coordenadas del correspondiente nodo de la imagen 2D. Mientras, las coordenadas  $Z$  de los vértices del modelo van variando y constituyen las variables del proceso de optimización. Las coordenadas  $Z$  serán las que le darán la profundidad al modelo una vez reconstruido.

Así se establece un problema de optimización donde el objetivo es minimizar la función  $F(Z)$ , donde  $Z = (z_0, z_1, \dots, z_{n-1})$  es el conjunto de coordenadas  $Z$ , y constituye un vector de variables independiente, y  $n$  es el número de vértices, y es el que determina el orden del problema. La función objetivo ( $F$ ) contiene una formulación numérica de la regularidad MSDA que toma un valor distinto de cero cuando el modelo tridimensional no determina los mismos ángulos para todas las parejas de aristas concurrentes en un mismo vértice, y se anula para aquellos modelos tridimensionales en los que se los ángulos formados por todas las parejas de aristas concurrentes en un mismo vértice son iguales.

Leclerc y Fischler [LF 92] mejoraron la función objetivo de Marill añadiendo una nueva regularidad: la Planicidad de Caras. Encontraron que era obvio que la planicidad de caras fuera una propiedad que se cumple en la figura 2D. Sin embargo, esta regularidad podía cumplirse trivialmente, y como solución construyó una función objetivo combinación lineal de funciones ponderadas por un parámetro variable  $\lambda$  para escapar del mínimo trivial:

## Percepción Artificial de Dibujos Lineales

---

$$F = \lambda F_1^2 + (1-\lambda) F_0$$

*Ecuación 2-1*

Donde  $F_1$  es el MSDA de Marill, y  $F_0$  es su regularidad de planicidad de caras.

Siguiendo los pasos de Marill y también los de Leclerc, Brown y Wang [BW96] proponen la MSDSM: Minimum Standard Deviation of Segment Magnitudes, como sustitución del MSDA. En su artículo critican y comparan su idea con el artículo de Marill, y admiran su capacidad para reconstruir sin necesidad de la utilización de modelos. Se trata un artículo de mérito bastante limitado, puesto que sólo es útil para un número muy limitado de objetos.

Lipson y Shpitalni [LS96] mejoraron el enfoque de las regularidades y fueron quienes sentaron definitivamente las bases y alcanzaron resultados significativos. En [LS95] presentan su intérprete de bocetos y en [LS96] formulan matemáticamente las siguientes regularidades: la Mínima Desviación Estándar de ángulos, el Paralelismo de Líneas, la Planicidad de Caras, Isometría, Ortogonalidad de Esquinas, Ortogonalidad Oblicua de Caras, Simetría Oblicua de Caras, Ortogonalidad de Líneas, Perpendicularidad de Caras, Caras Prismáticas, Colinealidad de Líneas y la planicidad de Chaflanes Oblicuos. La función objetivo la expresan como suma de las contribuciones de cada una de las regularidades y esta función llegará a ser cero cuando todas las regularidades sean cumplidas por completo. Matemáticamente puede ser formulada como:

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

*Ecuación 2-2*

Donde:

$\alpha_j$  es el peso ponderado dado a la regularidad número j.

$R_j(z)$  es la j-ésima regularidad.

Este enfoque es seguido por Company y otros [CGC98], [CGC99] que se centran en obtener un modelo tridimensional tentativo lo más cercano posible al objeto final que se quiere obtener: una estimación preliminar de la forma del cuerpo representado en el boceto a mano alzada inicial. Pero para ello primero necesitan identificar la tipología del objeto, ya que se observa que todos los algoritmos estudiados dependen principalmente del tipo o naturaleza del objeto, y en función de esta tipología elegir un método de reconstrucción

distinto. Para ello [CCG99] construyen una clasificación de poliedros basado en reglas fundamentalmente obtenidas de la percepción [Hof00], [Pal99] con el propósito de que éstas controlen automáticamente el proceso y que se aplique el criterio de selección de la reconstrucción más apropiado. Una vez se ha obtenido el modelo tentativo ya se aplica el Algoritmo de Optimización. Aunque en algunos casos se obtiene ya con el modelo tentativo el modelo final 3D que se perseguía. En [CCC00] se muestran los diferentes algoritmos que se han utilizado.

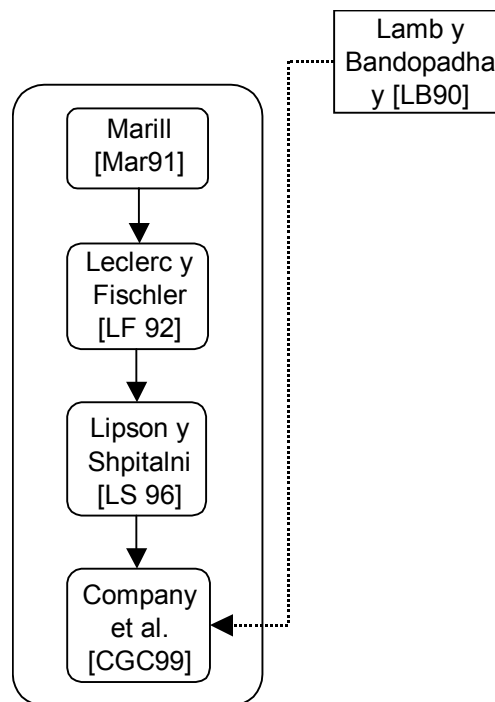


Figura 2-16. Gráfico resumen del método de las regularidades.

## 2.4. RECONSTRUCCIÓN TRIDIMENSIONAL DE MODELOS DESDE MÚLTIPLES VISTAS

La reconstrucción tridimensional desde vistas múltiples no ocupa el objetivo fundamental de esta tesis, pero debido a la obvia relación que ha existido a lo largo del tiempo entre ambos problemas, se ha decidido incluir un breve estado del arte de la misma.

## Percepción Artificial de Dibujos Lineales

---

Un estado del arte bastante detallado sobre la reconstrucción tridimensional de sólidos desde múltiples vistas es suministrado por Nagendra y Gujar [NG88] y posteriormente por Wang y Grinstein [WG93]. En estos dos artículos la clasificación se realiza a partir del tipo de modelo obtenido: B-Rep o CSG. En este trabajo se va a seguir la misma línea en la clasificación, aunque se destacarán otros elementos diferenciadores a lo largo de este apartado, teniendo como resumen la Tabla 2-3 al final de esta sección.

Los modelos CSG utilizan primitivas sólidas, transformaciones de cuerpo rígido y operaciones booleanas, creando un árbol del modelo final. Mientras que los modelos B-Rep o alámbricos son modelos delimitados por superficies. Cada superficie límite está a su vez limitada por aristas, mientras que cada arista está limitada por dos vértices. Se trata de dos tipos de modelos utilizados frecuentemente en los sistemas CAD/CAM.

Los métodos para ambos tipos de modelos se basan fundamentalmente en el emparejamiento de vértices o de contornos.

En dibujos multivista ortogonales, cada vista o proyección suministra para cada uno de los vértices del objeto dos de las tres coordenadas que determinan al vértice en un espacio tridimensional. Por ejemplo si se hace coincidir el plano XY con el plano horizontal y el plano XZ con el frontal, y se tiene la proyección de la planta se están suministrando las coordenadas  $(x, y)$ , y mediante el alzado se proporcionarían las coordenadas  $(x, z)$ . De esta forma, con dos vistas ortogonales se obtendrían las tres coordenadas de un espacio 3D -con una coordenada por duplicado puesto que se da en dos de estas vistas-.

El principal problema se encuentra en encontrar las diferentes proyecciones de un mismo vértice. Para este emparejamiento se utiliza la coordenada que se proporciona por duplicado. Pero en la mayoría de los casos se puede encontrar más de un vértice que tenga el mismo valor en una coordenada, y de esta forma se pueden llegar a dar varias posibles soluciones del emparejamiento. El número de permutaciones es proporcional al número de pares de vértices compartiendo una coordenada en común. Si esto fuera así, la solución recaería sobre un conjunto finito de posibles objetos. Algunos de esos objetos pueden ser rechazados por violar algunos requerimientos topológicos, mientras que otros permanecen como posibles soluciones al emparejamiento.

Es decir, en líneas generales la base del problema es la resolución del emparejamiento de vértices y contornos.



La reconstrucción partiendo de múltiples vistas es un campo bastante cercano a los métodos de reconstrucción de escenas desde múltiples imágenes (por ejemplo desde imágenes estereográficas).

En la Figura 2-20 se muestra un gráfico cronológico de todos los autores que han tratado hasta el momento el problema de la reconstrucción de modelos 3D partiendo de vistas múltiples. En el gráfico se observa la clasificación general en función del tipo de modelo que se obtenga de la reconstrucción. También se ven las principales relaciones que han tenido unos autores sobre otros. Fundamentalmente con las flechas se quiere reflejar aquellos algoritmos que son seguidores de los anteriores.

### 2.4.1. Representación Alámbrica / B-Rep

Uno de los primeros trabajos sobre vistas múltiples fue realizado por Idesawa [Ide73], quien trató tres vistas de los dibujos de ingeniería sobre una base rigurosamente matemática. Idesawa muestra que su teoría es aplicable tanto a objetos con superficies planas como superficies curvas, pero en su ejemplo práctico sólo muestra un ejemplo con multitud de caras planas, pero ninguna curva. Las tres vistas que necesita como datos de partida deben tener todas las aristas visibles, sin hacer distinción entre aristas vistas y ocultas, como si se tratase de objetos con caras transparentes.

Los pasos fundamentales de su método son:

- ❑ Generar vértices 3-D candidatos desde puntos 2-D.
- ❑ Generar aristas 3-D candidatas desde vértices 3-D.
- ❑ Generar caras 3-D desde arista 3-D estableciendo circuitos.
- ❑ Combinar las caras para obtener el objeto final 3-D.

Este enfoque se basa fundamentalmente en la resolución de problema de emparejamiento de uniones y líneas y la principal diferencia entre los autores que resuelven el problema utilizando el enfoque B-Rep está en las técnicas utilizadas para lo que Idesawa llamó “*eliminación de figuras fantasmas*” y el tipo de superficies contenidas en los objetos que son capaces de reconstruir. Por esta razón he decidido ir enumerando la amplia variedad bibliográfica recogida en orden cronológica, indicando su mayor aportación.

Wesley y Markowsky en “Fleshing out projections” [WM81] amplían el algoritmo que ya presentaron en “Fleshing out Wireframes” [WM80] el cual tenía como objetivo encontrar

## Percepción Artificial de Dibujos Lineales

---

todos los posibles sólidos poliédricos que pudieran corresponder al conjunto de proyecciones de un sistema alámbrico que se dan como entrada. En [WM81] amplían este algoritmo para que sea más general e incluso utilizan la información de identificar aristas vistas y aristas ocultas. Hacen una clasificación de los vértices en Clase I y Clase II, y con ellos consiguen realizar unos pseudo-modelos alámbricos, y a partir de estos se buscan las caras candidas o caras virtuales. Un pseudo-modelo alámbrico no es más que un modelo alámbrico que posee elementos del modelo que se quiere obtener, pero con un subconjunto de elementos clasificados como inciertos. Si la proyección puede ser etiquetada, se aplicará el algoritmo Wire Frame, y en el caso de que la proyección no sea etiquetable es cuando se utilizará este nuevo algoritmo de Projections.

Sus dibujos de entrada admiten tanto líneas sólidas como discontinuas y las vistas pueden ser completas o vistas de detalle. Sólo admiten objetos poliédricos cuyas proyecciones contienen sólo líneas rectas, aunque los autores afirman que es extensible a objetos con superficies curvas debido al componente lógico de este algoritmo que es topológico por naturaleza.

Sus artículos con un enfoque altamente matemático, muestran la naturaleza de puzzle que tiene el problema. Se obtienen múltiples soluciones, y como es natural cuanto mayor es el número de proyecciones que se presentan como entrada, menor es el número de soluciones, puesto que hay menor ambigüedad.

Haralick y Queeny [HQ82] enfocaron el problema como un problema de etiquetado, de identificación de nodos y líneas, y su método es ampliado por Lysak y otros [LDK95] a curvas.

Sakurai [Sak83] extiende el algoritmo de Wesley y Markowsky, incluyendo como entrada líneas rectas y arcos circulares que no tienen porqué tener una orientación predeterminada o restringida, ya sean líneas continuas como discontinuas, reconstruyendo modelos sólidos que pueden tener superficies cilíndricas y cónicas, toroidales y esféricas. El modelo sólido solución puede ser múltiple en el caso de que las proyecciones de partida sean ambiguas.

Preiss [Pre84] trata el problema como un problema de propagación de restricciones con el siguiente enfoque: primero un conjunto de vértices candidatos se emparejan utilizando las restricciones aplicadas sólo a los vértices y posteriormente se aplican restricciones a las aristas que quedaron desemparejadas. Las aristas sin sentido se borran del conjunto candidato. Al final las aristas son relacionadas en circuitos utilizando restricciones

aplicables a las aristas en el circuito y restricciones aplicables entre circuitos. En su artículo muestra los resultados de aplicar el método a cuerpos con caras planas y cilíndricas.

Gu y otros [GTS 86] extienden el algoritmo de Sakurai para tratar con un número más amplio de objetos y también extraen ideas sobre reconocimiento de patrones de Aldefeld [Ald83]. Su algoritmo encuentra todas las posibles soluciones y ofrece buenos resultados para poliedros y cilindros, en los que las condiciones se relajan considerablemente:

- ❑ Los ejes de cualquier cilindro ya no necesitan ser paralelos a uno de los ejes coordenados, es suficiente con que sean paralelos a un plano coordenado. Y el plano que corta al cilindro sólo necesita ser paralelo o perpendicular a un plano coordenado.
- ❑ Se permiten arcos elípticos, hipérbolas, y curvas regulares de mayor orden en la proyección si sus ejes de simetría son paralelos a un eje coordenado. De esta forma también se extienden los tipos de intersecciones entre objetos.

Gujar y Nagendra, [GN89], construyen un método que reconstruye poliedros desde tres de sus vistas ofreciendo múltiples soluciones. Aprovechan la información que les proporcionan las líneas continuas y discontinuas que acepta en la entrada al sistema, y que reconocen como aristas vistas y ocultas. Hacen una clasificación de subobjetos y elaboran dos exámenes para seleccionar los ciertos de los inciertos, con lo cual ganan en tiempo de resolución.

En el artículo de Chen y otros [CPC92] lo que muestran es un conjunto de reglas que facilitan la reconstrucción, cuya misión principal es determinar la veracidad o falsedad de caras 3D y la relación de coexistencia o no-coexistencia entre caras. Se preocupan de dar métodos que agilicen los métodos de búsqueda de caras 3D, y así reducir el espacio de búsqueda del problema. Tratan sólo con objetos con caras planas, aunque dicen extenderlo a objetos con partes cilíndricas.

Yan y otros [YCT94] presentan un método que sigue los pasos descritos al principio de esta sección pero además aprovechan la información que reciben de las aristas discontinuas, representando aristas ocultas como en el caso de [GN89]. Esta información la utilizan para eliminar aristas redundantes, y para solucionar casos patológicos. Como solución ofrecen todos los posibles objetos que las proyecciones ortogonales pueden representar y necesitan tres vistas como entrada a su sistema. Este trabajo es continuado posteriormente por Sastry y otros [SSM99].

## Percepción Artificial de Dibujos Lineales

---

También Ah-Soon y Tombre siguen en [AT95] los pasos de “Fleshing out projections”, y presentan un prototipo de un sistema que combina la reconstrucción geométrica con el reconocimiento simbólico y el emparejamiento. Sus datos de entrada son tres vistas perfectas y las caras de los objetos a representar pueden ser tanto planas como curvas, pero sólo acepta cilindros con el eje paralelo a uno de los planos de proyección. Añade simbología a la geometría, aunque se trata de una innovación muy incipiente.

You y Yang [YY96] siguen la metodología de Yan [YCT94] pero tratan con objetos poliédricos cerrados y también con superficies cuádricas siempre que éstas tengan su eje paralelo a uno de los planos coordenados.

Masuda y Numa [MN97] propusieron un método eficiente para convertir proyecciones ortogonales en modelos sólidos basados en topología de volúmenes abiertos (non-manifold) y un método para subsanar algunos errores de proyecciones ortográficas incorrectas. Utilizan un modelo de descomposición celular entre un modelo alámbrico y un modelo sólido final, el cual simplifica los últimos pasos de la reconstrucción 3D, pero la generación del modelo alámbrico continua utilizando el método de Idesawa y Wesley. Así su proceso consta en primer lugar de la obtención de un modelo alámbrico y a partir de él un modelo B-Rep; a partir de éste y mediante identificación de circuitos se obtienen las células y con ellas un modelo celular. La combinación de células determina la forma del sólido. Acepta como entrada líneas rectas, circulares y elipses, puesto que es capaz de resolver superficies planas, cilíndricas, cónicas, esféricas y toroidales, eso sí, siempre que tengan sus ejes de revolución paralelos a un eje de principal.

El trabajo de Masuda ha sido comercializado por IBM y muestra de ello se puede obtener en [Mas]

Kuo [Kuo98a], [Kuo98b] necesita tres vistas ortográficas como entrada al sistema. Reconstruye el modelo alámbrico, y luego todas las caras candidatas las encuentra empleando el método de búsqueda que llama del *mínimo-ángulo-interno*. Una vez las caras han sido detectadas en cada una de las vistas, se detectan las superficies redundantes y se borran utilizando el método que llama de Decisión en Cadena. Este método es en esencia una búsqueda combinatoria que repite operaciones utilizando algunas reglas de decisión cuyo objetivo es la reducción del espacio de búsqueda. Al final todas las caras son ensambladas, y se buscan todas las posibles soluciones. El método es totalmente automático y las superficies del modelo pueden ser tanto planas como cuádricas.

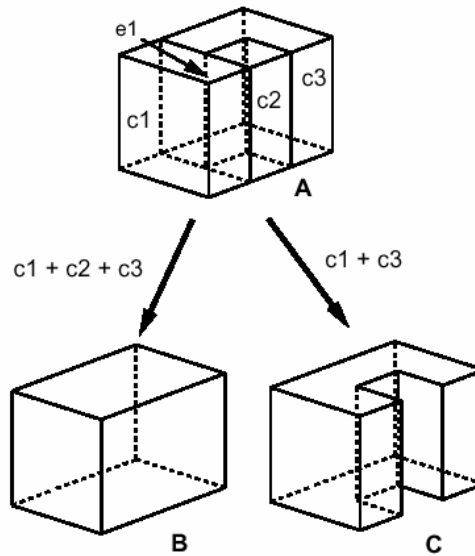


Figura 2-17. Modelo celular y modelos sólidos posibles (extraído del artículo de Masuda).

Por su parte Shin y Shin [SS98] aplican un algoritmo siguiendo los pasos de Wesley y de Yan en el caso de poliedros y para objetos curvos se basa en Sakurai. Lo mejora utilizando propiedades geométricas y la topología de las primitivas geométricas. Con la utilización de relaciones topológicas entre primitivas 2D y 3D gana eficiencia. Es capaz de reconstruir superficies planas y curvas muy limitadas: cilindros y toros pero siempre que tengan sus ejes de revolución paralelos a los ejes principales.

Tanaka y otros [TIH98] proponen un sistema capaz de descomponer automáticamente un dibujo 2D de un conjunto ensamblado dado por sus proyecciones ortogonales, en sus componentes 3D mediante un sistema de ecuaciones de elementos sólidos (parecidos a los bloques virtuales que Wesley y Markowsky propusieron). Cada elemento sólido se clasifica en: elemento verdadero de un componente del conjunto, y elemento falso o incierto. En principio se piensa en un método que combina todos los posibles elementos sólidos y luego se proyectan cada uno de ellos en las vistas 2D para comprobar si la solución coincide con el dibujo de conjunto, pero por supuesto, esta combinación es altamente ineficiente. Para evitar esto, y agilizar el proceso los elementos se organizan en conjunto de ecuaciones de elementos sólidos y de esta forma se minimiza el número de elementos inciertos. Esto consiste fundamentalmente en imponer cuatro condiciones que se deben cumplir, y las combinaciones se van realizando mientras alguna de estas condiciones no se cumpla, o

## Percepción Artificial de Dibujos Lineales

---

quede algún elemento incierto. El sistema busca todas las soluciones posibles y trabaja también con superficies curvas.

Para la reconstrucción de los componentes, Tanaka sigue los métodos utilizados por Wesley y Markowsky para superficies planas y por Sakurai para las superficies curvas.

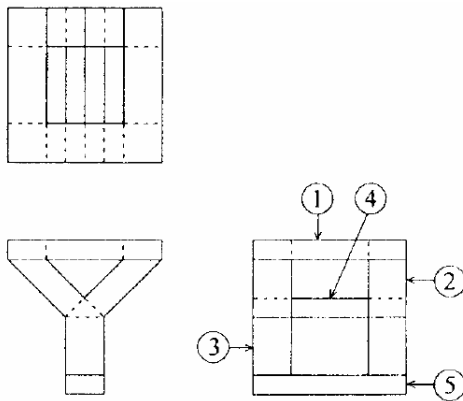


Figura 2-18. Datos de partida de un ejemplo de Tanaka

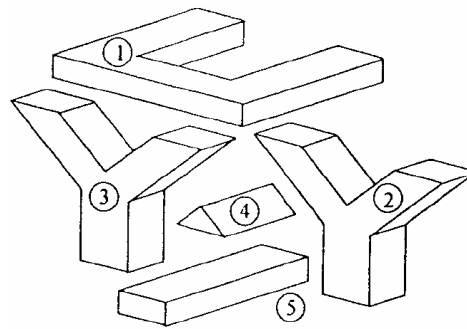


Figura 2-19. Una de las cinco soluciones que el algoritmo de Tanaka ofrece

Suh y otros [SKK99] proponen un nuevo algoritmo de dos-fases. Suh reconstruye todo tipo de objetos poliédricos con caras planas partiendo únicamente de dos vistas 2D ortográficas. Intenta reducir el espacio de búsqueda que se utiliza en estos procesos debido al tiempo computacional que esto supone. Quiere obtener todos los posibles objetos que las dos vistas pueden generar, pero también quiere que el programa sea capaz de elegir la más plausible: la que el ser humano elegiría. Esto lo realiza en dos fases: una primera fase reconstruye los objetos parcialmente a partir de las caras candidatas representadas sobre cada área del dibujo lineal. En la segunda fase reconstruye el objeto totalmente a partir de los objetos parcialmente reconstruidos en la fase anterior, añadiendo otras caras candidatas, validándolas geoméricamente. El algoritmo lleva consigo una búsqueda combinatoria basada en reglas de decisión de las caras y de reglas heurísticas. Las reglas de decisión comprueban la validez geométrica de las caras y las reglas heurísticas ayudan a aumentar la velocidad de búsqueda.

## 2.4.2. CSG - Geometría Constructiva de Sólidos

Aldefeld [Ald83] vio que una parte compleja podía verse como compuesta de varios objetos elementales (Primitivas CSG). Es por ello que utilizó atributos y relaciones (árbol CSG) de objetos elementales en el proceso de reconstrucción. Así propuso un esquema de patrones donde las formas 2-D son contrastadas con hipotéticos objetos 3-D hasta que se encuentra una coincidencia importante entre los dos. Su filosofía es pensar que un objeto complejo está formado por objetos elementales que pertenecen a un conjunto de clases predefinidas, y reconocer estos objetos elementales haciendo uso de los conocimientos sobre su patrón de clase dependiente de sus representaciones 2D. Su algoritmo clasifica los objetos en diferentes clases y utiliza modelos para guiar el proceso de interpretación.

Siempre necesita tres proyecciones como entrada, y su limitación principal es que sólo es válido para objetos extruidos de espesor constantes. Trabaja con elemento curvos, aunque siempre sus bases son paralelas a uno de los planos de proyección y cada objeto elemental debe representarse como si fuera un cuerpo aislado.

Aldefeld y Richter [AR84] muestran un algoritmo guiado por el usuario, y lo compara con el sistema de Aldefeld [Ald83] totalmente automático. El sistema totalmente automático está basado en el reconocimiento de patrones y en la búsqueda heurística, y aunque no necesita para nada la intervención del usuario está limitado en el tipo de modelos capaz de reconstruir. Mientras, el método asistido acepta objetos más generales y mucho más complejos, pero la solución dependerá en gran medida de los conocimientos del usuario.

Chen y Perng [CP88] propusieron un proceso de reconstrucción compuesto de tres fases: descomposición, reconstrucción y composición. Primeramente descompone las vistas en varios tipos predefinidos de subvistas, luego reconstruye cada sub-parte mediante una operación de translación y una operación de recorte de plano. Finalmente, la parte es compuesta mediante la utilización de relaciones de volumen desde su correspondiente árbol CSG. Este método interactivo con el usuario puede tratar con objetos poliédricos que no son de espesor constante.

Meeran [MP93] presenta en su artículo aspectos geométricos y lógicos del reconocimiento de formas desde dibujos lineales bidimensionales, que son implementados en su programa Prolog, el cual es capaz de reconocer protrusiones y depresiones.

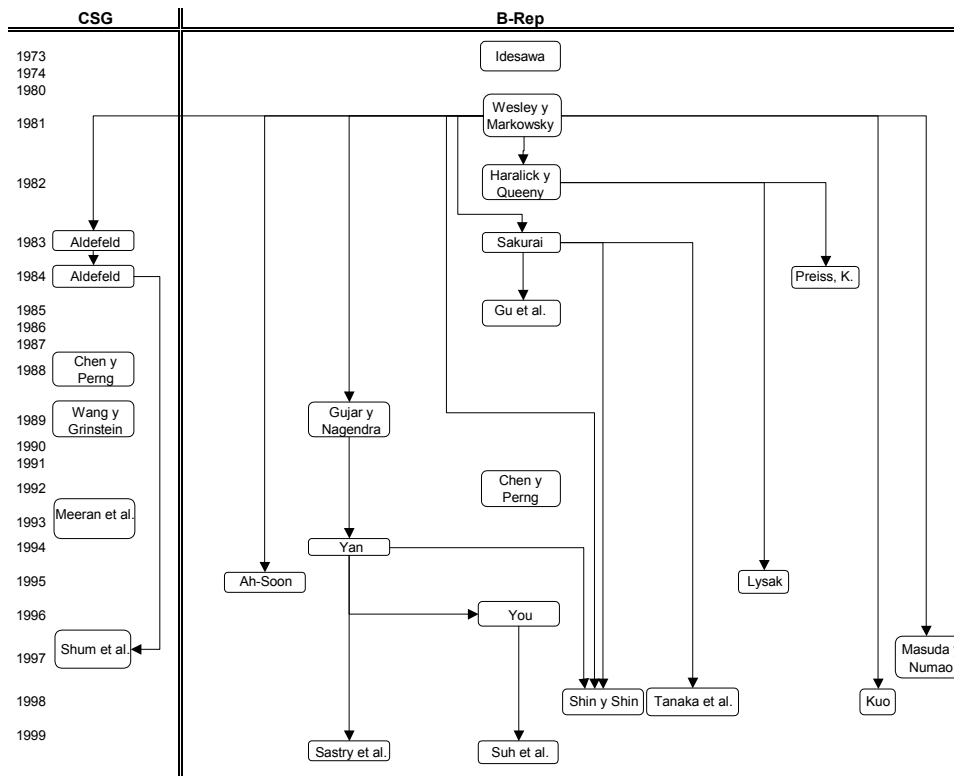


Figura 2-20. Resumen cronológico de los autores de algoritmos de reconstrucción partiendo de vistas múltiples.

En los artículos de Shum y otros [SLY97] y [SLY01] además de comentar algunos de los artículos anteriormente citados, también presenta su método dentro del CSG. Su método necesita de las seis vistas del sistema diédrico multivista sin aristas ocultas, que represente a un poliedro de caras planas. También admite cilindros rectos de revolución siempre y cuando su eje sea perpendicular a uno de los planos principales de proyección, es decir, admite como entrada sólo a rectas y circunferencias. Tampoco admite agujeros ciegos, o pasantes que cambien de tamaño, ni tampoco otras superficies de revolución como son los conos o superficies toroidales. Su método agrupa las seis vistas en tres parejas, y por cada pareja de vistas, una de ellas – llamada la vista- $g$  – es extruida gradualmente de acuerdo con la información que suministra su vista pareja – llamada vista- $d$  -. Durante este proceso se generan primitivas sólidas extruidas. Finalmente todos los sólidos extruidos se intersectan para ofrecer un único objeto solución.



## **2.5. RESUMEN DE LA CLASIFICACIÓN**

---

De la literatura recogida sobre Reconstrucción de modelos 3D se ha realizado un resumen en forma de dos tablas, diferenciando en cada una los algoritmos cuyo punto de partida es una única imagen (Tabla 2-2) o múltiples vistas (Tabla 2-3). En ellas se pueden observar las tendencias a través del tiempo de los algoritmos desarrollados, además de mostrar un resumen claro y rápido de las principales características de cada algoritmo.

A continuación se estudian por separado los criterios de clasificación de los métodos de reconstrucción tridimensional que se han incluido en las tablas resúmenes.

### **2.5.1. Tipos de superficies**

El tipo de superficies que es capaz de reconstruir un sistema es fundamental para comprobar la versatilidad de una aplicación. Pero la fiabilidad y eficacia en la resolución de un tipo particular de superficies también es un aspecto importante.

En el resumen se ha distinguido entre algoritmos que aceptan en sus soluciones modelos sólidos sólo formados por superficies planas, frente a los que aceptan superficies curvas. Es decir, que se distingue entre reconstrucción de objetos poliédricos y el resto.

Claro está que es mucho más difícil realizar la reconstrucción de superficies curvas, y por eso mismo la mayoría de autores que lo han intentado se han limitado a reconstruir los tipos más sencillos de superficies curvas: figuras de revolución o extruidas con su eje perpendicular a uno de los planos de proyección, cuádricas y poco más. Es decir, han acabado reconstruyendo casos particulares de superficies curvas.

### **2.5.2. Interacción**

La interacción se refiere principalmente a quien toma las decisiones críticas. Si el enfoque que se le está dando al problema es obtener una aplicación automática, o se pretende construir una herramienta amigable para que el diseñador construya el modelo con ayuda del ordenador.

### **2.5.3. Representación de líneas ocultas en 2D**

La diferencia fundamental respecto a la consideración de las aristas ocultas estriba en que los algoritmos que aceptan aristas ocultas como datos de partida –ya sean identificados como líneas ocultas o simplemente como líneas- poseen la topología completa. Mientras, aquellos autores que consideran la proyección de un objeto opaco, en el que no se muestran las líneas ocultas, tienen que abordar: el problema de determinar la topología trasera (la topología de la parte no vista del objeto).

### **2.5.4. Datos de entrada**

Ya se ha comentado que la entrada al sistema, el dibujo lineal 2D, puede realizarse por medio de un dibujo en papel que es posteriormente escaneado, también directamente sobre una tableta digitalizadora, o cualquier otro mecanismo que permita dibujar en dos dimensiones directamente sobre la aplicación. Pero la diferenciación fundamental que aquí se ha considerado no ha sido el periférico ni el carácter inmediato o diferido de la entrada, sino que ha sido si este dibujo de entrada es: un dibujo perfecto (proyección perfecta de un objeto), un dibujo con pequeñas imperfecciones (sus líneas no se unen perfectamente en los vértices, o no respetan las reglas de la proyección), o un boceto realizado a mano alzada. La diferencia fundamental así se encuentra en la tolerancia que tenga el programa para aceptar y procesar las imperfecciones que contenga el dibujo de entrada, así como la necesidad de un preprocesado más o menos importante de la figura de entrada antes de comenzar la reconstrucción.

Año	Autores	Enfoque					Tipo de Superficies		Interacción		Tipo de Representación		¿Representa las Líneas ocultas (2D)?		Dibujo de Entrada		
		Etiquetado	Espacio del gradiente	Programación lineal	Progresivo	Identificación de primitivas	Regularidades	Planas	Curvas	SI	No	B-Rep	CSG	SI	No	Perfecto	Pequeñas impresiones
1963	Roberts	x				x		x		x				x		x	
1968	Guzmán	x						x		x				x			
1971	Huffman	x						x		x				x			
	Clowes	x						x		x				x			
1973	Mackworth							x		x				x			
1975	Waltz	x								x							
1978	Sugihara	x															
1980	Kanade	x															
1982	Sugihara			x						x							
1986	Sugihara			x						x							
1987	Malik	x								x							
	Wei		x							x							
1989	Wang y Grinstein									x							
1990	Lamb y Bandopahay				x					x							x
1991	Marill							x									
	Wang							x									
1992	Leclercy y Fischler							x									
	Wang y Grinstein							x									
1993	Marti et al.	x						x									
	Branco et al.							x									
1994	Shimshoni y Ponce							x									
1995	Grimstead y Martin							x									
	Grimstead y Martin							x									
1996	Lipson y Shpitalni							x									
	Parodi							x									
1999	Company et al.							x									
	Varley y Martin							x									
2000	Brown y Wang							x									
	Varley y Martin	x						x									

Tabla 2-2. Métodos de reconstrucción partiendo de una única vista.

## Percepción Artificial de Dibujos Lineales

Año	Autores	Enfoque		Tipo de Superficies		Interacción		Representa las Líneas ocultas (2D)		Datos de Entrada		
		B-Rep	CSG	Planas	Curvas	Sí	No	Sí	No	Dibujo perfecto	Pequeñas imprecisiones	Boceto a mano alzada
1973	Idesawa	x		x			x	x		x		
1981	Wesley y Markowsky	x		x			x	x		x		
1982	Haralick y Queeny	x		x			x					
1983	Sakurai	x		x	x	x		x				
	Aldefeld		x	x	x		x	x		x		
1984	Preiss, K.	x		x	x		x	x		x		
	Aldefeld y Richter		x	x	x	x		x		x		
1986	Gu et al.	x		x	x		x	x		x		
1988	Chen y Perng		x	x	x	x						
1989	Gujar y Nagendra	x		x			x	x				
1992	Chen et al.	x		x	x		x					
1993	Meeran y Pratt		x	x	x		x	x		x		
1994	Yan y otros	x		x			x	x		x		
1995	Ah-Soon y Tombre	x		x	x			x		x		
	Lysak et al.	x		x	x		x					x
1996	You y Yang	x		x	x			x			x	
1997	Masuda y Numao	x		x	x			x			x	
	Shum et al.		x	x	x	x		x		x		
1998	Kuo	x		x	x		x	x			x	
	Shin y Shin	x		x	x			x		x		
	Tanaka et al.	x		x	x		x	x			x	
1999	Suh et al.	x		x				x		x		
	Sastry	x		x		x		x		x		
2001	Shum et al.		x	x	x	x		x		x		

Tabla 2-3. Métodos de reconstrucción partiendo de vistas múltiples

## 2.6. CONCLUSIONES

La búsqueda selectiva ha permitido obtener la inmensa mayoría de los trabajos más relevantes en la Reconstrucción tridimensional a partir de dibujos lineales. Se ha contrastado los resultados de la búsqueda con las referencias cruzadas entre los diferentes trabajos, llegando a la conclusión de que se dispone de una gran mayoría de los trabajos publicados.

En el ámbito de los algoritmos de reconstrucción partiendo de vistas múltiples, han ido avanzando los que generan modelos CSG y los B-Rep. Eso sí, hay que destacar que la gran mayoría de los autores se han centrado en la obtención de modelos B-Rep.

Debido a que este tipo de reconstrucción parte del sistema multivista, que es un método construido por el hombre, el proceso inverso no tiene porqué ser un proceso natural. Es por ello que los métodos aquí son fundamentalmente geométricos. No podemos hacer una distinción clara entre enfoques geométricos y perceptuales en la reconstrucción de modelos partiendo de una única vista.

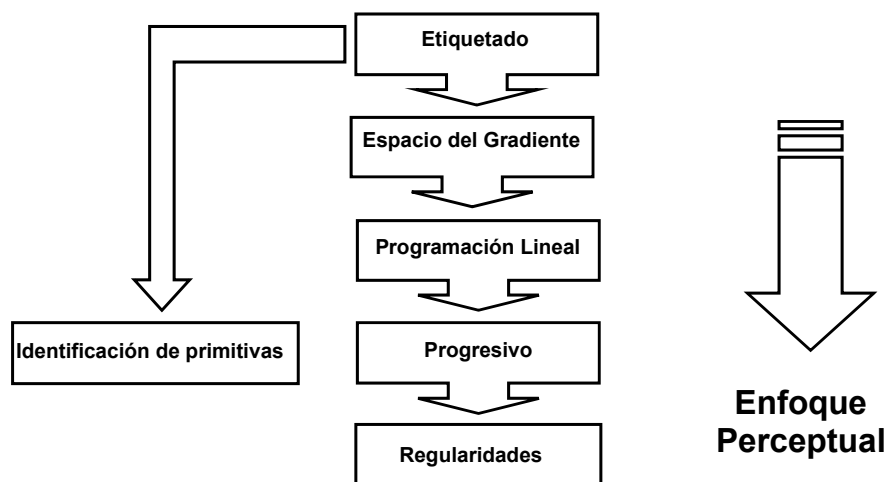


Figura 2-21. Evolución de los métodos que parten de una Única Vista.

En los métodos que toman como entrada una única vista, durante los años setenta, y siguiendo la inspiración de Guzmán y con Huffman y Clowes como precursores, el enfoque predominante fue el del etiquetado de líneas. Este enfoque fue seguido por numerosos autores con la intención de conseguir ampliar el alcance del método a un mayor número de tipos de objetos.

Este enfoque va evolucionando y de él se desprende el enfoque del espacio del gradiente y de aquí el de la programación lineal. Estos enfoque continúan utilizando el etiquetado de líneas, pero ya no como la parte fundamental del método. Pero continúan siendo algoritmos puramente geométricos, con una base matemática muy fuerte. El enfoque de la Programación Lineal que tuvo gran fuerza durante los años ochenta y a Sugihara como fundador, comienza a dejar entrever la importancia que la percepción posee en el objetivo de la Reconstrucción 3D. Esta idea queda implícita en los trabajos de Sugihara, aunque no la desarrolla de forma explícita.

## Percepción Artificial de Dibujos Lineales

---

La idea de que no hay que intentar encontrar información únicamente en el grafo, sino que también hay que aportar conocimientos psicológicos de los seres humanos va siendo cada vez más aceptada por la comunidad de estudiosos de la Reconstrucción de modelos 3D en los años noventa. Es por ello que por la misma época aparecen dos nuevos enfoques: el enfoque progresivo y el enfoque de las regularidades. Estos dos enfoques se irán desarrollando posteriormente, y mostrarán la importancia en aumento de las reglas de la percepción visual humana y de su formulación.

Por supuesto, estos enfoques perceptuales no son puros; combinan la información geométrica que se pueda obtener del grafo de partida con los conocimientos psicológicos, pero son los psicológicos los que predominan.

La idea ataca de raíz al problema de reconstrucción: por una parte la geometría por si sola, es incapaz de resolver el problema (porque no hay correspondencia biunívoca entre imágenes 2D y modelos 3D), y por otra parte, la percepción visual está poco estudiada y es poco susceptible de formulaciones algorítmicas. Por ello, todos los autores se han apoyado en ambos enfoques (geométrico y perceptual) para elaborar sus métodos. A pesar de que la preponderancia de uno de los dos enfoques no siempre ha sido explícita, ésta ha condicionado las características de todos los métodos conocidos. La falta de una utilización clara y eficiente de aportaciones provenientes de uno de los dos enfoques ha limitado las prestaciones de muchos de los métodos propuestos.

La idea de los métodos perceptuales es la que escoge la autora de esta tesis, aunque sin dejar de lado las características geométricas de los modelos y su formulación. Desde mi punto de vista, tan importante es la geometría como la percepción. En concreto el enfoque de las regularidades es el elegido, no sólo porque en la percepción queda todo un mundo que explorar, sino porque, además, hay muchas regularidades ya identificadas pero que necesitan de una formulación más rigurosa para que el resultado de su detección sea eficiente.

# **Capítulo 3.**

# **La Reconstrucción 3D y las Regularidades**

---

## **3.1. INTRODUCCIÓN**

---

En el capítulo de introducción se ha hecho una descomposición muy simple del proceso de reconstrucción tridimensional en tres grandes bloques: datos de partida, reconstrucción geométrica 2D y reconstrucción geométrica 3D. En dicho capítulo también se han hecho algunos comentarios genéricos sobre el estado del arte en cada una de estas partes del proceso. Además, en el capítulo de antecedentes se han comentado las aportaciones de quienes han trabajado sobre algunos de los aspectos de la reconstrucción tridimensional.

Pese a ello, entendemos que es preciso repasar con mayor detalle aspectos como la reconstrucción 2D, aunque no estén directamente relacionados con las aportaciones principales de la tesis. El motivo es que la reconstrucción 2D de la aplicación REFER ha sido mejorada durante el proceso de implementación y validación experimental de las aportaciones de la tesis. Por ello, explicar las aportaciones propias y su proceso de validación, requiere explicar con un cierto detalle todos aquellos aspectos de la reconstrucción en los que se ha tenido que hacer aportaciones complementarias.

Por otra parte, en los capítulos precedentes también se ha indicado que el objetivo general de esta tesis supone una aportación a la fase de reconstrucción 3D mediante una técnica concreta: la reconstrucción mediante optimización matemática basada en regularidades. Se trata de una técnica desarrollada por otros autores, sobre la que la tesis hace una

aportación original. Sin embargo, también creemos oportuna una explicación más o menos detallada de toda la técnica, con el objeto de incidir en otras pequeñas aportaciones originales que hemos realizado en la misma.

En definitiva, las aportaciones marginales o complementarias al proceso de reconstrucción, realizadas durante la puesta a punto de la herramienta informática empleada para validar esta tesis, requieren una contextualización en el proceso global de reconstrucción tridimensional. En este capítulo se realiza la mencionada contextualización de las aportaciones de la tesis en el ámbito de la reconstrucción geométrica, al tiempo que se describen las aportaciones complementarias realizadas en el método de reconstrucción geométrica empleado.

### 3.2. RECONSTRUCCIÓN 2D

---

Es muy difícil conseguir formas 3D válidas partiendo de un boceto tosco. No obstante, la reconstrucción tridimensional basada en optimización puede ofrecer buenos resultados siempre que el dibujo de partida se aproxime a una proyección paralela de una forma real. Sin embargo, si el dibujo de partida es un boceto tosco, con coordenadas  $(x, y)$  imprecisas, el inflado ofrecerá formas distorsionadas o incluso modelos virtuales imposibles de construir en la realidad. Es muy difícil evitar este problema añadiendo más y más condiciones a la función objetivo. La estrategia elegida por el grupo REGEO para resolver este problema ha sido la reconstrucción 2D *previa*, que produce como salida un dibujo lineal geoméricamente corregido.

Las interfaces caligráficas normalmente están configuradas para convertir dinámicamente (*mientras* el usuario dibuja) bocetos de líneas en segmentos rectos, unir segmentos que terminan cerca unos de otros, asegurar el paralelismo, etc. Es lo que se ha venido a llamar Reconstrucción 2D on-line. Pero puesto que este trabajo se centra en la reconstrucción de dibujos lineales que pueden provenir tanto de un boceto a mano alzada posteriormente vectorizado, como de un boceto introducido mediante una interfaz caligráfica, nos centraremos en la reconstrucción 2D que se realiza *después* de la introducción de los datos del boceto. De este modo, se presentará una reconstrucción 2D off-line, que arregla los bocetos después de acabados.

Así, en la aplicación REFER se ha implementado una reconstrucción 2D off-line basada en una técnica muy innovadora. Se trata de un algoritmo que post-procesa los bocetos mediante una técnica de optimización. En efecto, primero hace ciertas tareas típicas de la



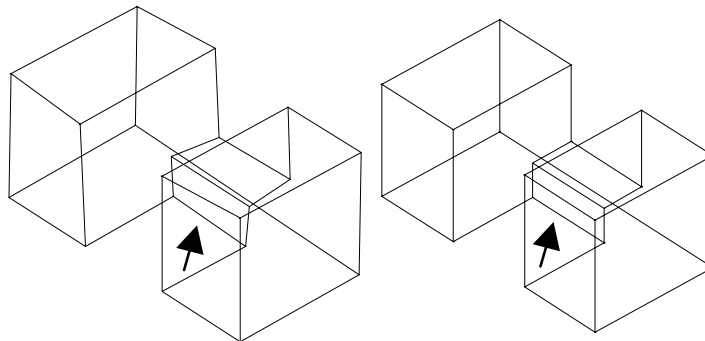
reconstrucción 2D: las líneas de longitud nula son eliminadas y los vértices perdidos se conectan. Pero, a continuación realiza un proceso de optimización para mejorar el paralelismo y la colinealidad principalmente. Se trata de un proceso de optimización donde las variables son las coordenadas  $x$  e  $y$  de todos los vértices, y el paralelismo y la colinealidad de líneas configuran la función objetivo.

Así, los dibujos 2D arreglados se obtienen mediante optimización de una función objetivo definida como la suma ponderada de señales perceptivas. Las variables de optimización son las coordenadas de todos los vértices  $X = (x_1, x_2, \dots, x_n)$  e  $Y = (y_1, y_2, \dots, y_n)$ .

$$F(X, Y) = \sum \alpha_j R_j(X, Y)$$

*Ecuación 3-1*

La formulación detallada de las dos regularidades 2D consideradas en la versión actual de REFER se puede encontrar en §3.5, donde se hace una descripción más detallada de todas las regularidades y de su formulación.



*Figura 3-1. Boceto inicial y dibujo lineal de salida de una forma poliédrica.*

Los resultados de este método de reconstrucción 2D son buenos, en cuanto a calidad del modelo final obtenido, pero requieren unos tiempos de cálculo (de segundos en los dibujos simples, y minutos en dibujos complejos) que lo hacen claramente impracticable hasta que se acelere la resolución del proceso de optimización. En la Figura 3-1 se muestra un boceto típico y su dibujo lineal corregido (se señala la convergencia inicial de algunas líneas finalmente paralelas).

## Percepción Artificial de Dibujos Lineales

---

Las limitaciones “teóricas” del enfoque aparecen debido a que la colinealidad y el paralelismo son señales perceptivas, es decir, son propiedades probables del modelo final. Puntos de vista accidentales pueden causar detecciones erróneas de falsos paralelismos si dos aristas no paralelas son accidentalmente proyectadas (Figura 3-2) como líneas paralelas en el dibujo.

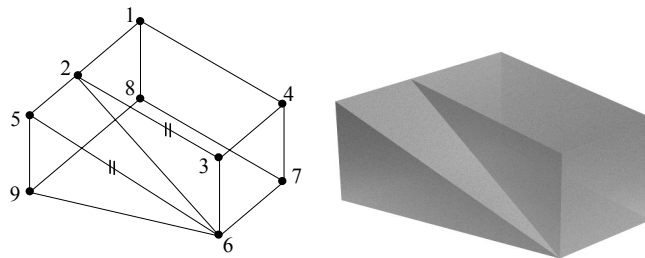


Figura 3-2. a) Dibujo lineal con un falso paralelismo de líneas, b) la solución perceptualmente mejor.

## 3.3. LA OPTIMIZACIÓN

---

La técnica de reconstrucción mediante optimización de una función objetivo basada en regularidades tiene algunos inconvenientes que le son propios. Pero también comparte problemas genéricos de cualquier técnica de optimización matemática. A continuación, vamos a intentar poner de manifiesto algunos de estos problemas, y las soluciones que hemos adoptado durante el desarrollo de la tesis.

### 3.3.1. El Inflado y la Función Objetivo

El sistema de referencia principal utilizado en reconstrucción tridimensional mediante optimización está orientado de modo que los ejes de coordenadas  $X$  e  $Y$  son coplanarios con el plano del dibujo inicial, y el eje de coordenadas  $Z$  sale perpendicularmente al plano del dibujo formando así un sistema ortogonal de coordenadas, generalmente orientado a derechas. A este sistema de referencia se le llama *Sistema de Coordenadas de Inflado*.

Con el sistema de coordenadas así definido, la reconstrucción consiste en encontrar un conjunto de coordenadas  $Z$  de los vértices del dibujo que determine el modelo tridimensional deseado. Pero hay un número infinito de objetos tridimensionales cuya proyección ortogonal coincide con el dibujo lineal dado, y es lo que llamamos *Extensión*

*Ortografía* [Mar91]. Aunque sólo unos pocos o sólo uno de estos modelos encaja con los requerimientos perceptuales, de modo que muchos de los modelos en la extensión ortográfica no son aceptables, como son los modelos virtuales que contienen caras retorcidas no planas (ver Figura 3-3).

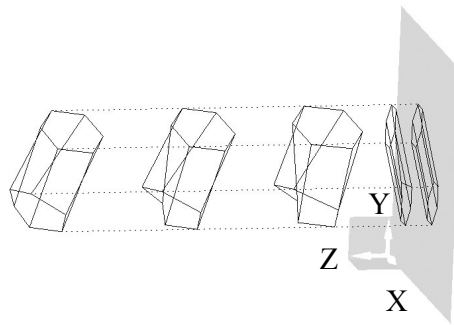


Figura 3-3. Proceso de inflado. Algunos modelos de la extensión ortográfica intermedios, junto con el correcto.

El modelo psicológicamente aceptable se obtiene optimizando una función objetivo, definida como una suma ponderada de señales perceptivas: como recoge la Ecuación 3-2.

$$F(Z) = \sum \alpha_j R_j(Z)$$

Ecuación 3-2

Donde  $\alpha_j$  es el coeficiente de ponderación  $j$ -ésimo, y  $R_j(Z)$  es la regularidad  $j$ -ésima. De forma que la formulación matemática de las regularidades debe expresarse en función de las variables independientes  $Z$ . Deben ser formuladas de modo que sean iguales a cero cuando se consiga cumplir la condición completamente, y muy diferente de cero cuando no se cumple. Es decir, se computan penalizaciones de aquellos rasgos perceptuales de la figura que no se cumplen en el modelo. De las regularidades y su formulación se hablará de forma más detallada en §3.5.

El problema de reconstrucción tridimensional, planteado como un problema de “inflado” de la figura plana para obtener un modelo que minimice la función objetivo de la Ecuación 3-2, es un buen método, porque permite obtener soluciones válidas en casi todos los casos. De hecho, sólo las figuras “patológicas” causan problemas. Pero la intención del diseñador es plantear figuras lo más claras posibles, por lo que las situaciones patológicas están fuera de nuestro objetivo y no constituyen ningún problema.

El principal problema que presenta el método está en que la optimización matemática de la Ecuación 3-2. no es simple, dado que la función a optimizar no es lineal, y presenta mínimos locales [Con01].

### 3.3.2. Técnicas de Optimización Matemática “local”

En un principio se utilizó el Hill-Climbing como método para minimizar matemáticamente la función objetivo debido a su facilidad de implementación y a su robustez [Mar91]. Además, su repetibilidad permite comparar resultados. Brevemente, para una variable de diseño de dimensión  $n$ ,  $Z = (z_1, z_2, \dots, z_n)$ , se elige un “paso”  $s_1$  arbitrario y se comprueban  $2n$  valores:

$$\begin{aligned} Z_1 &= (z_1 + s_1, z_2, \dots, z_n), \\ Z_2 &= (z_1 - s_1, z_2, \dots, z_n), \\ &\dots \\ Z_{2n-1} &= (z_1, z_2, \dots, z_n + s_1), \\ Z_{2n} &= (z_1, z_2, \dots, z_n - s_1) \end{aligned}$$

Ecuación 3-3

y se seleccionan como el nuevo  $Z$  al valor  $Z_i$  correspondiente a la función objetivo menor. El proceso se repite en un bucle hasta que la mejora de la función objetivo es menor que un valor límite impuesto, o cuando el número máximo de búsquedas (de iteraciones) se ha sobrepasado. Entonces se salta a un nuevo “escalón” eligiendo un nuevo paso  $s_2$ , menor que el anterior, y se repite el proceso. Normalmente, se realizan entre tres y diez escalones o bucles con valores decrecientes ( $s_1 > s_2 > \dots$ ).

Esta técnica es robusta, aunque no todo lo rápida que se desearía. Hasta el momento, no hemos considerado técnicas de optimización más potentes, basadas en determinación de gradientes, porque se entiende que calcular gradientes de una función objetivo es complejo y sólo es rentable si la función objetivo cuyos gradientes se determinan es una función eficiente y simple. Dado que en la sección §3.5 justificamos que las funciones actuales aún no son simples ni eficientes, parece prematuro emplear dichas técnicas.

### 3.3.3. Sintonizar y Normalizar la Optimización

La reconstrucción mediante técnicas simples como el Hill-Climbing es altamente sensible a los parámetros de optimización. Por ejemplo, se puede comprobar fácilmente que cuando un paso inicial  $s_1$  de una búsqueda unidireccional del algoritmo de optimización es demasiado grande, la solución no guarda la proporcionalidad con las asunciones psicológicas (Figura 3-4).

La diferencia es relevante desde el punto de vista de la percepción, porque los observadores humanos asumen que el objeto representado en el dibujo lineal debe tener dimensiones regulares en todas las direcciones [Pal99] [Hof00], puesto que han sido representados eligiendo un punto de vista que claramente remarca cualquier tipo de desproporción que el modelo pueda tener. Es indiscutible que las dimensiones “exactas” no se pueden inducir ni del boceto ni del dibujo lineal mejorado, pero el requisito perceptual de guardar las proporciones sí que es razonablemente exigible. El problema es fácil de resolver normalizando la magnitud del paso  $s_1$  como una proporción de las dimensiones máximas del dibujo de partida, es decir, del  $\max(\Delta x, \Delta y)$ . Esto hace la optimización totalmente independiente de las dimensiones del dibujo lineal de partida. Se sugiere el rango entre 10-20% porque es una forma indirecta de imponer la suposición psicológica de proporcionalidad.

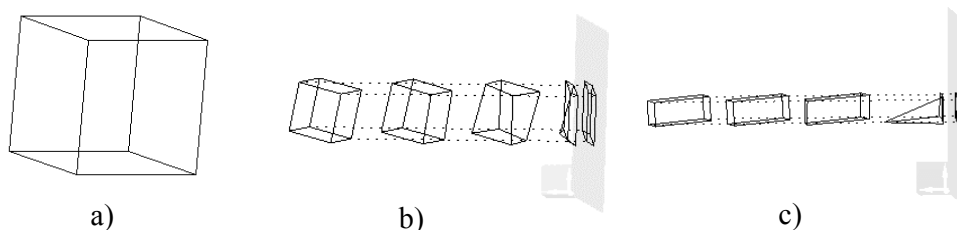


Figura 3-4. Grandes pasos en Hill-Climbing dejan modelos no proporcionales

- a) muestra el dibujo original,
- b) es el modelo psicológicamente correcto (obtenido con  $s_1 = 0.2 \max(\Delta x, \Delta y)$ ),
- c) modelo no proporcionado (obtenido con  $s_1 = 2.0 \max(\Delta x, \Delta y)$ ).

Con relación al número de escalones o bucles a utilizar, de cinco a diez es un buen rango para obtener buenas soluciones en una cantidad de tiempo razonable (aumentando la precisión con diez y disminuyendo el tiempo de cálculo con cinco). Respecto al decremento del paso de un bucle al siguiente, de nuestra experiencia se desprende que un decremento

de  $\Delta z$  constante, (como  $s_1= 0.125$ ,  $s_2= 0.0625$ ,  $s_3= 0.03125$ ,  $s_4= 0.015$ ,  $s_5= 0.007$ , propuesto por Leclerc y Fischler [LF92]) es conservador, puesto que el criterio de parada corta a aquellos escalones donde no se consiguen mejoras significativas. Por tanto, se considera que es mejor una reducción más progresiva. De todos modos, evitar excesivas diferencias en escalones consecutivos es una garantía contra fallos del algoritmo. De hecho, el mejor comportamiento se obtiene cuando el nuevo paso es el 30%-40% del paso actual. Además se ha llegado a la conclusión que cuando se emplean más de cinco pasos, los pasos intermedios no tienen un efecto cuantificable en el resultado, especialmente cuando el boceto inicial es altamente impreciso. Sin embargo, el valor adoptado por el  $\Delta z$  menor (el salto más pequeño) está claramente relacionado con la precisión final, y se recomienda que el  $\Delta z$  más pequeño debe estar hecho directamente proporcional a la desviación más grande que se acepte para cumplir las regularidades.

### 3.3.4. Técnicas de Optimización Matemática “global”

El problema de optimización planteado ofrece como inconveniente la obtención de soluciones que son mínimos locales de la función objetivo [Con01]. Para evitar esta limitación se pensó en utilizar métodos de optimización con comportamiento global.

Se puede destacar que Brown y Wang [Brow96] asumieron que “indudablemente los métodos estándar para minimizar las dificultades de los mínimos locales podrían ser empleados con tasas normales de éxito... (incluido pero no limitado a la familia de algoritmos de Simulated Annealing)”. Sin embargo, nuestra experiencia fue desanimadora, porque introdujimos un algoritmo Simulated Annealing que se suponía era capaz de encontrar un mínimo global y nos encontramos con que fallaba (tanto como lo hace un algoritmo de búsqueda local), cuando partimos de un modelo inicial malo, y cuando las regularidades utilizadas para generar la función objetivo son pobres o están mal definidas [CGC99], [Con01].

Para la resolución de estos problemas, además de la búsqueda e implementación de otros métodos de optimización que sean más adecuados al problema de reconstrucción, se han abordado varias estrategias complementarias. En primer lugar se intentó aplicar estrategias heurísticas para comenzar la búsqueda a partir de modelos tentativos bastante parecidos (topológicamente) al modelo buscado (más adelante en la sección §3.4 se verá de forma extendida estas estrategias). Otra estrategia fue aplicar la reconstrucción 2D. Y por otra

parte, se ha intentado mejorar la formulación de las regularidades propuestas por otros autores, así como se han detectado “falsas regularidades” y se han desarrollado nuevas y más potentes regularidades. En todas estas acciones he participado, pero a las que he dedicado más tiempo, y de hecho a continuación se hablará más extensamente hasta el final de este documento, han sido las que tratan con las regularidades y su formulación.

### 3.4. MODELOS TENTATIVOS

---

Desafortunadamente los dibujos lineales de entrada al sistema son una mala solución inicial para la optimización mediante técnicas locales, puesto que son mínimos locales, donde algunas regularidades se satisfacen trivialmente (como por ejemplo planaridad de líneas). Las estrategias desarrolladas [Con01] para evitar estos óptimos triviales están todas basadas en reglas heurísticas para tomar un modelo inicial *tentativo*, y todos ellos recaen en una de las dos categorías de inflado iterativo y generación directa.

Las técnicas de inflado iterativo están basadas en “Falsas Regularidades”. El principio MSDA (acrónimo en inglés de Mínima Desviación Estándar de Ángulos) definido por Marill [Mar91] es el primer método de inflado iterativo. No está ligado a ninguna señal perceptiva de la imagen. Se trata de una regla heurística basada en el hecho de que los poliedros regulares y convexos cumplen esta suposición, por lo tanto parece razonable pensar que este enfoque sólo es aceptable para poliedros convexos y regulares. Sin embargo, se usa con éxito en otros casos, porque muchos poliedros tienen tendencia a cumplir criterios de convexidad y regularidad de ángulos. Para un análisis más extensivo sobre las limitaciones del MSDA puede consultarse [BW96].

Leclerc y Fischler [LF92] presentaron una función objetivo que tiene en cuenta por un lado el inflado (con un sumando  $F_I$  formado por un conjunto de falsas regularidades), y por otro la optimización (con un sumando  $F_O$  que contiene a aquellas regularidades que se satisfacen de modo trivial en la solución inicial). En último lugar, en  $F_A$  se incluyen todas aquellas regularidades que siempre se pueden aplicar, por no ser satisfechas trivialmente en la solución inicial (la simetría es un buen ejemplo de esto).  $F_I$  y  $F_O$  se multiplicaban por un parámetro  $\lambda$  (con un valor inicial menor o igual a uno y decreciente a cero tal y como avanza la optimización) para escapar de la solución trivial utilizando criterios de inflado, mientras se garantiza que la solución final dependa sólo de regularidades verdaderas:

## Percepción Artificial de Dibujos Lineales

---

$$F(Z) = F_A(Z) + \lambda F_I(Z) + (1-\lambda) F_O(Z)$$

*Ecuación 3-4*

Este enfoque puede dar problemas, puesto que pueden producirse oscilaciones indeseadas, de forma que causen un mal funcionamiento de los algoritmos de optimización. Además, pequeños cambios en los parámetros de ajuste pueden hacer caer al inflado en algún mínimo local, del que luego es muy difícil escapar con el algoritmo de optimización. Es decir, introducir falsas regularidades genera una función objetivo mucho más compleja lo que conlleva más mínimos locales. Por tanto, el algoritmo puede escapar de una solución inicial ( $Z=0$ ), pero a costa de un alto riesgo de crear más soluciones falsas.

Hemos dicho arriba que el otro grupo de métodos de obtención de modelos tentativos son los denominados métodos directos, no iterativos. La estrategia más sencilla perteneciente a esta categoría de definición de un proceso directo para obtener un modelo tentativo 3D es el inflado aleatorio, que fue sugerido (aunque no implementado) por Brown y Wang [BW96]. El método aleatorio asigna a cada vértice un valor aleatorio de  $z_i$  en un intervalo normalizado  $\max(\Delta x, \Delta y)$ . Sorprendentemente, la evaluación realizada muestra que para algunas formas no demasiado complejas, la optimización desde un modelo obtenido mediante este método ofrece soluciones topológicamente válidas y con un ratio de fallo pequeño (uno de cada quince intentos). Además, en caso de fallo, es decir si el valor final de la función objetivo difiere claramente de cero, los siguientes modelos aleatorios darán un resultado topológicamente correcto, puesto que la probabilidad de que cuatro fallos consecutivos se sucedan es tan baja que nunca nos ha ocurrido durante las pruebas. Concluyendo, creemos que este enfoque es una alternativa en aquellos casos donde otros enfoques de modelos tentativos fallan.

La “reconstrucción preliminar” propuesta por Lipson y Sphitalni [LS95] también pertenece a la categoría de definición de un proceso determinista para obtener un modelo 3D tentativo. Sus antecedentes más claros fueron los estudios de Kanatani [Kan86] y Lamb y Bandopadhyay [LB90]. En este método, en un primer lugar se obtienen los ángulos dominantes de la distribución angular de las líneas del grafo, y con ello se calculan las direcciones principales del modelo. Cuando se obtienen tres ángulos dominantes se considera que corresponden con las tres direcciones principales. Luego se asigna una coordenada  $z_i=0$  a un nodo arbitrario. El proceso avanza asignando coordenadas  $Z$  a aquellos conectados a aquellos ya conocidos, cuyas coordenadas se calculan determinando las longitudes y las orientaciones de las aristas que unen el nodo conocido y



los que se están calculando. Si la forma tiene una distribución angular con exactamente tres direcciones principales el cálculo exacto de coordenadas puede extenderse a todos los nodos, dando lugar a un modelo previo bueno. Nosotros hemos adoptado este “inflado axonométrico”, y lo hemos mejorado, desarrollando un algoritmo que contempla más casos particulares y ampliando la tipología de los objetos con los que se trata [CCG99] y [CCP03].

## 3.5. REGULARIDADES

---

La forma de reconstruir que se ha considerado más oportuna en esta tesis es la que se cree más natural, es decir mediante señales perceptivas; que es el modo más cercano al modo en el que los humanos reconstruyen mentalmente escenas tridimensionales. Así la formulación de la función objetivo está basada únicamente en señales perceptivas. Estas señales, también llamadas regularidades o artefactos, han sido estudiadas desde el campo de la psicología [Hof00] y [Pal99] y algunas de ellas también han sido formuladas para la aplicación de las mismas en la reconstrucción tridimensional partiendo de una única vista [LS96]. Algunas de las formulaciones aquí ofrecidas ya fueron dadas por Lipson en su tesis [Lip98] aunque aquí se criticarán algunas de las formulaciones que en ella se dieron y se aportarán nuevas formulaciones alternativas a las mismas.

Además, determinar la categorización de las regularidades con el objetivo de evitar el mal comportamiento de la función objetivo es aún un problema abierto. En otras palabras, el objetivo es reducir la complejidad de la función objetivo para mejorar el ratio de éxito de la optimización del proceso de reconstrucción. Con este propósito, la distinción entre regularidades verdaderas y falsas, ya mencionada en la sección dedicada al trabajo relacionado sobre modelos tentativos es pertinente.

### 3.5.1. Regularidades falsas y verdaderas

Diferentes investigadores han estudiado las señales perceptuales o regularidades con planteamientos y objetivos diversos [LF92], [LS96], [GM95], [VM00a]. Para nuestro propósito, regularidad, artefacto o señal perceptiva pueden considerarse sinónimos, y entenderse como propiedades de la imagen que son reflejo de propiedades del modelo.

Siguiendo esta lógica, las falsas regularidades son características consideradas deseables en el modelo que no corresponden con ninguna característica detectable en la imagen. Mientras, las regularidades verdaderas reflejan alguna propiedad de la imagen que “debe”

corresponder con algunas propiedades equivalentes en el modelo buscado. La correspondencia entre la propiedad de la imagen y la propiedad del modelo no es determinista, pero tiene una alta probabilidad.

En otras palabras, todas las regularidades son propiedades que consideramos deseables que las cumpla el modelo que buscamos, pero no son restricciones: no son condiciones que tengamos la certeza de que el modelo las debe cumplir obligatoriamente.

La distinción entre regularidades falsas y verdaderas consiste en que las regularidades "falsas" no están relacionadas con características detectables en la imagen, sino con propiedades que, por otro camino, hemos llegado a considerar deseables en el modelo.

Resumiendo, una regularidad falsa es un artefacto para obtener modelos con alguna característica especial que consideramos deseable en el modelo 3D pero que no está ligada a ninguna señal identificada en el dibujo bidimensional. Por su parte, una regularidad verdadera es una característica detectada en el dibujo 2D y es muy probable que se corresponda con una característica en el modelo 3D.

### 3.5.2. Planicidad De Caras

La planicidad de caras es una regularidad verdadera para formas poliédricas, y probablemente la más importante de ellas. Es una regularidad verdadera porque el contorno de las caras se puede detectar en la imagen de partida, y es muy importante porque obliga a que la topología del modelo se aproxime mucho a la deseada (las caras alabeadas son fuente habitual de topologías erróneas). La identificación de los contornos de las caras se suele abordar como un problema de teoría de grafos como se verá extensamente en el Capítulo 4. En resumen, es obvio que esta regularidad tiene una fase previa clara: la detección de las caras del modelo en el dibujo bidimensional. De ello se hablará extensamente en el próximo capítulo.

Una vez calculadas las caras del modelo en el dibujo 2D, el cómputo de la regularidad comienza calculándose la ecuación del plano para cada una de las caras. Se ha decidido determinar ese plano como el plano más probable constituido por los vértices contenidos en esa cara. Así, para el cálculo de los coeficientes  $a$ ,  $b$ ,  $c$  y  $d$  de la ecuación del plano  $ax+by + cz + d = 0$ , se ha calculado el área del polígono, que forman los vértices y las aristas de la cara, proyectada sobre los planos coordenados. Así  $a$  se calculará como el área del polígono proyectada sobre el plano coordenado  $YZ$ ,  $b$  sobre el plano  $XZ$  y  $c$  sobre el  $XY$ . El coeficiente  $d$  se calculará mediante la sustitución del plano por uno de los

vértices del plano. De ese modo obtenemos el trío  $(a, b, c)$  que forma el vector unitario normal al plano, también llamado *vector director*.

$$\begin{aligned}
 A &= \frac{\sum y_j z_{j+1} - y_{j+1} z_j}{2} & a &= \frac{A}{\text{Módulo}} \\
 B &= \frac{\sum z_j x_{j+1} - z_{j+1} x_j}{2} & b &= \frac{B}{\text{Módulo}} \\
 C &= \frac{\sum x_j y_{j+1} - x_{j+1} y_j}{2} & c &= \frac{C}{\text{Módulo}}
 \end{aligned}$$

$$\text{Módulo} = \sqrt{A^2 + B^2 + C^2}$$

Ecuación 3-5

Una vez se conoce la ecuación del plano más probable se calcula la distancia de cada uno de los vértices que componen la cara con él:

$$\text{distancia}_i = |ax_i + by_i + cz_i + d|$$

Ecuación 3-6

y el cálculo del coste de esta regularidad se formula como la suma de distancias de cada uno de los vértices a su plano teórico.

### 3.5.3. Paralelismo De Líneas

Desde un punto de vista perceptual, y en términos generales, dos líneas que parecen paralelas en el dibujo deben ser paralelas en el mundo tridimensional. Esta regularidad es verdadera porque se basa en la detección de una propiedad geométrica de la imagen de partida. El fundamento geométrico de esta suposición es que en proyección cilíndrica el paralelismo es un invariante. El fundamento perceptual puede encontrarse en [Hof00.]

La formulación que de ella se hace en [Lip96] se considera adecuada y es por ello que no se va a repetir otra vez aquí. El coste de la falta de paralelismo de cada pareja de aristas evaluadas se computa en función del ángulo que forman. De modo que se compara el paralelismo de todas las parejas de aristas posibles en el modelo y el valor de la regularidad será la suma de todos los costes. Además se asume que algunas aristas de la

imagen pueden ser “casi” paralelas, es decir, que puede parecer que la intención del diseñador era hacerlas paralelas aunque geoméricamente no los sean. Entonces, se utiliza un coeficiente de ponderación que penaliza más la falta de paralelismo de aquellas líneas que parecen “más” paralelas en la imagen y penaliza menos aquellas cuyo paralelismo en la imagen es más dudoso.

El paralelismo de líneas es probablemente la regularidad más intuitiva, aunque tiene como principal limitación que sólo es totalmente cierta si la proyección del modelo es paralela o cilíndrica (es decir, axonométrica). Por lo tanto, no es válida para representaciones perspectivas.

### 3.5.4. MSDA

Marill [Mar91] alegando que los seres humanos perciben los objetos de la forma más sencilla posible, sugirió que era deseable obtener modelos con todos sus ángulos iguales. Así formuló la regularidad MSDA que es el acrónimo en inglés de “Mínima Desviación Estándar de Ángulos”. La búsqueda de ángulos iguales para todo un modelo se realiza sin ningún tipo de detección previa. Es decir se supone que ésta es una característica deseable y por ello debe ser cumplida por todas las entidades del modelo. Por lo tanto, ésta es una regularidad falsa puesto que no se detecta ninguna señal perceptiva en la imagen 2D que haga deducir que la característica deba cumplirse para cada pareja de líneas que concurran en cada vértice en el modelo 3D.

Sin embargo, la utilización de este tipo de regularidades en solitario puede considerarse como una estrategia que permite “despegar” al modelo del plano bidimensional. Pero si se utiliza junto a más regularidades, y puesto que no tiene porqué cumplirse en todo el modelo 3D, en muchos casos produce modelos distorsionados. Además hace que la función objetivo tenga un mayor número de mínimos locales de los cuales es difícil salir y complica la función objetivo.

Como se dijo previamente en §3.4, la experiencia de su utilización ha mostrado que es una herramienta válida para permitir escapar del mínimo trivial en un primer paso, y posteriormente que sean otras las regularidades que se optimicen para obtener la solución a la reconstrucción.

### 3.5.5. Verticalidad De Líneas

Esta regularidad se basa en la observación de que cuando dibujamos, siempre seleccionamos un punto de vista, de modo que la altura del objeto quede en dirección vertical. De este modo, la regularidad de verticalidad de líneas asume que cualquier línea que se dibuje 2D en el boceto inicial, se mantendrá vertical en 2D.

Desde mi punto de vista esta regularidad, una línea vertical en un boceto lleva normalmente la dirección que indica la altura en el modelo, pero ésta no suele ser paralela al plano de proyección (ver Figura 3-5). Así la regularidad considero que es verdadera aunque no estoy de acuerdo en la formulación que de ella se ofrece en [Lip96]. La formulación es cierta cuando se tiene una proyección caballera: dos de los tres ejes del triedro de referencia son paralelos al plano de proyección. En cualquier otro caso, para que esta fórmula sea cierta hay que hacer un cambio de sistema de referencia, es decir hay que encontrar la dirección vertical del modelo para poder computar realmente esta regularidad.

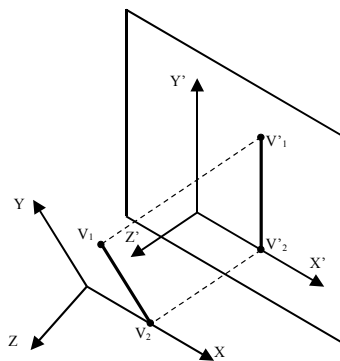


Figura 3-5. Cabeceo del sistema de referencia.

### 3.5.6. Isometría

La regularidad de isometría hace referencia a que la longitud de las entidades 3D deben ser proporcionales a las longitudes en 2D. Una variante de esta regularidad es la mínima desviación estándar de longitudes de segmentos que propusieron [BW96].

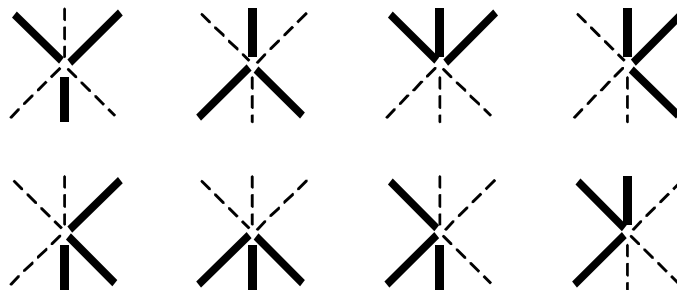
Esta regularidad aunque relativamente cierta, de acuerdo con las leyes de la proyección, únicamente es cierta en las tres direcciones principales de la proyección. Es decir, sólo si

se identifican las tres direcciones principales de un modelo, es de esperar que esta regularidad sea cierta. En cualquier otra dirección esto no tiene porqué cumplirse, o al menos en la misma proporción.

Por lo tanto, para que esta regularidad sea verdadera, la formulación matemática de esta regularidad debería limitarse a aquellas líneas que son paralelas a las direcciones principales del modelo. En cualquier otro caso ésta sería una regularidad falsa.

### 3.5.7. Ortogonalidad De Esquinas

Una unión de tres líneas que matemáticamente puede ser la proyección de una esquina 3D ortogonal es ortogonal en el espacio. La ortogonalidad de esquinas se plantea como que para determinar matemáticamente cuando una terna de segmentos de la imagen puede corresponderse con la proyección de una esquina ortogonal, se establece el hecho de que el ángulo formado entre segmentos consecutivos no debe exceder de  $90^\circ$ . De acuerdo con ello se presentan las ocho combinaciones posibles de segmentos que podrían corresponder a proyecciones de esquinas ortogonales de un modelo (ver Figura 3-6).



*Figura 3-6. Posibles proyecciones de esquinas ortogonales*

Para cada uno de los nodos definidos en la imagen, se evalúa si los segmentos de rectas concurrentes en el vértice corresponden a alguna de las ocho configuraciones definidas como proyecciones de esquinas ortogonales. Por lo tanto, es una regularidad muy similar a la MSDA, pero enfocada desde un punto más real, puesto que se deben comprobar las condiciones previas y se acepta como regularidad verdadera.

### 3.5.8. Ortogonalidad De Líneas

Esta regularidad, establece que todos los pares de segmentos de la imagen que concurren en un mismo vértice deben corresponderse con aristas perpendiculares en el modelo tridimensional, salvo que dichos segmentos resulten colineales. Ésta vuelve a ser una regularidad falsa, puesto que su aplicación no corresponde con ninguna señal perceptiva detectada en la imagen 2D. Además, es una generalización de la ortogonalidad de esquinas y refuerza a otra regularidad falsa como es la mínima desviación estándar de ángulos. Así se desaconseja totalmente su aplicación, puesto que no responde con detecciones reales en la imagen, y por tanto tampoco con las intenciones del bocetador.

### 3.5.9. Perpendicularidad De Caras

La perpendicularidad de caras está basada en la idea de que todas las caras adyacentes son perpendiculares entre sí.

De nuevo sin ningún tipo de justificación basada sobre la imagen inicial, se aplica indiscriminadamente sobre todas las caras. Por lo tanto es una falsa regularidad que al mismo tiempo refuerza todas aquellas regularidades de perpendicularidad expuestas hasta el momento.

### 3.5.10. Colinealidad de Líneas

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*

Esta regularidad también fue formulada por Lipson, pero de una forma, en mi opinión, ineficiente, puesto que aplica su fórmula a todos los pares de líneas. Es decir es una característica que puede ser reconocida en la figura 2D, y sin embargo se formula como si de una regularidad falsa se tratara. Es por ello que se propone una detección previa de la regularidad, calculándose  $w_i$  (coeficiente de ponderación de cada una de las líneas en función de su valor en el dibujo 2D) según:

## Percepción Artificial de Dibujos Lineales

$$w_i = \mu_{0,10} \left[ \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_2) + \max \left[ \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_3), \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_4), \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_5), \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_6) \right] \right]$$

Ecuación 3-7

siendo:

$$\mu_{a,b}(x) = \max \left[ 0, 1.1 \cdot e^{-\left(\frac{x-a}{b}\right)^2} - 0.1 \right]$$

Ecuación 3-8

Donde  $\hat{l}'_1$  y  $\hat{l}'_2$  son los vectores unitarios de la pareja de aristas comparadas, y si una tercera línea, con vector unitario  $\hat{l}'_3$ , correspondiente a una arista ficticia que une los extremos de las dos aristas comparadas (Figura 3-7). Si la línea  $L_3$  también es paralela a alguna de las anteriores significará que las dos aristas iniciales son colineales.

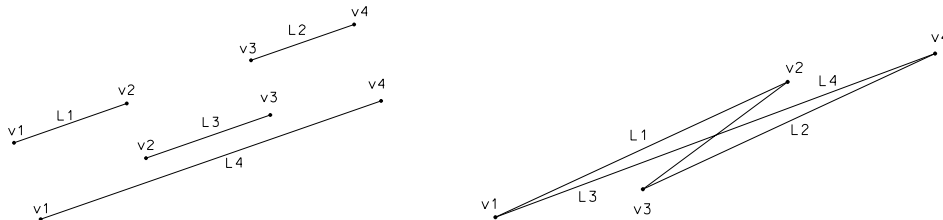


Figura 3-7. Comprobación de colinealidad de dos aristas  $L_1$  y  $L_2$

Para intentar paliar algunos errores ocasionales que se pudieran tener, se han considerado también los vectores unitarios  $\hat{l}'_4$ ,  $\hat{l}'_5$ ,  $\hat{l}'_6$  (Figura 3-8). Así el coste de la colinealidad para cada par de aristas queda expresado como:

$$\text{Coste\_Colinealidad}_{1,2} = \omega_{1,2} \cdot \left[ \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_2) + \max \left[ \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_3), \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_4), \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_5), \cos^{-1}(\hat{l}'_1 \cdot \hat{l}'_6) \right] \right]$$

Ecuación 3-9

La detección de líneas colineales así formulada ha dado buenos resultados, y con ello el computo en la función objetivo de esta regularidad es más eficiente, y sí se puede considerar verdadera.



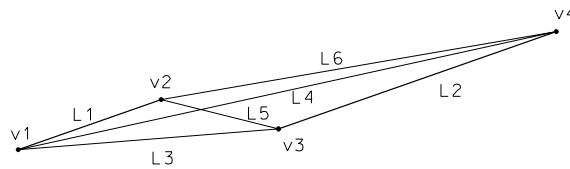


Figura 3-8. Las dos aristas comparadas ( $L_1$  y  $L_2$ ) y las cuatro mediciones ( $L_3$ ,  $L_4$ ,  $L_5$  y  $L_6$ )

### 3.5.11. Simetría Facial Oblicua.

Esta es una regularidad verdadera e interesante de la cual se hablará ampliamente en elCapítulo 5. Aunque creemos que la simetría facial puede ser considerada como una señal indirecta de una más general como es la simetría del modelo (Capítulo 6).

## 3.6. CONCLUSIONES DEL ESTUDIO DE REGULARIDADES

---

En este capítulo se ha explicado con detalle cuáles son los recursos con que cuenta la aplicación informática sobre la que se ha validado esta tesis. Además se ha hecho un repaso de las regularidades hasta el momento estudiadas. Se ha realizado una clasificación de ellas entre verdaderas y falsas.

Las falsas regularidades aunque pueden considerarse características deseables en el modelo final, no tienen porqué ser deseable en cualquier tipo de modelo, o en el modelo en conjunto. Puede que esa característica sólo sea deseable en algunas de las entidades del modelo y no en todas. Es decir, computando estas regularidades se exige a todas la entidades del modelo que cumplan unos requisitos que si se obtuviese el modelo teórica o perceptualmente correcto no se cumplirían. Es decir se le pide al modelo que cumpla unas propiedades que realmente no se desean.

Las regularidades falsas distorsionan la función objetivo: la complican, hacen lentos los tiempos de cálculo y hace aumentar el número de mínimos locales de la función, de los cuales no es sencillo salir en el proceso de optimización. Con todo ello se contribuye a que sea más difícil obtener el modelo deseado optimizando una función que no está formulada correctamente.

Desde mi punto de vista, estas regularidades no deberían sumarse a la función objetivo, dejando en ella aquellas regularidades que sí son ciertas, que tal y como están formuladas

## **Percepción Artificial de Dibujos Lineales**

---

hasta el momento son: Planicidad de caras, Paralelismo de Líneas, Ortogonalidad de Esquinas, Colinealidad de Líneas, y las Simetrías. Sin embargo, se considera que falsas regularidades pueden utilizarse para escapar del mínimo trivial, que es la imagen 2D, como se ha expuesto en la sección §3.4 sobre los modelos tentativos, y el uso de la MSDA en ellos.

# Capítulo 4. Detección de Caras de Poliedros en Representaciones 2D.

---

## 4.1. INTRODUCCIÓN

---

Un problema importante en el ámbito de la reconstrucción tridimensional de objetos poliédricos a partir de imágenes planas de los mismos es la detección temprana de sus caras. Se trata de una información a veces interesante para mejorar la eficiencia de ciertos procesos y a veces imprescindible para que algunos métodos sean viables. Pero también se trata de una información difícil de obtener.

En efecto, el problema es complejo porque para que la información resultado de la detección de caras sea útil al proceso de reconstrucción, las caras se deben determinar en la imagen, antes del proceso de reconstrucción y, por tanto, sin disponer de ninguna información de la profundidad del modelo. Por ello, es insuficiente un enfoque basado en la geometría proyectiva, que vincule la imagen disponible con el modelo deseado. Los métodos propuestos más efectivos hasta la fecha utilizan reglas heurísticas (principalmente extraídas de la psicología de la percepción, aunque muchas veces no de un modo explícito) y formulan el problema mediante la teoría de grafos [Wil86].

Planteado como un problema de grafos, el problema de la detección de caras consiste en determinar todos los *ciclos* de un grafo que cumplan unos requisitos dados. Se debe recordar que un *ciclo* (o circuito) es un camino cerrado (que empieza y acaba en el mismo vértice); que, a su vez, un *camino* es un paseo en el cual cada arista y cada vértice se

recorre una sola vez, y que un *paseo o recorrido* es una sucesión finita de vértices y aristas, donde se van enlazando sucesivamente vértices con aristas.

Para seguir éste capítulo se deben tener unos conocimientos básicos de la terminología de la teoría de grafos. Un buen punto de partida, tanto para familiarizarse con la terminología de grafos, como para resolver el problema propuesto, es el estado del arte sobre algoritmos para resolver problemas de grafos que se puede encontrar en [DET94]. Su contenido es suficiente para conocer el estado del arte del tema que nos ocupa.

### 4.1.1. Figuras vistas como grafos

Los *grafos* están formados por nodos conectados mediante líneas. Habitualmente, los nodos se denominan vértices, y las líneas aristas, lo que determina una clara analogía entre los grafos y las imágenes obtenidas como resultado de una proyección de un modelo poliédrico alámbrico o B-rep. La proyección de una arista del modelo resulta en una línea de la imagen, y la proyección de un vértice del modelo es un nodo de la imagen.

Comparar la imagen resultante de una proyección de un objeto poliédrico con un grafo es útil cuando existen relaciones unívocas entre nodos y vértices, y entre líneas y aristas. A tal fin, se debe exigir un “punto de vista general” para dibujar la figura, porque garantiza *la regla de las visiones genéricas* [Hof00]:

*“Construir únicamente aquellos mundos visuales para los cuales la imagen es una visión estable (es decir, genérica)”.*

De hecho, basándose en esta regla de la percepción, Hoffman estableció otras dos reglas:

- Regla 1: Interprete siempre la línea recta de una imagen como una línea recta tridimensional.
- Regla 2: Si los extremos de dos líneas coinciden en una imagen, interprete siempre que coinciden en tres dimensiones.

Estas reglas explican porqué la percepción rechaza incontables interpretaciones tridimensionales de las imágenes planas. Si observamos la imagen de Kopfermann, en la parte izquierda de la Figura 4-1, y de acuerdo con la Regla 1, debemos interpretar cada línea como una arista recta, sin esquinas. La regla 1 impide interpretar la imagen como un cubo, debido a que algunas líneas deberían descomponerse en dos aristas. También la regla 2 impide interpretar el nodo central como dos vértices diferentes del cubo. Con un pequeño movimiento del objeto original, obtenemos una vista genérica (la figura de la

derecha) y la imagen cumple las dos reglas anteriores. Entonces nos resultará fácil reconstruir el cubo.

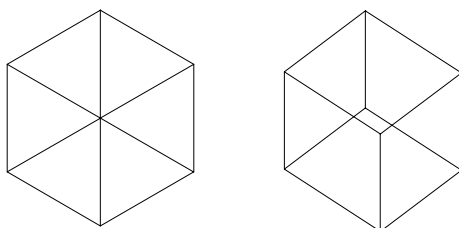


Figura 4-1. Figuras de Kopfermann

Resumiendo, la analogía entre imágenes y grafos, junto con la regla de las visiones genéricas, hace fácil identificar cada vértice y cada arista de un modelo poliédrico alámbrico o B-rep. Además, permite abordar el problema de la detección de caras de poliedros como un problema de determinación de ciclos que cumplen ciertas reglas.

### 4.1.2. Sobre los poliedros y sus caras

Los poliedros son un tipo particular de *politopos*, que son todos aquellos subconjuntos del espacio euclideo que se construyen con elementos rectos. En efecto, los *poliedros* son cuerpos geométricos limitados por un conjunto de polígonos planos, o caras, tales que cada uno de los lados (aristas) pertenece a un número par de polígonos contiguos, y dos polígonos cualesquiera con un lado común pertenecen a distintos planos.

El número de caras de un politopo no está determinado, pero el de algunos poliedros sí. Además, los métodos de detección de caras no son universales (al menos no lo son los más eficientes), por lo que es conveniente distinguir diferentes tipos de poliedros.

Se denomina *2-variedades* (o "2-manifold") a los cuerpos en los que en cualquier punto de su superficie se puede construir un disco, de tamaño arbitrariamente pequeño, que sea topológicamente equivalente a un disco sobre un plano. En la parte izquierda de la Figura 4-2 se ha representado un poliedro 2-variedad. En el centro se ha representado un politopo (no es un cuerpo sólido) sin variedad (no existe ningún disco topológicamente equivalente a un disco sobre un plano para el punto dado), y en la parte derecha se ha representado un poliedro que no es 2-variedad (se puede determinar más de un disco sobre planos diferentes para el punto dado).

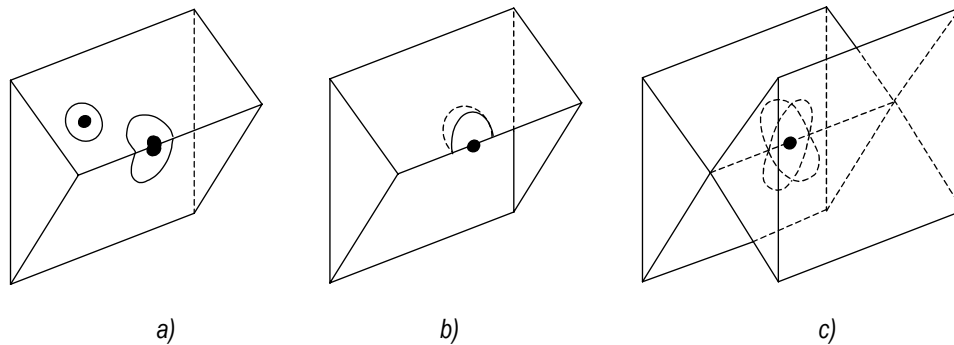


Figura 4-2.. a) Poliedro 2-variedad, b) Polítopo sin variedad, c) Poliedro que no es una 2-variedad.

La diferenciación es importante, porque el número de caras de un poliedro que no sea 2-variedad no puede ser determinado a priori. Por el contrario, el número de caras de un poliedro 2-variedad está determinado por la fórmula de Euler-Poincaré:

$$F + V - E = 2 \cdot (S - G)$$

*Ecuación 4-1. Ecuación de Euler-Poincaré*

Siendo  $F$  el número de caras,  $V$  el número de vértices,  $E$  el número de aristas,  $S$  el número de objetos diferentes y  $G$  la suma de géneros de todos los objetos. Para un objeto simple,  $S$  es la unidad.

El género de una superficie se puede medir a partir del número máximo de curvas cerradas que se pueden inscribir en la superficie sin dividirla en más de una región. Las curvas no deben intersectarse entre ellas y el género es el número de curvas menos uno. Cualquier curva cerrada divide a la esfera en dos regiones, y, por tanto, el género de la esfera es cero. En un toro, se necesitan al menos dos curvas cerradas para aislar una región, por lo que el género es uno. Del mismo modo, una esfera con una "asa" también es de género 1, porque es topológicamente equivalente a un toro (ver Figura 4-3).

Los poliedros sin agujeros son de género cero (topológicamente equivalentes a la esfera). Los poliedros con agujeros pasantes son equivalentes a una esfera con asas (un asa por cada agujero) y su género es, en general, igual al número de agujeros pasantes.

Los poliedros con agujeros ciegos (no pasantes) son equivalentes a la esfera (género cero), pero se representan mediante grafos no conexos.

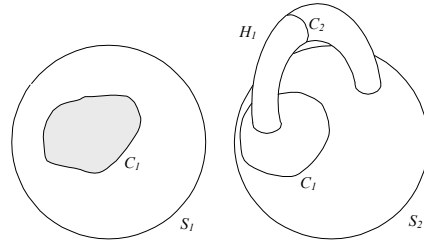


Figura 4-3. Esfera  $S_1$  con curva  $C_1$  que divide su superficie en dos regiones (izquierda), Esfera  $S_2$  con asa  $H_1$  y dos curvas  $C_1$  y  $C_2$  que dividen su superficie en dos regiones (derecha)

En los poliedros sin agujeros, cada cara está determinada por un polígono. En los poliedros con agujeros (tanto si son pasantes como si no) algunas caras están determinadas por más de un polígono; uno de los cuales es exterior. A los polígonos interiores de una cara de un poliedro con agujero se les suele denominar anillos (*rings*).

En general es fácil determinar que un grafo corresponde a un poliedro con agujeros, porque basta comprobar que se trata de un grafo no conexo. Se dice que un grafo es *conexo* si para cada par de vértices existe al menos un camino de una o más aristas que conecta los dos vértices. Hay excepciones, como la mostrada en la parte derecha de la Figura 4-4-d).

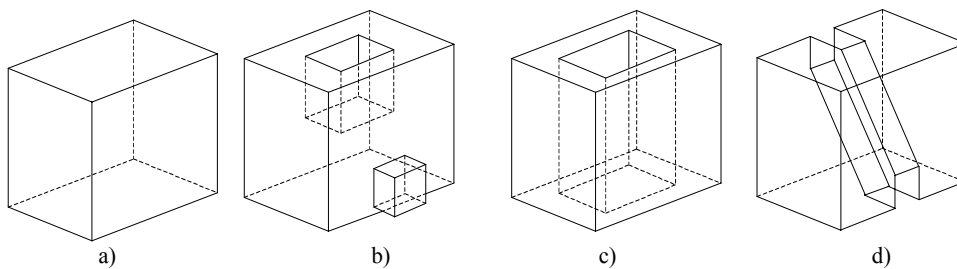


Figura 4-4. a) Poliedro de género cero y representado por un grafo conexo, b) poliedro de género cero representado por tres componentes conexas, c) poliedro de género 1 y representado por dos componentes conexas, d) poliedro de género 1 representado por un grafo conexo.

Resumiendo, el número de caras de un poliedro *euleriano* (2-variedad de género cero) puede ser fácilmente determinado, contando el número de vértices y aristas del grafo que representa a dicho objeto.

$$F + V - E = 2$$

Ecuación 4-2

En el presente capítulo, describiremos un método sencillo y eficiente para determinar si la representación corresponde a un poliedro euleriano, y encontrar, a continuación, los ciclos que representan todas y cada una de sus caras. No obstante, veremos que hay complicaciones, en el caso de que la conectividad del grafo no cumplan unos requisitos mínimos.

### 4.1.3. Sobre la conectividad de los grafos

Ya hemos indicado que un grafo es conexo si es posible alcanzar cualquier vértice desde cualquier otro vértice a través de aristas. Existen, no obstante otras definiciones que son pertinentes al caso.

La *conectividad de un grafo* es el número mínimo de vértices que deben ser eliminados para obtener un grafo no conexo, o un grafo con un único vértice. Se dice que un grafo es *k-conexo* si su conectividad es mayor o igual que  $k$ . Un conjunto de vértices se denomina *separador* si su eliminación produce un grafo no conexo, o un grafo con un único vértice. Si la eliminación de un solo vértice produce un grafo no conexo, éste se denomina *vértice de corte* o *articulación*.

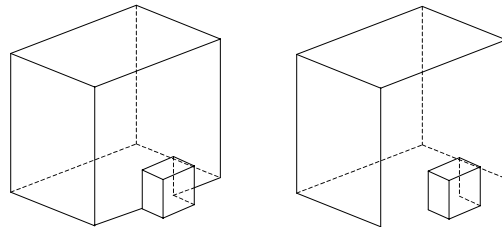


Figura 4-5. Grafo conexo de grado mínimo igual a tres y con una conectividad de aristas de dos (izquierda), dos bloques no conexos resultantes de eliminar las dos aristas del conjunto de corte (derecha).

Análogamente la *conectividad de aristas* de un grafo es el número mínimo de aristas que deben ser eliminados para obtener un grafo no conexo, o un grafo con una única arista. Un *conjunto de corte* ("cut-set") es aquel conjunto mínimo de aristas cuya eliminación desconecta un grafo. La *conectividad de aristas* es el número mínimo de aristas que definen un conjunto de corte (Figura 4-5). Una *arista de corte* es aquella que por sí misma desconecta el grafo.



Determinar la conectividad es un paso previo a la clasificación del poliedro representado en un grafo, y como ya se ha dicho, esta clasificación es necesaria para elegir el método de detección de caras más eficiente. Pero, calcular la conectividad no es trivial, aunque su límite superior sí que está determinado. En efecto, se demuestra que la conectividad de vértices ( $k_V$ ) es siempre menor o igual que la conectividad de aristas ( $k_E$ ), y ambas son menores o iguales que el *grado*  $\delta$  del vértice de menor grado del grafo (siendo el grado o *valencia* el número de aristas que concurren en un vértice).

$$k_V(G) \leq k_E(G) \leq \delta$$

Ecuación 4-3

Los métodos más usuales de cálculo de la conectividad se basan en determinar el máximo número de *camino disjuntos*. Es decir, el máximo número de caminos que enlazan cualquier pareja de vértices sin tener en común ninguna arista.

#### 4.1.4. Sobre la planaridad de los grafos.

Un grafo se dice que es *plano* cuando estando dibujado en un plano, sus aristas sólo se intersectan en los vértices. Es decir, no tiene cruces en sus aristas. Y se dice que un grafo es *planar* cuando es *isomorfo* de un grafo plano.

Dos grafos se dice que son *isomorfos* si entre sus vértices hay una correspondencia uno a uno, con la propiedad de que el número de arista que unen cualquiera dos vértices de un grafo es igual al número de aristas que unen los vértices correspondientes en el segundo grafo.

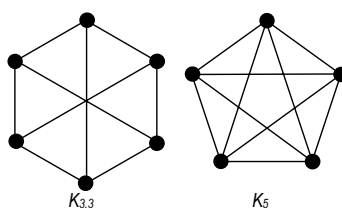


Figura 4-6. Grafos  $K_{3,3}$  y  $K_5$ .

Kuratowski dio la primera caracterización de los grafos planos, mediante el teorema que dice: Un grafo es plano si y sólo si no contiene ningún *subgrafo isomorfo* al  $K_{3,3}$  o al  $K_5$  (ver Figura 4-6).

Pero desgraciadamente, su caracterización no incluía un algoritmo para comprobar la planaridad. Posteriormente se han propuesto diferentes algoritmos, con coste de ejecución lineal, tal como se resume en el estado del arte de Di Battista [DET94].

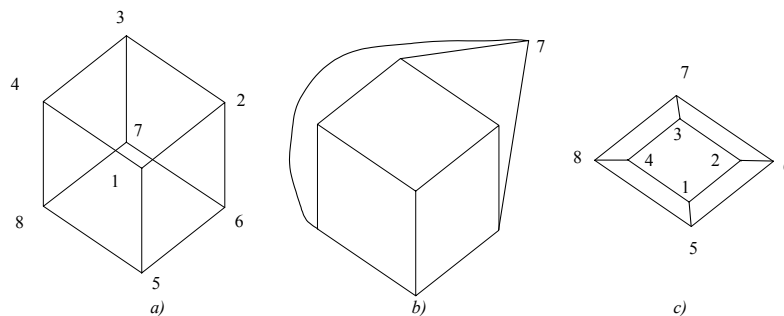


Figura 4-7. a) Grafo original no plano (izquierda), b) y c) dos formas diferentes de sumergirlo en un plano.

Podemos resumir, de acuerdo con Di Battista, que los algoritmos que encuentran representaciones planas de grafos planos con grado lineal de complejidad, son difíciles de entender e implementar. Lo que supone una limitación importante para usarlos en aplicaciones prácticas.

## 4.2. ALGORITMOS DE DETECCIÓN DE CARAS ESTUDIADOS

---

La evolución histórica ha llevado a métodos de detección de caras esencialmente basados en planteamientos topológicos, en detrimento de los métodos basados en planteamientos geométricos que fueron utilizados inicialmente. En efecto, los métodos de etiquetado de dibujos lineales pueden ser entendidos como precedentes de la detección de caras, desde Huffman [Huf71] y Clowes [Clo71], hasta Varley y Martín [VM00a], [VM00b]. Pero, actualmente, los métodos más generales para la detección de caras en dibujos lineales se pueden clasificar en dos grupos. En ambos grupos se asimila el dibujo a un grafo. En un caso se detectan las caras del grafo sumergiéndolo en un plano (aplicando la teoría de

grafos exclusivamente), y en otro caso se determina un conjunto amplio de caras “potenciales” y posteriormente descartan las que no cumplen ciertos criterios (que pueden ser geométricos, perceptivos, heurísticos, etc.).

Los métodos de inmersión del grafo, o planarización, tienen una aplicabilidad limitada a aquellas figuras representables por medio de grafos planares. Por su parte, los métodos de determinación de caras potenciales y posterior obtención de las caras reales tienen costes computacionales inaceptables para aplicaciones interactivas. Además, los criterios heurísticos que emplean son una mezcla de propiedades métricas, propiedades de la percepción y otras consideraciones que no han sido objeto de ningún tratamiento sistemático.

### 4.2.1. Roberts

En el trabajo de Roberts sobre percepción de objetos tridimensionales a partir de dibujos lineales, se incluía un apartado completo sobre “reconocimiento de polígonos” (capítulo V, en [Rob63]).

Su método comienza clasificando todas las aristas conectadas a cada vértice dependiendo de su orientación (por su ángulo respecto a una dirección de referencia). Entonces, la búsqueda de cualquier cara empieza en una arista aleatoria, y se elige aquella arista adyacente (conectada al vértice final de la arista precedente) que forme el menor ángulo según un sentido de giro. El proceso se repite hasta que el vértice inicial es alcanzado de nuevo, y el circuito se cierra en un ciclo que define la cara. Moviéndose en el contorno de la figura (cara exterior infinita) en un sentido y en el resto de caras interiores en el sentido opuesto. Se han obtenido todas las caras interiores cuando todas las aristas interiores, que no pertenecen al contorno, son compartidas por dos caras (aristas 1-2, 1-4 y 1-6 en la Figura 4-8).

Roberts usó este método para identificar caras en representaciones que no contienen aristas ocultas, dado que asumía representaciones de modelos opacos. Es por ello por lo que se considera el contorno de la proyección como una cara exterior infinita.

El método de Roberts no es útil para nuestros objetivos, porque, en nuestro método, la información que proporcionan las aristas ocultas deben estar representadas en la imagen.

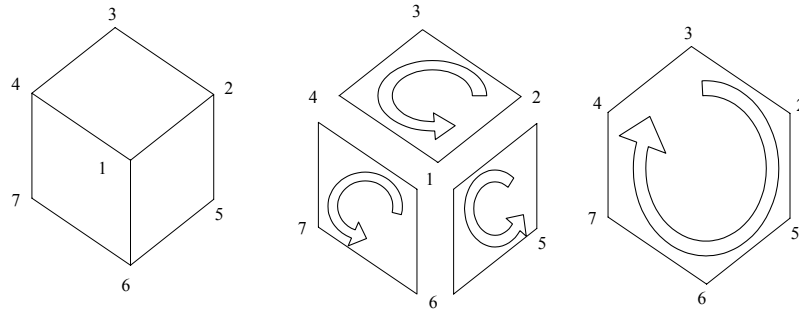


Figura 4-8. Detección de caras en una representación sin aristas ocultas

### 4.2.2. Leclerc y Fishler

En 1992 Leclerc y Fischler [LF 92] presentaron un algoritmo basado en las siguientes apreciaciones heurísticas:

- ❑ Dos enlaces (o aristas proyectadas) paralelos representan aristas paralelas en el espacio: Los enlaces que conecten dos nodos de dos enlaces paralelos, 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.
- ❑ El contorno formado por un circuito simple de líneas del dibujo que no contengan enlaces internos, se corresponde con una cara plana.

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

En definitiva, el algoritmo que proponían es:

- ❑ Encontrar todos los ciclos de más de tres líneas que no se autointersecten. Una vez encontrados todos los ciclos, se toman como caras:
  - Aquellos ciclos que estén totalmente vacíos en su interior de vértices y líneas.
  - Aquellos ciclos que son convexos y completamente libres de ciclos internos.

- Se toman también como caras aquellos circuitos definidos por el siguiente algoritmo, que no son subconjuntos de ninguna cara anterior:

Cualquier camino de tres aristas en el que la primera y la tercer arista son paralelas. En general, se extiende el criterio para considerar como caras a todos los caminos en los que las aristas impares sean paralelas entre sí, y las pares también sean paralelas entre sí.

Cualquier pareja de líneas paralelas determina un plano, independientemente de que formen parte de un ciclo o un camino. Por tanto, se considera que definen una cara, salvo cuando definen un subconjunto de alguna cara previa.

La primera consideración es que Leclerc y Fischler excluyen los ciclos de tres líneas porque definen figuras trivialmente planas, por lo que no son útiles para su objetivo que es comprobar la planicidad de las caras detectadas. En una generalización del método destinada a detectar todas las caras, no se podrían excluir los ciclos de tres líneas.

La segunda consideración es que Leclerc y Fischler definen como caras a las parejas de líneas paralelas, aunque no formen parte de un ciclo. Este criterio es correcto cuando el objetivo es detectar el máximo número posible de condiciones de planicidad del objeto a reconstruir. Por tanto, es correcto en un algoritmo de detección de *condiciones de planicidad*; pero no detecta *auténticas caras* del objeto, por lo que se debería eliminar de un algoritmo de detección de caras en sentido estricto.

Por lo demás, se trata de un método totalmente heurístico, que aunque ofrece buenas apreciaciones acerca de cómo los humanos percibimos las figuras planas, ofrece algunos errores en sus resultados. Además, la aplicación de un método de este tipo hace que desde un primer momento se deban hacer multitud de comprobación, lo que hace que el algoritmo sea computacionalmente muy costoso y lento.

Adicionalmente, la condición métrica de paralelismo impide la aplicación del método a figuras que no mantengan invariantes las relaciones de paralelismo del modelo tridimensional, es decir, que no es válido para las proyecciones perspectivas.

### 4.2.3. Lipson y Shpitalni

Los estudios de estos autores son posteriores a los de Leclerc y Fischler, y en su artículo [LS96] se definen dos métodos diferentes. Proponen un método particular para detectar

## Percepción Artificial de Dibujos Lineales

---

caras de poliedros 2-variedad de género cero, y un método más general que en un primer paso determina caras candidatas y posteriormente descarta las caras falsas.

El método para detectar caras de poliedros 2-variedad de género cero propuesto por Lipson y Shpitalni se limita a sumergir el grafo en un plano (Figura 4-7) y determinar las caras del grafo sumergido. Caracterizan los casos en que la solución no es única (caso de grafo con conectividad de aristas menor o igual que tres), y aplican métodos heurísticos para elegir una de las diferentes soluciones posibles. Dan como referencia el libro de Gibbons [Gib85], para el algoritmo que permite detectar el nivel de conectividad del grafo.

Lipson y Shpitalni no dicen explícitamente cómo detectar las caras del grafo sumergido, aunque un método sencillo a aplicar en ese caso podría ser el de Roberts, descrito arriba, que puede adaptarse fácilmente para obtener una implementación práctica. Su propuesta tampoco presenta ningún algoritmo para sumergir el grafo en un plano. Este problema sí que es complejo; aunque hay métodos propuestos para solucionarlo, tal como se detalla en el estado del arte de Di Battista [DET94]. Además, para encontrar un dibujo plano de un grafo, primero hay que determinar si el grafo es planar.

Resumiendo, se puede decir que el método de Lipson y Shpitalni es teóricamente bueno y simple, pero computacionalmente presenta serias complicaciones. Además, su rango de validez es ciertamente muy limitado (poliedros 2-variedad de género cero). La única justificación reside en que el segundo método (que veremos a continuación) presenta unos tiempos de cálculo excesivos para una aplicación interactiva, y los autores buscaron un modo de reducir estos tiempos, aunque fuera para un subconjunto de los problemas considerados.

El segundo método de detección de caras propuesto por Lipson y Shpitalni está basado en reglas topológicas. Este método fue el que en un primer momento se seleccionó para su aplicación en el programa REFER [Con01]. Este algoritmo consta de varias fases:

- ❑ En un primer lugar, se identifican los circuitos fundamentales. En un grafo euleriano, el número de circuitos fundamentales es el resultado de  $E-V+1$ , siendo  $E$  el número de aristas y  $V$  el número de nodos o vértices.
- ❑ Se generan las caras potenciales del grafo: se calculan el resto de los circuitos cíclicos del grafo, por combinación de circuitos fundamentales, que constituyen todas las caras potenciales del modelo. La combinación se realiza sumando los enlaces de los dos circuitos fundamentales y eliminando los enlaces comunes a

las dos. Los criterios para no aceptar una combinación como una cara potencial son:

Si al combinar dos o más circuitos fundamentales, el circuito resultante tiene aristas que se interceptan entre sí, no se acepta.

Si al combinar dos o más circuitos fundamentales se obtiene más de un circuito resultante, tampoco se acepta.

- ❑ Se seleccionan las caras del modelo: recorriendo un árbol de búsqueda mientras se evalúan condiciones de compatibilidad entre las caras, basadas en el rango máximo de las aristas y de los vértices, y en el grado de los vértices. Para reducir el número de posibles soluciones se comprueba el teorema de adyacencia. Y por último, se aplica una función de conformidad, basada en la percepción, que dice:

*“Un humano típicamente selecciona una configuración que verifique que tantos vértices como sean posibles participarán en tantas caras como sean posibles”.*

- ❑ Se selecciona la solución final. Se aplica cuando la función de conformidad tiene el mismo valor para más de una combinación de caras. En ese caso la selección se realiza basada en otros indicadores como es la percepción de las regularidades de la imagen, en concreto en una regularidad: la ortogonalidad oblicua de la cara resultante.

Este método fue utilizado y mejorado por Conesa [Con01]. Pero se comprobó que se trataba de un método que además de ofrecer fallos, era muy costoso computacionalmente. Esto se debía a que estaba basado en demasiadas reglas y condiciones comparativas que hacían de la detección de caras una función muy lenta incluso para modelos sencillos. En modelos más complicados, se llegaba a tiempos inaceptables para una aplicación interactiva (¡del orden de minutos!).

### 4.2.4. Courter y Brewer.

El trabajo de Courter y Brewer [CB86] es la propuesta más eficiente que hemos encontrado para resolver el problema de la detección de caras en poliedros eulerianos.

El método pertenece al grupo de métodos que utilizan un planteamiento exclusivamente topológico. Está basado en la determinación de circuitos fundamentales y su combinación

siguiendo ciertos criterios para seleccionar aquellas caras potenciales obtenidas por combinación de circuitos fundamentales que tengan mayor probabilidad de ser caras del objeto. Se basa en el método de 1983 de Ganter y Uicker [GU 83]. Y al igual que en aquel método, los circuitos fundamentales se determinan a partir del árbol generador creado mediante BFS (Breath-First Search) y la matriz de adyacencia que Paton [Pat69] utilizó para la búsqueda de circuitos fundamentales.

Los circuitos fundamentales son un conjunto de circuitos linealmente independientes y, siguiendo a Paton [Pat69], se obtienen a partir del árbol generador. El *árbol generador* es un sub-grafo en el cual todos los vértices del grafo original se mantienen, pero solo quedan las aristas que mantienen unidos todos los vértices del grafo, sin formar ningún circuito cerrado. Las aristas del grafo que no pertenecen al árbol generador se llaman *cuerdas*. Cada vez que se añade una cuerda se forma un circuito independiente.

Para los cálculos se utiliza La *Matriz de Incidencia*,  $[X]_{N^\circ \text{ Vértices} \times N^\circ \text{ Vértices}}$ , que es una matriz donde cada fila y cada columna representa a un vértice y el contenido de cada uno de sus elementos  $x_{ij}$  será la unidad cuando entre el vértice  $i$  y el  $j$  exista en el grafo un enlace que los una. Si los dos vértices no están conectados el contenido será nulo.

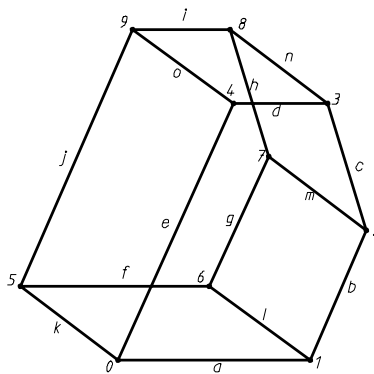


Figura 4-9. Grafo conexo.

Una vez obtenidos los circuitos fundamentales, se plantea un método para convertir estos circuitos en caras verdaderas. Para ello se combinan parejas de circuitos fundamentales y se obtiene su *Diferencia Simétrica*. La Diferencia Simétrica se define como “el resultado de la unión de los conjuntos de aristas de dos circuitos dados menos su intersección”. En otras palabras, la diferencia simétrica es el circuito formado por la unión de los dos grafos dados, menos su intersección.



## Capítulo 4. Detección de Caras de Poliedros en Representaciones 2D.

Para decidir qué parejas de ciclos se deben combinar se utilizan dos criterios básicos tomados de la Teoría de Grafos [Deo74]:

- ❑ Una arista no puede pertenecer a más de dos circuitos fundamentales.
- ❑ La suma del número de todas las aristas en todos los circuitos debe ser mínima.

La primera regla, obviamente limita el ámbito del método a poliedros 2-variedad, y es la que garantiza que el número de caras a buscar pueda determinarse por el método de Euler.

La segunda regla se cumple al minimizar la interacción entre aristas de diferentes ciclos. Por ello, se realiza un proceso que permite reducir cada vez aquellos dos circuitos con mayor número de aristas. El proceso se gestiona de un modo computacionalmente muy simple gracias a que se definen dos matrices:

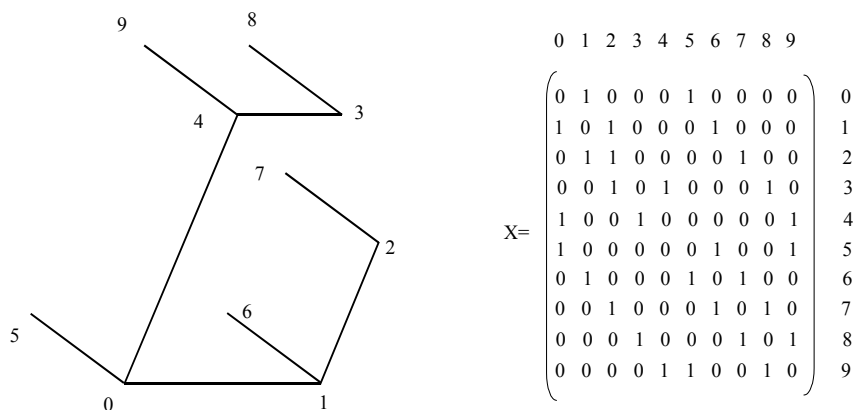


Figura 4-10. Árbol BFS y Matriz de Incidencia

- ❑ La *Matriz de Circuitos*,  $[C]_{N^{\circ} \text{ Circuitos} \times N^{\circ} \text{ Aristas}}$ , donde se registra la relación entre las aristas que pertenecen a cada circuito, de forma que  $C(i, j) = 1$  si la arista  $j$  es una de las que componen el circuito  $i$ , y  $C(i, j) = 0$  en caso contrario.
- ❑ Multiplicando los elementos de la *Matriz de Circuitos* se pueden calcular los elementos de la *Matriz de Interacciones*  $[Z]_{N^{\circ} \text{ Circuitos} \times N^{\circ} \text{ Circuitos}}$ . En efecto, cada elemento de la *Matriz de Interacciones*  $Z(i, j)$ , se obtiene como producto de  $C(i) \times C(j) = Z(i, j)$ . El elemento  $Z(i, j)$  de esta matriz indica el número de aristas que tienen en común los circuitos  $i$  y  $j$ . En los elementos de la diagonal

## Percepción Artificial de Dibujos Lineales

$Z(i,i)$  se obtiene que serán el número de aristas que contiene el circuito  $i$ -ésimo.

La reducción comienza comparando aquel circuito que tenga un mayor número de aristas (mayor valor de  $z_{ii}$ ) con el circuito que más aristas tiene coincidentes con el primero (con la cara  $i$  que mayor valor del elemento  $z_{ij}$  posea). Antes de sustituir el circuito elegido (el de mayor número de aristas) por el resultado de la Diferencia Simétrica, se realiza un test cuyo objetivo es comprobar si el circuito puede tener menos aristas después de la reducción.

Esto se comprueba del siguiente modo: llamando  $Z(i,i)$  a los elementos de la Matriz de Circuitos antes de la reducción y  $j$  al circuito con el cual se quiere reducir el circuito  $i$ , para que el circuito final tenga menor número de aristas que el inicial, se debe cumplir que:

$$Z(j, j) < 2 \cdot Z(i, j)$$

Ecuación 4-4

puesto que  $Z(i, j) = Z(j, i)$ .

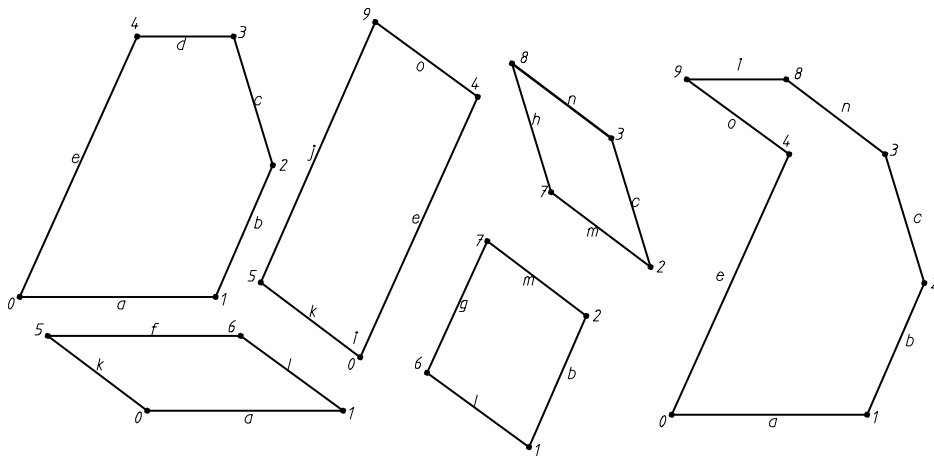


Figura 4-11. Posibles circuitos fundamentales de la Figura 4-9

Puede darse el caso de que la Ecuación 4-4 dé un valor límite. Esto ocurre si el número de aristas de la posible solución es el mismo que el número de aristas del circuito inicial. En estos casos Courter y Brewer adoptaron el criterio de sumar la columna  $i$  de la Matriz de Interacciones antes y después de la posible solución, que es un indicador de la interacción

## Capítulo 4. Detección de Caras de Poliedros en Representaciones 2D.

---

de la cara. Si la suma después de la aplicación de la posible solución es menor que la inicial, se acepta el cambio. En caso contrario no se acepta el cambio.

Una vez realizado el test (Ecuación 4-4), si es positivo se acepta la modificación, y se vuelve a buscar el circuito con mayor número de aristas. En caso negativo se vuelve a tomar como circuito a reducir el  $i$ , pero como  $j$  el siguiente circuito que tenga mayor interactividad con el  $i$ .

El proceso de reducción debería continuar hasta que la suma de cada una de las columnas de la Matriz de Circuitos nos diera un resultado menor o igual que 2, puesto que una misma arista como máximo no debe pertenecer a más de 2 caras.

Puesto que el número de circuitos fundamentales es el número de enlaces-nodos+1, y siguiendo la fórmula de Euler el número de caras de un grafo plano es enlaces-nodos+2. Sólo falta por añadir la cara que cerraría el elemento Euleriano. Para ello se buscan todas aquellas aristas que sólo pertenecen a un circuito y ellas formarán la última cara a añadir (cara exterior del grafo plano).

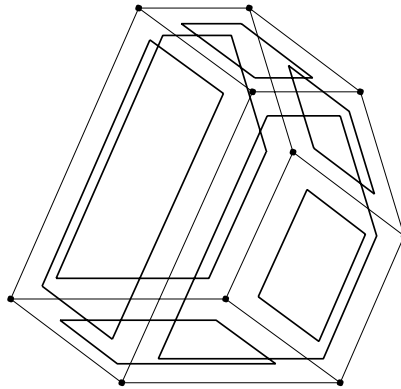


Figura 4-12. Caras detectadas en el dibujo 2D

El algoritmo de Courter y Brewer incluye un post-procesado de ciertas anomalías. El proceso consiste en detectar y borrar las caras inválidas, para, posteriormente repetir el proceso de detección de caras sobre el subgrafo resultante de considerar aquellas aristas del grafo original que aún pertenecen a menos de dos caras.

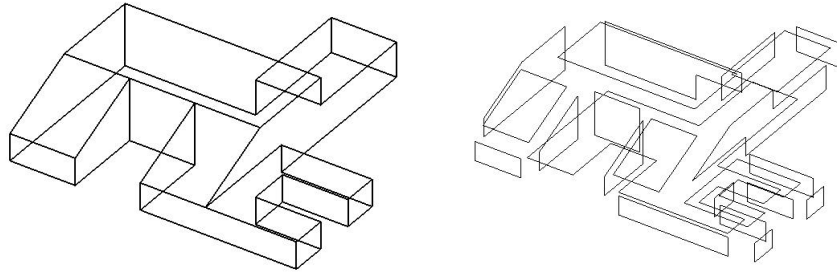


Figura 4-13. Cálculo de caras: 57 aristas, 37 vértices y 22 caras

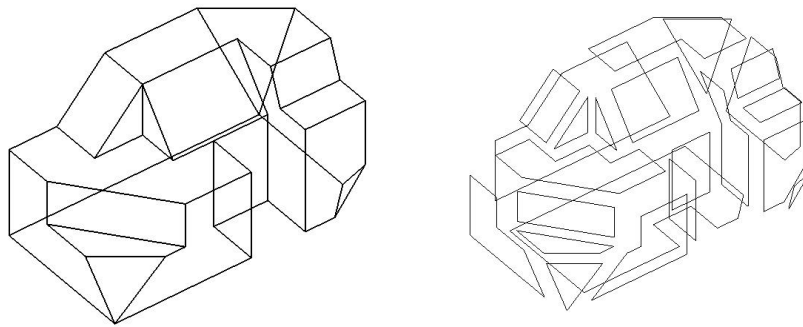


Figura 4-14. Cálculo de caras: 55 aristas, 35 vértices y 22 caras

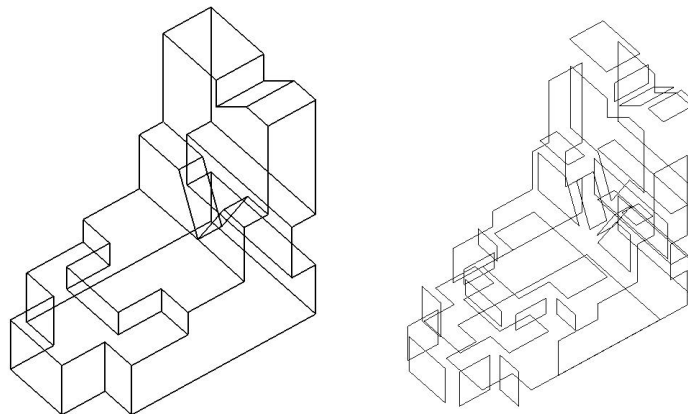


Figura 4-15. Cálculo de caras: 85 aristas, 56 vértices y 31 caras

Este algoritmo, a pesar de su antigüedad, por su sencillez tanto conceptual como computacional se piensa que es mucho más efectivo que los hasta ahora estudiados en la bibliografía. Sin embargo, este algoritmo no acepta objetos que no sean cerrados (2-variedad, que no admiten aberturas), mientras que el descrito por Lipson y Shpitalni sí. Sin embargo, por su eficiencia computacional se ha implementado este algoritmo en la versión actual de la aplicación REFER con ciertas mejoras, que se exponen a continuación.

### 4.3. MEJORAS APORTADAS

---

A pesar de que el algoritmo de Courter y Brewer ofrece buenos resultados en la mayoría de modelos testados, se observa que hay un conjunto de modelos, aquellos que son 2-conexos, en los que el algoritmo no siempre obtiene buenos resultados.

Un grafo 3-conexo tiene una importante propiedad, y es que tiene una única posible configuración cuando se le convierte en grafo plano. Sin embargo, aquellos grafos que pueden ser separados en dos sub-grafos independientes al eliminar dos vértices, o al cortar dos de sus aristas, es decir, los grafos 2-conexos, al hacerlos planos pueden adoptar dos posibles soluciones, y con ello dos posibles configuraciones y resultados de la detección de caras.

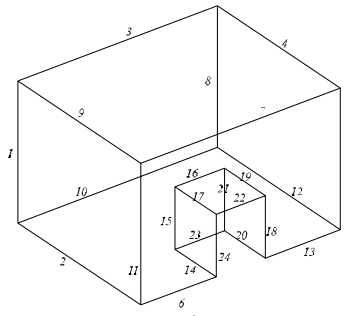


Figura 4-16. Objeto 2-conexo

En la figura siguiente (Figura 4-17) se muestra un objeto que aunque simple de interpretar desde la percepción, no parece tan fácil desde la teoría de grafos debido a que es 2-conexo y acepta dos posibles grafos planos asociados (ver Figura 4-17), ambos matemáticamente correctos.

Para detectar si un grafo tiene conectividad 2 no se ha utilizado ninguno de los métodos que ofrece la teoría de grafos [HT74], [CO 93] que aunque fiables son muy costosos de implementar. En el método expuesto en la siguiente sub-sección, y que ha sido aplicado en la aplicación REFER, se tienen como datos de partida una parte de las caras detectadas. Es decir, se calculan las caras siguiendo el método expuesto en la sección anterior, y a partir de los circuitos calculados (ya reducidos) se comienza la utilización de este método.

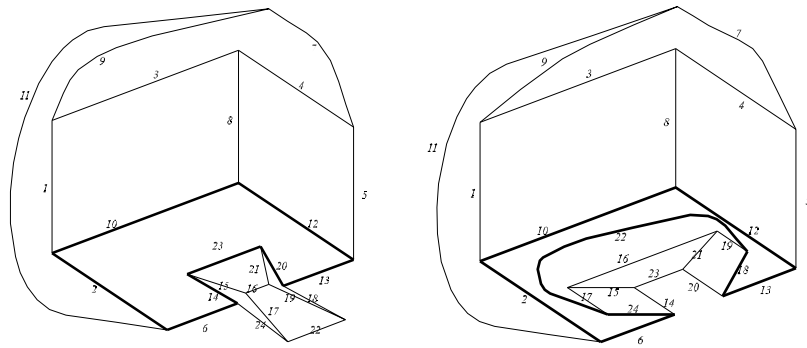


Figura 4-17. Dos posibles configuraciones planas de un modelo planar.

### 4.3.1. Detección de los vértices de corte

Utilizando el método de Courter y Brewer en grafos 2-conexos, se pueden obtener, indistintamente, las caras que perceptualmente tienen sentido, o aquellas que son topológicamente equivalentes pero carecen de sentido perceptual. Para detectar estas caras “cruzadas”, hemos desarrollado una estrategia basada en buscar:

*aquellas caras (o circuitos) que tienen en común más de un vértice perteneciente a aristas distintas.*

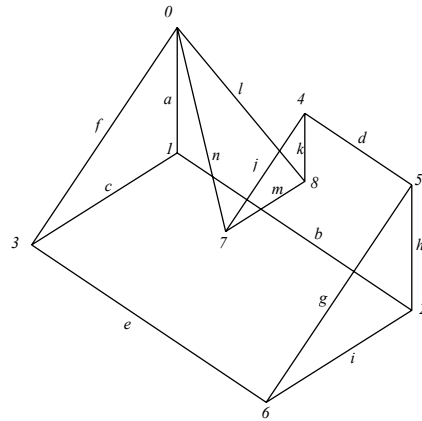
En efecto, son necesarios dos vértices pertenecientes a diferentes aristas porque cualesquiera dos caras adyacentes en el caso general poseen una arista en común (dos vértices), y para que fuese una cara falsa de un grafo 2-conexo las dos caras adyacentes necesitarían tener en común al menos otro vértice. Ese otro vértice puede pertenecer a otra arista compartida por las dos caras (que es el caso más general), o no. Las dos caras a las que perteneciesen de forma común todos estos vértices implicados son las caras sujetas a posible confusión y por ello sujetas a estudio. Así por ejemplo, en la Figura 4-18, las caras

4 y 6, calculadas antes de realizar la comprobación de cruces, tienen en común la arista  $d$  y el vértice  $0$ , de forma que el conjunto de los vértices de corte está formado por el vértice  $0$  y uno de los dos vértices ( $4$  ó  $5$ ) de la arista  $d$ .

En resumen, la característica a encontrar es una pareja de caras que compartan dos vértices no pertenecientes a la misma arista. Para detectar todas las parejas conflictivas se establece el siguiente procedimiento:

- ❑ Se crea una matriz  $CV$  cuyas filas representan caras calculadas, y las columnas representan a los vértices del grafo. Sus elementos  $(i, j)$  toman el valor 1 cuando la cara  $i$  contiene al vértice  $j$  y 0 en caso contrario.
- ❑ Se crea una matriz  $R$  producto de la matriz  $CV$  por sí misma, de modo que el resultado ofrezca una matriz cuadrada cuyo tamaño corresponde con el número de caras, y sus elementos serán un número entero cuyo valor corresponda con el número de vértices que tienen en común dos caras. En el caso de los elementos de la diagonal principal el valor que se recogerá será el número de vértices que posee cada cara.
- ❑ Se crea una *Matriz Comprobación* resultado de la operación  $\text{Matriz } R \cdot 2 \times Z$ , siendo  $Z$  la matriz de interacciones descrita en el método de Courter y Brewer (sección §4.2.4).
- ❑ Se comprueba el valor de los elementos de la Matriz  $R$ , de modo que a excepción de los elementos que pertenecen a la diagonal, los que tengan un valor mayor a dos son las caras sujetas a estudio.
- ❑ Las caras cuyo resultado en la Matriz Comprobación tengan un valor no nulo, serán también caras a estudiar, siempre que el mismo elemento en la matriz  $R$  sea mayor que la unidad. Deben exigirse las dos condiciones simultáneamente, dado que dos caras pueden tener un único vértice en común y eso no implica necesariamente que sea un grafo 2-conexo.

Este procedimiento es el empleado para encontrar caras vinculadas a vértices de corte de grafos 2-conexos. Los vértices comunes a las dos caras detectadas como conflictivas son fácilmente identificables recorriendo las filas de la matriz  $CV$  y de entre ellos se obtiene un conjunto de vértices de corte.



<i>CARA</i>	<i>ARISTAS</i>
<i>Cara 0</i>	<i>a, c, f</i>
<i>Cara 1</i>	<i>b, i, e, c</i>
<i>Cara 2</i>	<i>g, h, i</i>
<i>Cara 3</i>	<i>j, k, m</i>
<i>Cara 4</i>	<i>a, n, j, d, h, b</i>
<i>Cara 5</i>	<i>n, l, m</i>
<i>Cara 6</i>	<i>d, g, e, f, l, k</i>

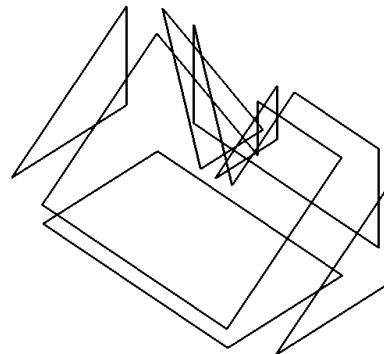


Figura 4-18. Caras calculadas antes de realizar la comprobación de cruces.

### 4.3.2. Elección de la posible combinación de caras

Una vez quedan detectadas las posibles caras cruzadas, los vértices de cada una de estas caras se clasifican en tres grupos: los vértices comunes a ambas caras, y las dos cadenas de vértices pertenecientes a cada una de las caras que quedarían desconexas si los vértices comunes fueran eliminados. De forma que las dos caras juntas forman cinco grupos de vértices: dos grupos por cada cara más los vértices que ambas tienen en común.



#### Capítulo 4. Detección de Caras de Poliedros en Representaciones 2D.

---

La combinación correcta de estos grupos es lo que nos dará como resultado las caras perceptualmente correctas.

Nombrando a los vértices de una cara con el índice 1 y a los de la otra cara con el índice 2 y a los diferentes grupos de vértices no comunes con las letras A y B, se tendrán cuatro grupos de vértices que serán denominados: 1 A, 1B, 2 A y 2B. En la Figura 4-18 las caras 4 y 6 son las caras sospechosas de ofrecer un cruce; los vértices que mantienen en común son los vértices 0, 4 y 5; y según la nomenclatura aquí establecida los cuatro grupos deberían ser:

cadena 1 A: vértices 1 y 2

cadena 2 A: vértices 3 y 6

cadena 1 B: vértice 7

cadena 2 B: vértice 8

En este punto, si alguna de las cadenas de vértices de alguna de las dos caras tiene un tamaño nulo, el problema es abandonado puesto que esto indica que el caso que se está tratando no es un problema de cruces, sino probablemente de caras falsas. Probablemente si uno de los cuatro grupos es nulo es porque una cara contiene a otra como es el caso de la Figura 4-19 y este tipo de problemas deben ser abordados desde otra perspectiva.

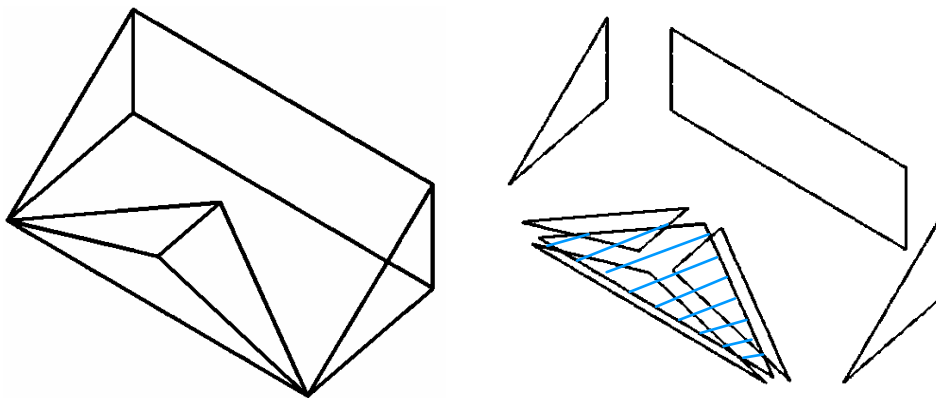


Figura 4-19. Grafo sobre el que se ha calculado una cara falsa.

La combinación de estos grupos de vértices con sus correspondientes aristas daría lugar a dos posibles soluciones “válidas” según la teoría de grafos, que tiene el grafo plano asociado a la figura. Para la elección de la combinación “perceptualmente correcta” de elementos que compongan las caras “verdaderas” se toma como criterio la cercanía entre

## Percepción Artificial de Dibujos Lineales

grupos de vértices. Este criterio se obtiene de la observación de diferentes ejemplos, cuyo resultado de la aplicación del algoritmo de detección de caras ofrecían caras cruzadas debido a su conectividad. En todos los casos para que no existan cruces, y la solución coincida con la que un observador humano elegiría, la combinación se realizaría tomando como criterio la proximidad de los grupos combinables. Es decir, emparejando uno de los tramos exteriores con el siguiente más cercano de los otros tres grupos. Para ello se calcula un vértice medio para cada uno de los grupos y luego se mide la distancia entre los mismos. Este vértice medio correspondería a la posición del centroide si se tratara de un polígono.

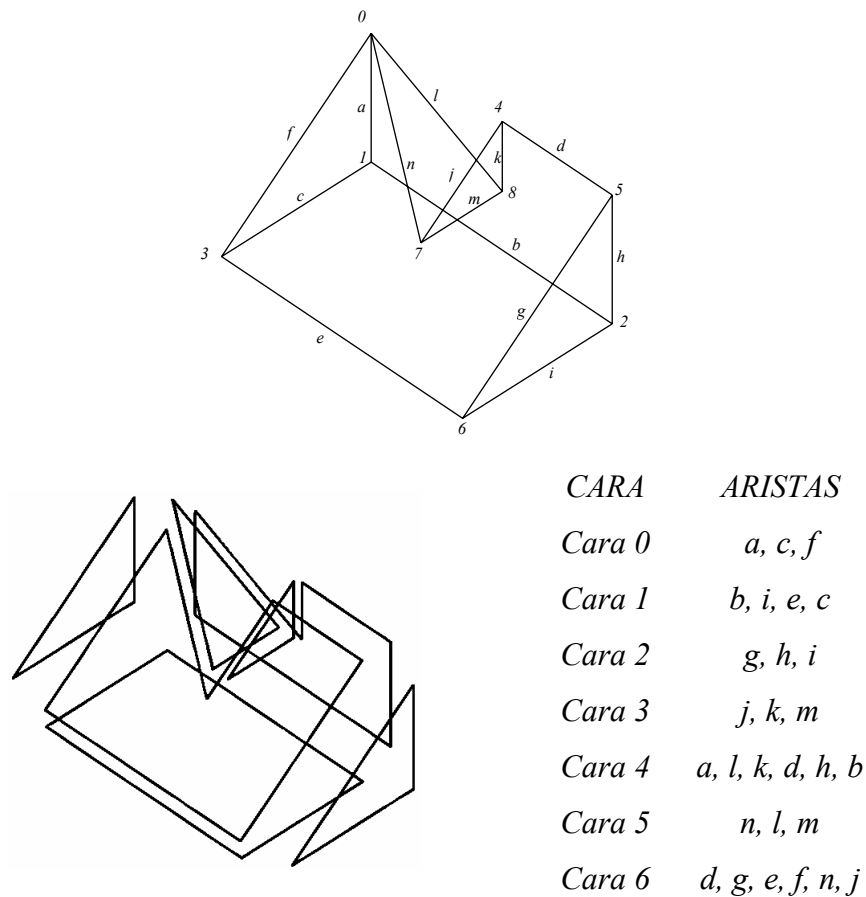


Figura 4-20. Caras detectadas después de la aplicación del algoritmo de resolución de caras cruzadas.

Se calculan seis distancias que se reducen a tres, que corresponden a los tres posibles emparejamientos:

$$\text{Distancia1} = \min (\text{Distancia1A1B}, \text{Distancia2A2B});$$

$$\text{Distancia2} = \min (\text{Distancia1A2A}, \text{Distancia1B2B});$$

$$\text{Distancia3} = \min (\text{Distancia1A2B}, \text{Distancia1B2A});$$

El valor de los tres que sea mínimo corresponderá al emparejamiento primeramente elegido. Las caras resultantes serán las aristas que tengan en sus extremos vértices de los dos grupos de vértices elegidos más los vértices comunes.

Para evitar que se hagan emparejamientos que no formen una cara, se debe comprobar que la elección conforme un circuito. Es decir, que no se acepten soluciones como 1B- 2B en el ejemplo de la Figura 4-18, que daría como resultado la “cara” formada por las aristas  $l, n, m, i, k, d$ . En ese caso se pasaría a elegir la siguiente combinación cuya distancia sea mínima.

### 4.3.3. Resultados

Los resultados han sido satisfactorios en la totalidad de los casos probados. Los tiempos de cálculo son notablemente inferiores al método propuesto por Lipson y Shpitalni, e implementado por Conesa en la versión anterior de REFER [Con01]. Además, todos los ejemplos correspondientes a grafos 2-variedad han sido resueltos correctamente, dado que las soluciones que los usuarios perciben son las que la aplicación obtiene. Algunos ejemplos de ello se muestran en la Figura 4-22 y Figura 4-23.

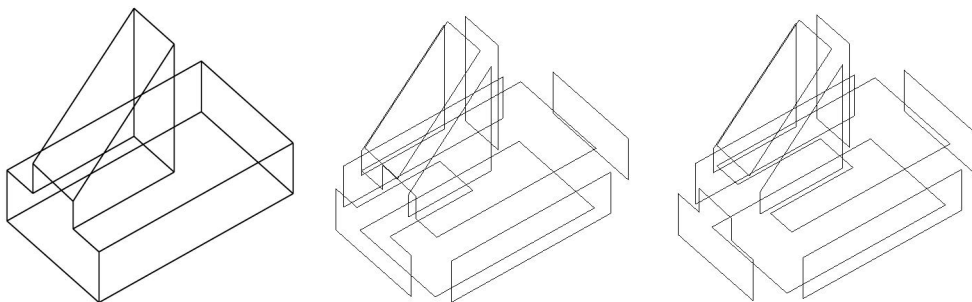


Figura 4-21. Cálculo de Caras antes y después de la aplicación del método de resolución de cruces.

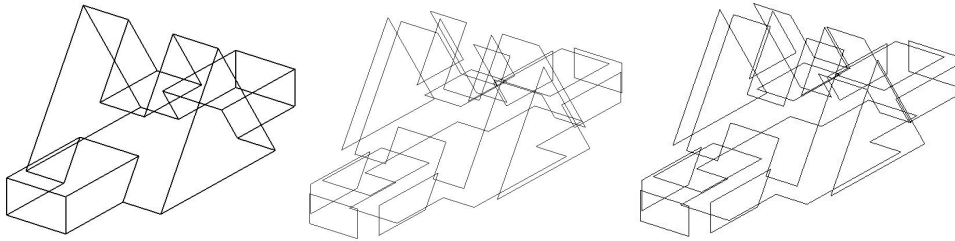


Figura 4-22. Ejemplo con 16 caras calculadas en menos de 1 segundo y con 4 posibles parejas de caras cruzadas.

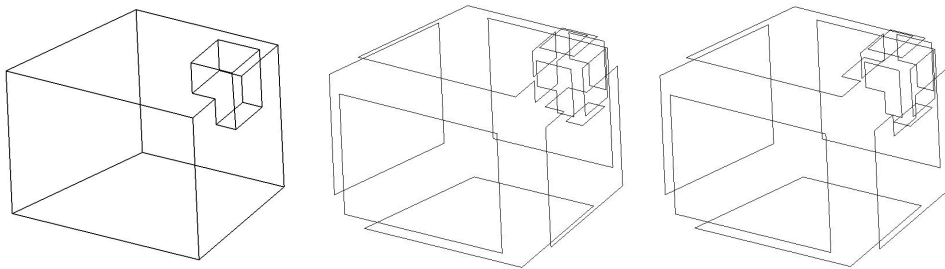


Figura 4-23. Otro caso donde se requiere la aplicación del método de resolución de cruces.

### 4.3.4. Limitaciones

El método empleado para el cálculo de caras en poliedros es rápido y eficiente. Se han realizado nuevas aportaciones para resolver algunas de las limitaciones que el mismo poseía y ha ofrecido resultados satisfactorios en todos aquellos poliedros equivalentes a grafos planares. Se han realizado nuevas aportaciones para resolver algunas de las limitaciones que el mismo poseía. Pero, sin embargo, sigue sin poder resolver el cálculo de caras falsas en grafos no planares como el correspondiente al poliedro representado en el ejemplo de la Figura 4-24.

Otra limitación importante en este método, y que también lo es para cualquiera de los algoritmos estudiados en esta tesis, es el tratamiento que se le ofrece a las caras de aquellos objetos que están compuestos por agujeros o unión de bloques cuando estos

conforman como resultado grafos no conexos. Un ejemplo se muestra en la Figura 4-25,

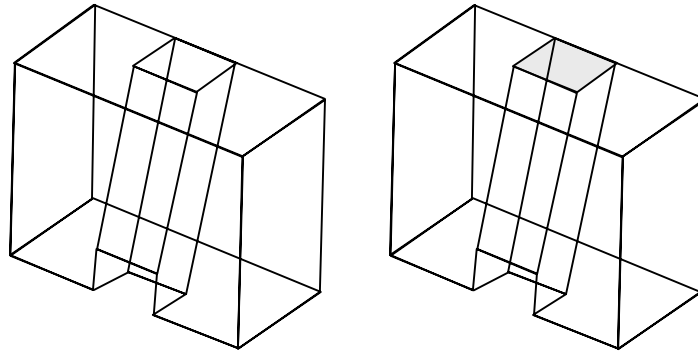


Figura 4-24. Modelo con posible cara falsa.

donde se observa como el algoritmo es incapaz de detectar correctamente las caras con agujeros (basta mirar la boca inferior del agujero), a menos que el usuario bocete las aristas auxiliares necesarias para “conectar” el agujero (tal como se muestra con las cuatro líneas diagonales dibujadas en la boca superior del agujero). En este caso, consideraría el agujero ciego y la base del mismo, una cara más del poliedro.

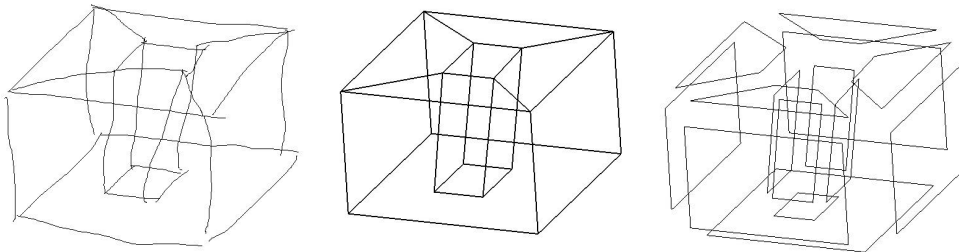


Figura 4-25. Modelo con agujero posiblemente pasante.

Aunque no se ha aplicado en el trabajo realizado en la tesis, queda como un trabajo futuro reconocer líneas curvas y con ello la reconstrucción de las superficies a las cuales pertenecen. Sin embargo, el método utilizado para la detección de caras es totalmente aplicable a modelos que posean superficies curvas.



# Capítulo 5. Simetría Facial Oblicua

---

## 5.1. INTRODUCCIÓN

---

La simetría es una propiedad deseable en la mayoría de los productos industriales, porque abarata los costes de producción e incrementa la utilidad de muchas piezas y con ello la versatilidad de los productos. Además, muchos objetos construidos por el hombre son simétricos por la facilidad de interpretación y de fabricación que esto supone. Algunos estudios [LMM01] revelan que la media de objetos simétricos es muy elevada en componentes y elementos industriales.

La simetría también juega un papel importante en el mundo y en la forma en la que los humanos perciben el mundo. Según decía Hoffman [Hof00], la *sencillez* y la *simetría*, constituyen las reglas fundamentales que rigen la percepción visual humana. En otras palabras, la *simetría* es una regla fundamental de la percepción visual humana [Pal99], e interviene en el proceso por el cual los humanos reconocemos los objetos [BET95]. Por ello, parece lógico pensar que con el objetivo de encontrar la reconstrucción de modelos tridimensionales partiendo de dibujos lineales, la simetría sea una de las características perceptivas más importantes a identificar.

Ulupinar [UN93] llegó incluso a decir en sus trabajos que la simetría da más información (o la misma) que el contorno de un objeto. Sin embargo, al igual que ocurre con otras reglas perceptuales fundamentales, es difícil formular la simetría en términos algorítmicos, y para descubrirla hay que desglosarla en *señales perceptivas* más simples y más fáciles de detectar. En la simetría de objetos poliédricos, la señal perceptiva más importante es la

*simetría facial oblicua* [Kan81], que se formula aprovechando la transformación afín que sufren las caras simétricas de los poliedros simétricos al aplicarles una proyección paralela.

Esta propiedad es, por tanto, altamente importante en la percepción humana y también en la reconstrucción de modelos tridimensionales a partir de dibujos. Dentro del campo de Visión por Computador también cabe destacar la importancia que se le ha dado a la simetría en estudios de ingeniería inversa [MLM01], en los que se deben tomar datos de modelos tridimensionales mediante escáneres tridimensionales para su posterior modelado en aplicaciones informáticas de modelado tridimensional.

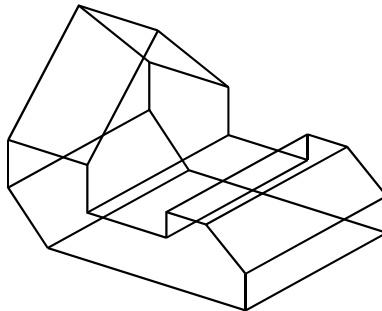


Figura 5-1. Representación de un poliedro con caras transparentes

La detección de la simetría *facial oblicua* requiere de la determinación previa de las caras en la imagen bidimensional. Pero a la resolución de este problema ya se ha dedicado el **Capítulo 4**.

### 5.1.1. Simetría

La palabra simetría en su primera acepción significa:

*“bien proporcionado, con equilibrio de formas, y la simetría se refiere a esa especie de concordancia de las partes que componen un todo”*

Pero como el propio Weyl [Wey51] apuntó ya en el año 1951, simetría tiene una segunda acepción estrictamente geométrica, cuya definición se basa en un concepto absolutamente preciso: la *Simetría bilateral*.

*“Un cuerpo, una configuración espacial, es simétrico con respecto a un plano dado  $E$ , si se transforma en sí mismo al reflejarse en  $E$ ”*



Según esta definición, la simetría se considera como una transformación del espacio en sí mismo, que lleva a un punto  $p$  arbitrario a su imagen especular  $p'$  con respecto a  $E$ . Tal y como define Palmer [Pal99], la simetría es una propiedad global de las figuras con respecto a una transformación dada, de modo que al aplicar dicha transformación la figura queda sin cambios.

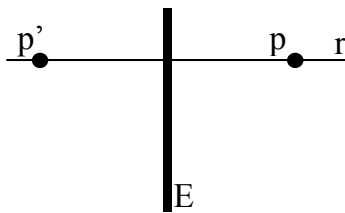


Figura 5-2. Reflexión en  $E$  de un punto  $p$ .

En otras palabras, cuando se habla de una figura simétrica en 2D, habitualmente se viene a hablar de aquellas figuras sobre las que se puede detectar un eje de forma que a ambos lados del mismo la figura es la misma. En este caso la simetría se llama *reflectiva* o *especular*. Aunque también se encuentra simetría cuando la figura se puede dividir en partes que pueden ser permutadas y se repiten a sí mismas giradas un ángulo constante. A esta operación de simetría se le llama de *rotación* o *simetría polar*.

De las posibles simetrías que podemos encontrar en el universo, la *simetría especular* de una figura plana, es la simetría más directa y la más relevante para este trabajo (Figura 5-2). Podemos decir que:

*la simetría especular está determinada por un eje (eje de simetría) coplanario con la figura que la divide en dos partes tales que a cada punto de una de las dos partes le corresponde un punto de la otra parte y se cumple que:*

- a) *ambos puntos equidistan del eje*
- b) *y determinan una recta perpendicular al mismo.*

### 5.1.2. Simetría Oblicua.

En el proceso de reconstrucción, objeto de esta tesis, las figuras de partida son proyecciones axonométricas de objetos tridimensionales, o bocetos que se asemejan mucho a este tipo de proyecciones. En el caso general, la forma de cualquiera de las caras

## Percepción Artificial de Dibujos Lineales

---

del modelo 3D tras la proyección, ya sea oblicua u ortogonal al plano de proyección, queda deformada, de modo que las caras del modelo se dice que presentan *oblicuidad*. Este hecho hace que la posible simetría asociada a las caras que presentan oblicuidad resulte deformada con ellas. La deformación en este caso hace referencia a una transformación, que hace que aunque los ejes de simetría continúen dividiendo a la figura en dos partes, la correspondencia entre sus vértices no se establezca mediante una arista perpendicular al eje de simetría, sino en una dirección no ortogonal, que se denomina *conjugada*. La justificación de la dirección conjugada proviene de la geometría proyectiva, dado que se considera que es la proyección de la dirección ortogonal de simetría original. A esta simetría cuya dirección de correspondencia entre puntos no es ortogonal se la ha denominado *Simetría Oblicua*.

El concepto de simetría oblicua permite que tras realizar una detección previa de caras, sea posible estudiar la *Simetría Facial Oblicua* en el propio plano del dibujo, es decir, sin necesidad de identificar previamente el modelo tridimensional del cual se supone que la figura dada es proyección. En la Figura 5-4 se ofrece la representación de una cara que podría ser una proyección arbitraria de la Figura 5-3. En ella se puede observar el eje  $E_1$ , que es el que llamaremos *eje de simetría oblicua*, y  $E_2$  es su *dirección conjugada* o *eje de oblicuidad*.

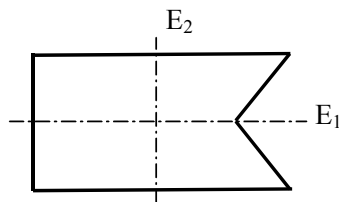


Figura 5-3. Figura plana con simetría especular.

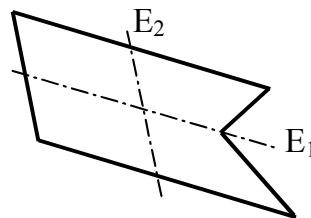


Figura 5-4. Eje de simetría oblicua y su conjugado de la proyección de una cara simple.

Desde otro punto de vista se podría considerar que si los puntos de una figura simétrica se representan con coordenadas oblicuas, se obtiene una figura con simetría oblicua. Y desde este planteamiento la simetría oblicua es más general que la simetría especular, puesto que no necesita tener sus ejes perpendiculares entre sí.

## 5.2. ANTECEDENTES

---

Desde hace algún tiempo se ha ido avanzando en los algoritmos que detectan simetrías, habiendo sido estudiada desde muchos puntos de vista. Así por ejemplo, Alt [AMW88] y Highnam [Hig86] basaron sus estudios en algoritmos que tenían como entrada un conjunto finito de puntos, y para comprobar si poseían simetría convertían los puntos en una cadena de caracteres y entonces simplemente comprobaban si dicha cadena era un palíndromo. También se han encontrado algoritmos que parten de figuras planas formadas por un conjunto de elementos, todos ellos incluidos dentro de una lista de patrones geométricos [Ata85]. Con otro enfoque, Robinson, buscó el centro de simetría de un conjunto de puntos [RK88]. Este trabajo previo le sirvió de base para un método posterior de búsqueda de ejes de simetría [Rob96].

Marola [Mar89] puso un mayor énfasis en la detección de ejes de simetría en figuras planas obtenidas a partir de imágenes digitalizadas, representadas mediante niveles de grises, que presentaban ruido debido principalmente a la obtención de los contornos de los objetos representados en la imagen.

Otros estudios como [STV99] analizan cómo computar la simetría y la obtención de medidas del grado de simetría en polígonos.

Pero como ya se ha dicho, la entrada a nuestro proceso de reconstrucción son dibujos lineales arreglados durante la propia fase interactiva de bocetado, por lo que los problemas de ruidos e imperfecciones de la entrada típicos de las imágenes captadas en entornos reales le son ajenos. Además, los dibujos de los que disponemos para la entrada a nuestro proceso de reconstrucción son o se pueden asimilar al resultado de la proyección de un modelo poliédrico sobre un plano. Esto hace que las caras poligonales que componen el poliedro en la representación 2D muestren su contorno deformado con respecto a la forma que tendrían en el modelo original. En la misma medida una cara que fuera simétrica en el modelo 3D tiene esa simetría deformada en la proyección 2D. Por lo tanto si queremos encontrar simetrías en figuras proyectadas debemos estudiar otro tipo de transformaciones como es la *simetría oblicua*.

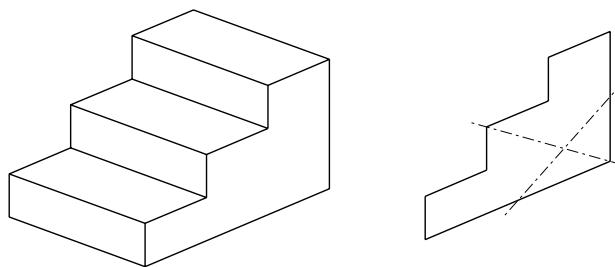
Kanade [Kan81] ya clasificó a la simetría oblicua como una regularidad y basándose en la observación dijo de ella que:

*“Una simetría oblicua representa una simetría real observada desde un punto de vista (desconocido)”*

## Percepción Artificial de Dibujos Lineales

---

Kanade en su estudio sobre esta regularidad, y siguiendo la línea del método de reconstrucción que en su época llevó a cabo, la aborda como una restricción que hay que aplicar dentro del espacio del gradiente. Demuestra que la posible simetría representa realmente una simetría si su gradiente se encuentra sobre una hipérbola, y además, minimiza la pendiente.



*Figura 5-5. Modelo 2D con una de sus caras con simetría facial oblicua destacada.*

Los pasos de Kanade y la búsqueda de la simetría facial oblicua desde el espacio del gradiente fueron seguidos por otros como [GM95].

Pero fue Friedberg [Fri86] quien abordó el tema en amplitud, y dio una formulación clara para la búsqueda de los posibles ejes de simetría oblicua y su posterior comprobación. Para ello formuló cómo deshacer la función oblicuidad con los supuestos ejes, de modo que dependiendo de la figura obtenida como resultado acepta o rechaza al posible eje de simetría oblicua como eje de simetría de la superficie plana. Así en su artículo presenta una condición necesaria, aunque no suficiente, para que un eje sea considerado eje de simetría oblicua.

Otro método fue estudiado por Oh [OAT88], cuyo objetivo es encontrar los ejes de simetría oblicua en polígonos para determinar la orientación de la superficie. Oh presenta un método sencillo pero que consume una gran cantidad de tiempo.

Gross, en [GB 91], vino a clasificar los métodos en globales y parciales. Su método local comparaba pares de puntos del contorno de la figura 2D y demostró que el método local tenía la ventaja respecto al global de no necesitar ninguna propiedad general del contorno. Debido a ello el método tiene una mayor probabilidad de no fallar cuando se trabaja con contornos parcialmente ocultos. El inconveniente principal del método de Gross [GB 94], es

que éste parte sabiendo a priori el número de ejes de simetría oblicua que posee el contorno y, además, ese número no puede ser mayor a cinco ejes.

Yip [YTL94], por su parte, utilizó la elipse de Fourier y la parametrización de la longitud del arco para encontrar los ejes de simetría oblicua y reconstruir la simetría dada una proyección cilíndrica en polígonos.

Alguno autores han basado sus estudios en la obtención de ejes de figuras con simetría oblicua que incluyen curvas en su contorno [Pos92]. En la mayoría de casos son figuras obtenidas de imágenes reales de las cuales no se pueden apreciar sus partes ocultas, de modo que para la obtención de la figura plana de partida se ha tenido que realizar un reconocimiento previo de contornos, lo que implica una fuente mayor de ruidos en la figura, y unos datos de partida bastante diferentes [GB91], [ST94], [CC 94], [CC 95], [SIT01], [SK01]. Mientras, otros [ZW94] toman las medidas de objetos reales sobre los que ya se saben a priori ciertas características del objeto.

Wen [WY 95] continuó los pasos de Friedberg en gran medida, y mostró un método sencillo para encontrar los puntos críticos de los contornos por los que posiblemente pasarán los ejes, y por los que, en consecuencia, se debía realizar la comprobación de simetría.

Por su parte Ulupinar y Nevatia [UN95] utilizan la simetría como una propiedad muy importante para reconstruir modelos 3D a partir de contornos, como ya habían hecho en trabajos anteriores [UN93]. Aunque su estudio en [UN95] se centra en el estudio de superficies generadas a partir de superficies de revolución.

En la tesis de Lipson [Lip98], y en su artículo [LS96], se presentó una formulación muy reducida, que no garantizaba una búsqueda exhaustiva de todos los ejes de simetría, porque no pretendía encontrar el número exacto de ejes de simetría oblicua o su posición exacta. Por el contrario, sus estudios se centran más en cómo se podían computar en la función objetivo de la reconstrucción mediante optimización la posibilidad de que las caras planas del objeto a reconstruir tuvieran un posible eje de simetría. Un estudio más detallado de este método se presentará en la siguiente sección (§5.3).

## 5.3. MÉTODOS ESTUDIADOS DE DETECCIÓN DE SIMETRÍA OBLICUA EN RECONSTRUCCIÓN

---

En el campo de la reconstrucción tridimensional hay dos estudios realmente relevantes que le hayan dado importancia a la simetría facial oblicua. El primer de ellos es el de Kanade [Kan81] y el segundo el trabajo de Lipson y Shpitalni [LS 96].

Los estudios de Kanade han sido utilizados en métodos de reconstrucción tridimensional por Grimstead [Gri97] y Varley [VM00c]. Estos no variaron el método de detección de simetría oblicua; simplemente utilizaron sus conocimientos en sus métodos propios de reconstrucción e incluyeron parte de los trabajos de Friedberg [Fri86].

### 5.3.1. Kanade

Kanade [Kan81] combina el espacio del gradiente, el etiquetado de líneas y la simetría oblicua para derivar el gradiente de las caras en una proyección axonométrica isométrica.

Para Kanade una simetría oblicua definida por  $\alpha$  (dirección del eje de simetría oblicua) y  $\beta$  (dirección conjugada) puede corresponder a una simetría real si y sólo si el gradiente de la superficie se encuentra sobre una hipérbola (Figura 5-6).

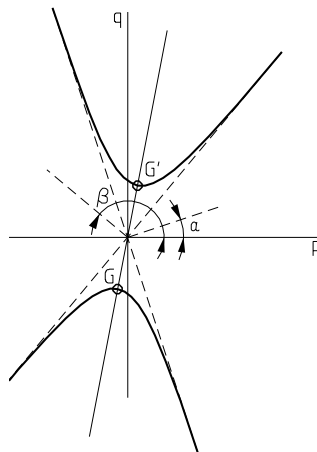


Figura 5-6. Hipérbola utilizada por Kanade

De los infinitos puntos, de coordenadas  $(p, q)$  en el espacio del gradiente, que se encuentran sobre la hipérbola, Kanade elige como solución los dos puntos más cercanos al origen, marcados como  $G$  y  $G'$  en la Figura 5-6. Estos puntos representan el gradiente de la cara simétrica correspondiente a la orientación con menor pendiente que puede producir una simetría facial oblicua partiendo de una simetría real en la cara. Ésta es una solución particular de entre las posibles, pero también es cierto que será la más sencilla, puesto que cuanto menos inclinación tenga la superficie respecto al plano de proyección, menor diferencia de longitudes habrá entre la figura plana real y la proyectada.

### 5.3.2. Lipson y Shpitalni

Lipson y Shpitalni proponen un modo simplificado en su artículo [LS96] (y en la tesis de Lipson [Lip98]) para computar la simetría oblicua en su función objetivo. Su técnica no tiene por objeto la determinación rigurosa de todos los ejes y de su posición exacta. Lo que pretenden es disponer de una regularidad de simetría que corrija la falta de simetría de los modelos que resultan al reconstruir utilizando otras regularidades (como la planicidad, la ortogonalidad, etc) que no garantizan ningún tipo de simetría. Por tanto, buscan una regularidad que sea fácil de calcular y que haga depender a la función objetivo *también* de la simetría.

El método que utilizan consta fundamentalmente de dos partes. La primera fase se basa en la elección de los posibles ejes de simetría oblicua en la figura proyectada. Para ello evalúa dos condiciones. Una de ellas es lo que vienen a llamar *simetría*, que no es más que un ratio entre las distancias de los vértices de la cara al supuesto eje. La relación la expresa como sigue:

$$sym_i = \frac{dist\_de\_v'_{k+i}\_al\_eje}{dist\_de\_v'_{k-i}\_al\_eje} - 1$$

Ecuación 5-1

siendo el supuesto eje de simetría oblicua a estudio el que pasa por el vértice  $v_k$  y  $v_{k+n/2}$ , siendo  $n$  el número de vértices que posee la cara. La nomenclatura utilizada denota  $v'_k$  a los vértices de la imagen 2D (antes de comenzar el proceso de reconstrucción), mientras que los  $v_k$  hacen referencia a los vértices en el espacio tridimensional.

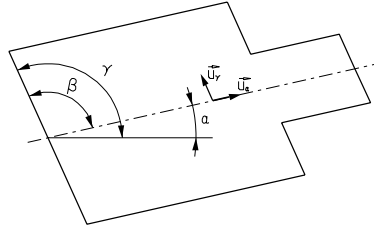


Figura 5-7. Contorno con simetría facial oblicua en la que se representan los vectores unitarios de las direcciones del eje de simetría oblicua y de su conjugado.

Debido a que las caras están delimitadas por aristas rectas, los posibles ejes de simetría deberán ser un vértice o un punto medio de arista. Es por ello que los valores que puede tomar  $k$ , son  $1/2, 1, 3/2, 2, \dots, n/2$ .

La Ecuación 5-1 se aplicará a todos los vértices. Es decir, se extiende para todo  $i$  que pueden tomar los mismos valores que  $k$ . Por otra parte, la segunda condición evaluada viene a comprobar la oblicuidad. Para ello emplea la Ecuación 5-2:

$$skew_i = \left[ \frac{(v'_k - v'_{k+n/2}) \cdot (v'_{k+i} - v'_{k-i})}{|v'_k - v'_{k+n/2}| \cdot |v'_{k+i} - v'_{k-i}|} \right] \cdot \left[ \frac{(v'_k - v'_{k+n/2}) \times (v'_{k+i} - v'_{k-i})}{|v'_k - v'_{k+n/2}| \cdot |v'_{k+i} - v'_{k-i}|} \right]$$

Ecuación 5-2

Con esta ecuación, a mi entender, lo que se pretende es medir el ángulo que forma el supuesto eje de simetría con la dirección de oblicuidad (la dirección transversal en la simetría oblicua). Esto mismo lo he interpretado como:

$$skew_i = (\vec{u}'_\alpha \cdot \vec{u}'_\gamma) \cdot (\vec{u}'_\alpha \times \vec{u}'_\gamma) = \cos \beta \cdot \text{sen} \beta k$$

Ecuación 5-3

Siendo  $u'_\alpha$  y  $u'_\gamma$  los vectores unitarios de las direcciones del eje de simetría oblicua y de su conjugado.

Es decir, para comprobar la oblicuidad calculan los vectores unitarios del eje y del eje transversal y los multiplican escalar y vectorialmente. Eso implica que el primer término sea el coseno de  $\beta$  y el segundo el seno del mismo ángulo pero con dirección perpendicular al plano del dibujo. Los autores justifican la utilización de esta formulación mediante el



comportamiento estadístico que los valores alternados resultado de multiplicar el producto escalar por el producto vectorial producen.

De esta forma se calculan los valores de las funciones *skew* y *sym* y para cada uno de los posibles ejes evaluados ( $n/2$ ) de cada cara. A continuación con estos valores se calcula la desviación típica de los mismos, teniendo como datos los valores proporcionados por la Ecuación 5-1 y la Ecuación 5-2 para cada eje de la cara de la cual se quiere obtener el eje de simetría oblicua.

Se calcula un coeficiente de ponderación  $w_k$  para cada posible eje  $k$ . Para ello Lipson y Shpitalni aplican la función  $\mu$  a la suma de las dos desviaciones estándar.

$$w_k = \mu_{0,0.2} \left( \begin{matrix} \sigma(skew_i) & + & \sigma(sym_i) \\ i=1/2,2/2,3/2,\dots,n/2 & & i=1/2,2/2,3/2,\dots,n/2 \end{matrix} \right)$$

Ecuación 5-4

La función  $\mu$  es muy recurrida por los autores en casi todas las regularidades que aplican en el proceso de reconstrucción. Se trata de una función exponencial descendente (Ecuación 5-5) cuyo objetivo es descartar las regularidades que inicialmente se cumplen poco. Los valores concretos de los parámetros  $a$  y  $b$  que adoptan Lipson y Shpitalni para determinar el coeficiente de ponderación de la regularidad de simetría son  $a=0$  y  $b=0.2$ .

$$\mu_{a,b}(x) = \max \left[ 0, 1.1 \cdot e^{-\left(\frac{x-a}{b}\right)^2} - 0.1 \right]$$

Ecuación 5-5

De todos los posibles ejes evaluados, se elegirá como eje de simetría oblicua de la cara el que ofrezca un mayor valor en la Ecuación 5-4. Además, ese valor será el coeficiente de ponderación que se aplicará en la función objetivo de la reconstrucción para comprobar la regularidad de simetría facial oblicua en esa cara ( $w_k$ ).

$$w_{\substack{\text{simetría} \\ \text{oblicua}}} = \max_{k=1/2,2/2,3/2,\dots,n/2} [w_k]$$

Ecuación 5-6

Así, una vez elegido el posible eje de simetría facial oblicua la función elegida para calcular el coste de la regularidad es la siguiente:

$$\alpha_{\substack{\text{simetría} \\ \text{oblicua}}} = w_k \sum_{i=1}^{n/2} \left[ \sin^{-1} \left( \left[ \frac{(v_{k+i} - v_{k-i})}{\|v_{k+i} - v_{k-i}\|} \right] \times \left[ \frac{(v_k - v_{k+n/2})}{\|v_k - v_{k+n/2}\|} \right] \right) \right]^2$$

Ecuación 5-7

Esto es equivalente a siendo  $u_\alpha$  y  $u_\gamma$  los vectores unitarios de eje de simetría elegido y su conjugado en tres dimensiones:

$$\alpha_{\substack{\text{simetría} \\ \text{oblicua}}} = w_k \sum_{i=1}^{n/2} \left[ \sin^{-1} (\bar{u}_\gamma \times \bar{u}_\alpha) \right]^2 = w_k \cdot \sum_{i=1}^{n/2} \beta^2$$

Ecuación 5-8

Para cada cara la Ecuación 5-7 ofrece el coste de la regularidad de simetría facial oblicua de esa cara dentro del compute de la función objetivo en la reconstrucción.

En definitiva, el objetivo principal del método es determinar *algún* eje de simetría oblicua con la condición de que tenga una alta probabilidad de corresponder a un eje de simetría del modelo a reconstruir. La limitación que supone no intentar una búsqueda exhaustiva, unida a la naturaleza altamente heurística del método, hace que no sea apropiado para el objetivo de la presente tesis.

## 5.4. MÉTODO DE DETECCIÓN DE EJES DE SIMETRÍA OBLICUA UTILIZADO

---

La oblicuidad se puede ver como una transformación afín. Por lo tanto se puede encontrar una función  $T$  que aplicada a todos los puntos del contorno de la cara real, nos proporcione como resultado la figura oblicua de ésta (Figura 5-8).

Así a través de una transformación afín podemos:

- transformar un contorno con simetría ortogonal en un contorno con simetría oblicua, o

- transformar un contorno con simetría oblicua en un contorno con simetría ortogonal,

siendo una transformación la inversa de la otra.

Llamaremos  $P = [x, y]$  a las coordenadas del punto cuando está proyectado y, por tanto, oblicuo y,  $Q = [u, v]$  a las coordenadas del punto simétrico, sin oblicuidad.  $T$  es a la función de oblicuidad:  $PT = Q$ . Si conseguimos encontrar esa función  $T$ , también seremos capaces de calcular su inversa  $T^{-1}$ .

$$Q = T^{-1} P$$

Ecuación 5-9

Si aplicamos la función  $T^{-1}$  al contorno proyectado obtendremos la imagen sin oblicuidad, y así se podrá comprobar si esa figura posee realmente simetría ortogonal. Es por ello que a continuación se detalla la formulación para encontrar esa función.

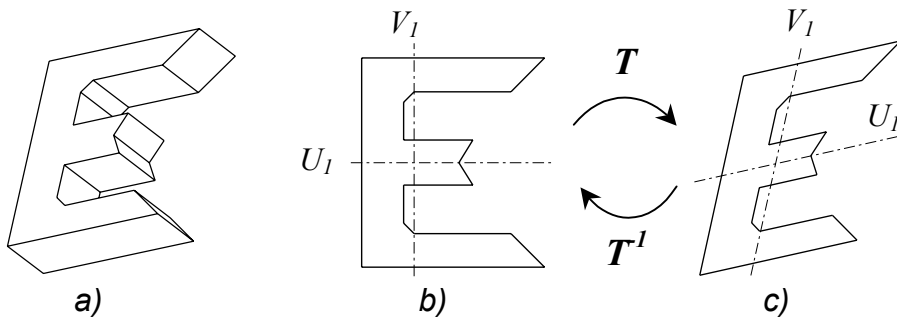


Figura 5-8. a) Objeto poliédrico b) Una cara simétrica con eje de simetría y eje conjugado c) Simetría oblicua obtenida por proyección cilíndrica de la cara

### 5.4.1. Formulación de la función inversa a la oblicuidad

Si un contorno cerrado tiene más de un eje de simetría oblicua, todos ellos pasarán por el centro del contorno [WY95]. Así, asumiendo que cada cara está compuesta por un material de densidad uniforme, cualquier eje de simetría de la misma pasará por su *centroide*, también llamado centro de gravedad o *centro de masas*. Además, considerando la oblicuidad como una transformación afín, es fácil comprobar que el centroide es un

## Percepción Artificial de Dibujos Lineales

invariante de esta transformación. Esto hace pensar que es aconsejable situar el origen de coordenadas en el centroide.

El centroide para una forma poligonal (un número discreto de vértices) se calcula como:

$$c_x = \frac{1}{6A} \sum_{i=0}^{N-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i)$$

$$c_y = \frac{1}{6A} \sum_{i=0}^{N-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i)$$

Ecuación 5-10. Centroide de una forma poligonal

siendo  $A$  el área de la figura:

$$A = \frac{1}{2} \sum_{i=0}^{N-1} (x_i y_{i+1} - x_{i+1} y_i)$$

Ecuación 5-11. Área de un polígono.

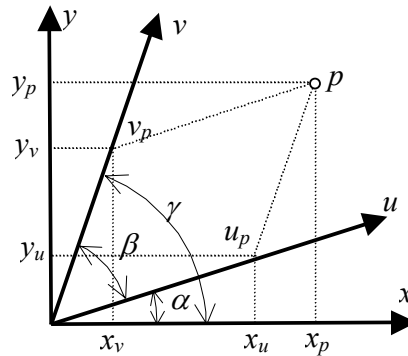


Figura 5-9. Coordenadas ortogonales y oblicuas de un punto  $p$ .

En la Figura 5-9 definimos un sistema de referencia ortogonal  $(X, Y)$  para cada cara del poliedro con origen su centro de masas, tal que  $(x_p, y_p)$  serán las coordenadas de un punto  $P$  del polígono que determina el contorno de la cara. También definimos un sistema de ejes de simetría oblicua con coordenadas  $(u, v)$ , referidas al eje de simetría  $(U)$  y al eje conjugado o eje de oblicuidad  $(V)$ . Si denotamos  $\alpha$  al ángulo de rotación y  $\gamma$  al ángulo entre

el eje de oblicuidad y el eje  $x$ , la oblicuidad viene dada por  $\beta = \gamma - \alpha$ , siendo  $\alpha$  y  $\gamma$  pertenecientes al intervalo  $[0, \pi]$ .

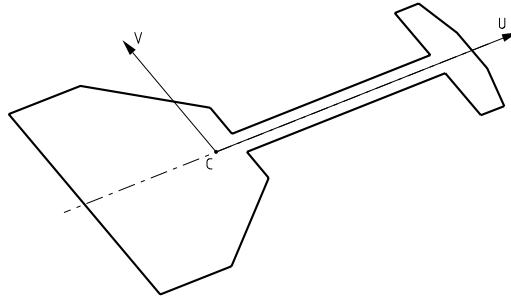


Figura 5-10. Ejes oblicuos en un contorno con simetría oblicua.

Asumiendo que la figura, de la que se quiere obtener los ejes de simetría oblicua, deriva de una figura plana y teniendo en cuenta (siguiendo el artículo de Friedberg [Fri86]) que la matriz estándar de Momentos de Inercia ( $m_{ab} = \sum x_k^a y_k^b$ ) debe ser diagonal,

obtenemos que  $\sum_{i=0}^{N-1} x_i y_i$  tiene que ser igual a cero. Y de aquí Friedberg deduce la siguiente

formulación de lo que él vino a llamar Condición Fundamental de Simetría. Siendo  $n_{ab}$  los elementos de la matriz de momentos  $N = \sum q_i^T q_i$  de inercia correspondientes a la figura oblicua, y siendo  $q_i$  las coordenadas 2D de los vértices oblicuos:

$$\alpha = \arctan \left[ \frac{n_{11} \cdot \tan \gamma - n_{02}}{n_{20} \cdot \tan \gamma - n_{11}} \right] = \arctan \left[ \frac{\sum x_i y_i \tan \gamma - \sum y_i^2}{\sum x_i^2 \cdot \tan \gamma - \sum x_i y_i} \right]$$

$$\gamma = \arctan \left[ \frac{n_{02} \cdot \cot \alpha - n_{11}}{n_{11} \cdot \cot \alpha - n_{20}} \right] = \arctan \left[ \frac{\sum_{i=0}^{N-1} \frac{y_i^2}{\tan \alpha} - \sum_{i=0}^{N-1} x_i y_i}{\sum_{i=0}^{N-1} \frac{x_i y_i}{\tan \alpha} - \sum_{i=0}^{N-1} x_i^2} \right] = \arctan \left[ \frac{\sum_{i=0}^{N-1} x_i y_i - \sum_{i=0}^{N-1} y_i^2}{\sum_{i=0}^{N-1} x_i^2 - \sum_{i=0}^{N-1} x_i y_i} \right]$$

Ecuación 5-12. Condición Fundamental de Simetría

Una vez obtenida la condición fundamental de simetría, podemos deducir la función de oblicuidad  $T$  como:

$$T = \begin{bmatrix} \cos \alpha & \sin \alpha \\ \cos \alpha \cot \beta - \sin \alpha & \sin \alpha \cot \beta + \cos \alpha \end{bmatrix}$$

Ecuación 5-13. Función oblicuidad

Cuya inversa nos ofrecerá la función que realmente estábamos buscando, que es la función que deshace la oblicuidad:

$$T^{-1} = \begin{bmatrix} \sin \alpha \cdot \cot \beta + \cos \alpha & -\sin \alpha \\ \sin \alpha - \cos \alpha \cdot \cot \beta & \cos \alpha \end{bmatrix}$$

Ecuación 5-14. Función de des-oblicuidad

De esa forma, si  $x$  e  $y$  son las coordenadas desoblicuizadas del vértice que se quiere encontrar, y  $u$  y  $v$  son las coordenadas oblicuas, se deduce que:

$$x = \frac{-\tan \gamma \cdot u + v}{\sin \alpha - \cos \alpha \cdot \tan \gamma}$$
$$y = \frac{\tan \alpha \cdot u - v}{\cos \gamma \cdot \tan \alpha - \sin \gamma}$$

Ecuación 5-15. Coordenadas no-oblicuas en función de las coordenadas oblicuas.

### 5.4.2. Aplicación del método

Por tratarse de caras poligonales, los posibles ejes de simetría oblicua de las caras de los objetos para la reconstrucción, a los que hemos denominado *ejes tentativos*, como  $U_1$  y  $U_2$  de la Figura 5-11, son líneas que deben pasar por uno de los vértices del polígono ( $v_i$ ) o por uno de los puntos medios de sus aristas ( $m_i$ ), además de pasar por el centroide ( $c$ ), como ya se comentó anteriormente. Si el polígono es simétrico tendrá, como máximo, tantos ejes de simetría como vértices.

Las parejas de puntos formadas por el centroide y los sucesivos vértices y puntos medios de la mitad del contorno determinan todos los ejes tentativos (por ejemplo  $v_1$  y  $c$  para  $U_1$ ).

Por todo lo dicho, el método se puede resumir en el siguiente procedimiento iterativo:

- Se elige una de las caras previamente calculadas y se calcula su centroide mediante la Ecuación 5-10.
- Se elige un primer vértice (o punto medio)  $v_i$  como perteneciente al eje tentativo.
- Una vez elegido  $v_i$ , se conoce  $\alpha$  y a partir de la Ecuación 5-12, se calcula  $\gamma$ . Así ya tenemos definidos un posible eje de simetría y su conjugado.
- Con los valores de  $\alpha$  y  $\gamma$  ya se puede calcular la matriz de la función inversa a la oblicuidad  $T^{-1}$  dada en la Ecuación 5-14, que se aplicará a todos los vértices del contorno, y con ello se obtiene una cara supuestamente simétrica sin oblicuidad. Pero habrá que comprobar si el resultado de la transformación ofrece una figura simétrica.
- 

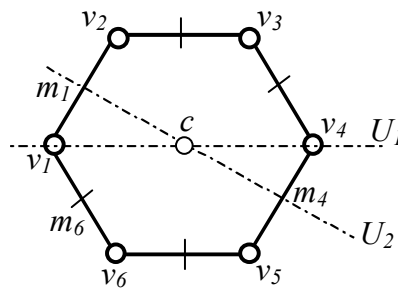


Figura 5-11. Ejes tentativos de simetría

Para comprobar si la cara resultante es simétrica se podría aplicar de nuevo la condición  $\sum_{i=0}^{N-1} x_i y_i = 0$ , pero esta condición es necesaria pero no suficiente para que el polígono sea simétrico. Sin embargo, tal y como se ha formulado la función  $T^{-1}$ , si la figura es simétrica después de la transformación, lo será respecto al eje  $X$ . Por lo tanto se debería cumplir que:

$$x_{k-i} = x_{k+i} \qquad \text{Ecuación 5-16. Condición de Simetría Total}$$

$$y_{k-i} + y_{k+i} = 0$$

## Percepción Artificial de Dibujos Lineales

---

siendo  $k$  la posición en la ordenación que ocupa el vértice o el punto medio por el que pasa el eje de simetría e  $i = [1, 2, \dots, n/2]$ .

Pero, hay que tener en cuenta que la figura de entrada al proceso de reconstrucción es una figura plana bidimensional, que puede estar delineada (con coordenada de vértices "perfectas") o bocetada (con coordenada de vértices "imperfectas"). En otras palabras, hay que recordar que el dibujo de partida, aunque debe contener todos los elementos del dibujo que quiere representar, no tiene porqué ser la proyección perfecta del objeto que representa. Así pues, dado que los dibujos con los que tratamos no tienen porqué ser perfectos, debemos establecer unos ciertos márgenes de aceptabilidad. En nuestro caso, en la aplicación REFER hemos incluido como criterio de aceptabilidad, un 10% del rango de los valores de las coordenadas del contorno:

$$|x_{k-i} - x_{k+i}| < 0.1 * |x_{max} - x_{min}|$$

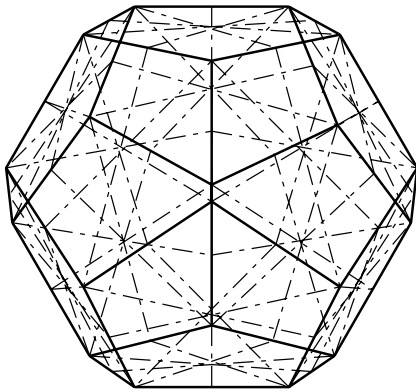
$$|y_{k-i} - y_{k+i}| < 0.1 * |y_{k-i} - y_{k+i}|$$

*Ecuación 5-17. Límites de Aceptabilidad de simetría.*

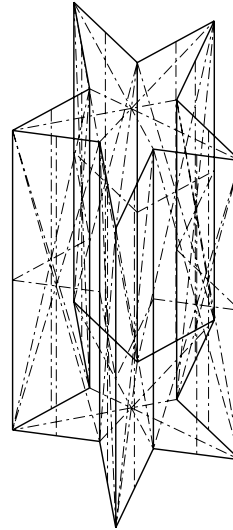
El proceso debe repetirse para todos los ejes tentativos de cada cara (es decir, para todas las parejas de puntos formadas por el centroide y los sucesivos vértices y puntos medios de la mitad del contorno), y para todas y cada una de las caras detectadas en el dibujo de entrada.

Como ejemplo del funcionamiento del algoritmo descrito, en la Figura 5-12 se muestran todos los posibles ejes de simetría de cada una de las caras calculadas en las representaciones planas de cuatro objetos poliédricos, calculados según el método aquí expuesto.

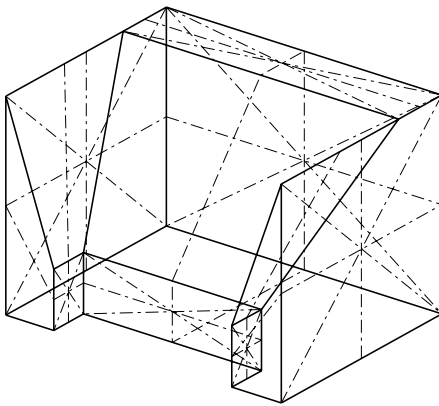




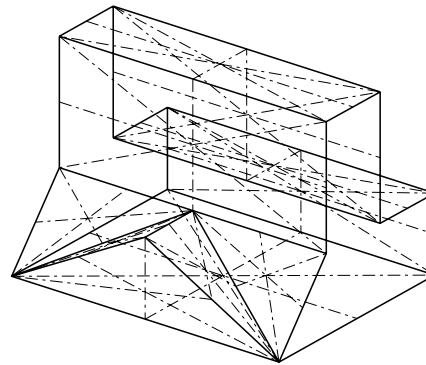
a) 30 aristas  
60 ejes de simetría facial oblicua



b) 30 aristas  
50 ejes de simetría facial oblicua



c) 29 aristas  
34 ejes de simetría facial oblicua



d) 27 aristas  
39 ejes de simetría facial oblicua

Figura 5-12. Modelos proyectados con todos sus ejes de simetría facial oblicua representados.

## 5.5. LA SIMETRÍA FACIAL OBLICUA COMO REGULARIDAD

---

Si el método que se quiere aplicar para la reconstrucción tridimensional de un grafo es la optimización de las regularidades detectadas en la imagen, habrá que estudiar la forma de computar la regularidad de simetría facial oblicua y su comportamiento en el proceso de reconstrucción. Es decir, cuando comienza el inflado hay que valorar cuán simétricas son las caras a las que previamente se les había detectado algún eje de simetría oblicua.

### 5.5.1. Cambio de Sistema de Referencia.

Una vez el inflado del modelo ha comenzado, las caras calculadas del mismo ya no se encuentran sobre un mismo plano 2D como cuando se realizó la detección de los ejes de simetría oblicua. Es por ello que para comprobar si una de las caras del modelo posee simetría es conveniente asociar un sistema de referencia a cada una de las caras, situando su centro en el centroide de la cara, que ahora también poseerá coordenada  $Z$  (Ver Figura 5-13). Es decir, debemos situar el origen de coordenadas en el centroide de cada una de las caras para una aplicación más cómoda de las fórmulas, de forma parecida a como lo hacíamos para calcular los posibles ejes de simetría facial. La diferencia fundamental, es que ahora el centroide ha de ser calculado después de calcular la ecuación del plano que contiene a la cara, de modo que sea el resultado de la intersección del plano que la contiene y la recta que contiene la proyección bidimensional del centroide.

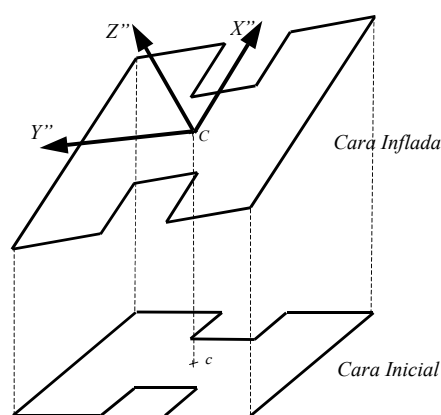


Figura 5-13. Cara inflada y su proyección.

Por lo tanto, hay que calcular la ecuación del plano para cada una de las caras del grafo. (§3.5.2)

Como ya se ha dicho, para comprobar la simetría se trasladan los ejes de coordenadas al centroide de cada una de las caras, de modo que se hace coincidir al plano  $XY$  del nuevo sistema de coordenadas con el plano en el que está contenida la cara. Además, el sistema de referencia se orienta de modo que el eje  $X$  quede coincidente con el supuesto eje de simetría facial. Por lo tanto habrá que hacer un cambio del sistema de coordenadas basado en:

- Una primera traslación del origen de coordenadas al centroide  $(x_c, y_c, z_c)$  en 3D de la cara plana:

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x_c & -y_c & -z_c & 1 \end{bmatrix}$$

Teniendo en cuenta que:

$$R = \sqrt{u_2^2 + u_3^2} \quad ; \quad R^2 = 1 - u_1^2$$

$$\sin N = \frac{u_2}{R} \quad ; \quad \cos N = \frac{u_3}{R} \quad ;$$

$$\sin L = u_1 \quad ; \quad \cos L = R$$

- Un giro del eje  $X$  un cierto ángulo  $N$ :

$$G_{X,N} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & u_3/R & u_2/R & 0 \\ 0 & -u_2/R & u_3/R & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

*Ecuación 5-18*

## Percepción Artificial de Dibujos Lineales

- Un giro del eje  $Y$  un ángulo  $-L$ :

$$G_{Y,-L} = \begin{bmatrix} R & 0 & u_1 & 0 \\ 0 & 1 & 0 & 0 \\ -u_1 & 0 & R & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 5-19

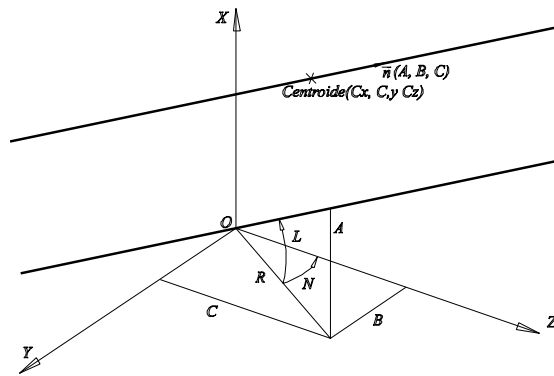


Figura 5-14. Cambio de sistema de referencia.

Con estos tres movimientos hacemos que el eje  $Z$  coincida con la dirección de la normal al plano, y las coordenadas  $XY$  se encuentren sobre el plano en el que se encuentre la cara. La composición de estas tres transformaciones se resume en:

$$T \cdot G_{X,N} \cdot G_{Y,-L} = \begin{bmatrix} R & 0 & u_1 & 0 \\ -u_1 u_2 / R & u_3 / R & u_2 & 0 \\ -u_1 u_3 / R & -u_2 / R & u_3 & 0 \\ -x_c R + \frac{u_1 u_2}{R} y_c + \frac{u_1 u_3}{R} z_c & -\frac{u_3}{R} y_c + \frac{u_2}{R} z_c & -u_1 x_c - u_2 y_c - u_3 z_c & 1 \end{bmatrix}$$

Ecuación 5-20

Después de esta transformación el sistema de referencia obtenido es denominado  $X'$ ,  $Y'$ ,  $Z'$ , y para obtener el sistema final deseado ( $X''$ ,  $Y''$ ,  $Z''$ ), sólo queda hacer coincidir uno de los ejes con el posible eje de simetría facial oblicua a medir. Para ello, se ha elegido el

eje  $X$ . Así la última transformación de coordenadas será un giro alrededor del eje  $Z$  un ángulo  $\theta$ , que se puede expresar como:

$$G_{Z,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} eje_{x'} & -eje_{y'} & 0 & 0 \\ eje_{y'} & eje_{x'} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 5-21

siendo  $eje_{x'}$  y  $eje_{y'}$  las componentes  $x'$  e  $y'$  respectivamente del vector unitario de la línea que define el eje de simetría respecto a los últimos ejes coordenados.

Así se puede resumir:

$$\begin{aligned} x'' &= R \cdot eje_{x'} \cdot (x - x_c) + \frac{u_3 \cdot eje_{y'} - u_1 \cdot u_2 \cdot eje_{x'}}{R} \cdot (y - y_c) + \frac{-u_1 \cdot u_3 \cdot eje_{x'} - u_2 \cdot eje_{y'}}{R} \cdot (z - z_c) \\ y'' &= -R \cdot eje_{y'} \cdot (x - x_c) + \frac{u_3 \cdot eje_{x'} + u_1 \cdot u_2 \cdot eje_{y'}}{R} \cdot (y - y_c) + \frac{u_1 \cdot u_3 \cdot eje_{y'} - u_2 \cdot eje_{x'}}{R} \cdot (z - z_c) \\ z'' &= u_1 \cdot (x - x_c) + u_2 \cdot (y - y_c) + u_3 \cdot (z - z_c) \end{aligned}$$

Ecuación 5-22

### 5.5.2. Cálculo del Coste de la Regularidad.

El coste de la regularidad de simetría facial oblicua se ha formulado mediante dos sumandos. Un primer valor será la suma de las coordenadas  $Y''$  de los vértices que componen la cara que contiene el eje de simetría. El motivo es bastante obvio, puesto que el eje de simetría lo hemos hecho coincidir con el eje  $X''$ , si la cara fuese simétrica respecto a ese eje, este valor debería ser nulo.

Pero esta condición no es suficiente para asegurar la simetría. Un ejemplo sencillo de ello se muestra en la Figura 5-15, y por ello se añade un segundo sumando. Éste computa los valores de las coordenadas  $x''$ , de modo que, si el punto se encuentra en el lado positivo de las  $y''$  sumaremos su coordenada  $x''$  y se la restaremos en caso negativo. Obviamente, ese sumatorio debía ser nulo en caso de una figura simétrica respecto al eje  $X$ .

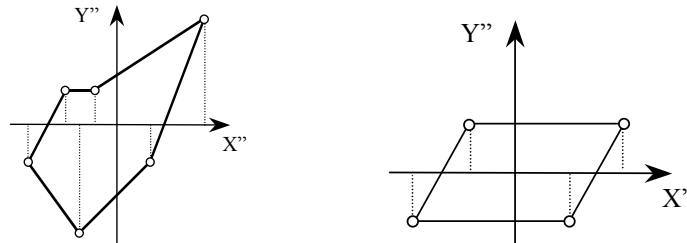


Figura 5-15. Ejemplos sin simetría cuya suma de coordenadas  $Y''$  de sus vértices es nula.

$$\sum y'' = 0$$

$$\sum_{y'' < 0} x'' - \sum_{y'' > 0} x'' = 0$$

Ecuación 5-23

En definitiva, la formulación del coste de la simetría facial oblicua que proponemos es la dada en la Ecuación 5-24.

$$\text{Coste\_de\_Simetría\_Facial\_Oblicua}_i = w_k \cdot \frac{1}{\text{núm. de ejes}} \left( \sum y'' + \sum_{y'' < 0} x'' - \sum_{y'' > 0} x'' \right)$$

Ecuación 5-24

teniendo  $w$  el valor 1 para cada  $k$  eje de simetría facial encontrado en la cara  $i$ .

Obviamente, El coste de la simetría como regularidad puede ser formulado de muchas formas distintas. La elección de esta formulación se ha basado en la sencillez tanto matemática como computacional, y la implementación y prueba del método han ratificado unos tiempos de ejecución aceptables (incluso con una programación no optimizada) y un porcentaje de error (entendido en términos de obtener modelos que no respeten las simetrías detectadas) razonablemente bajos.

## 5.6. CONCLUSIONES

---

En este capítulo se ha tratado una aproximación relativamente simple a una de las regularidades más importantes que podemos encontrar en los dibujos e imágenes de

componentes industriales. Pese a la importancia de esta característica, ha sido muy poco estudiada en el campo de la reconstrucción, y cuando se ha estudiado, ha sido de un modo muy circunstancial, sin profundizar en la formulación y en la detección de ejes de simetría facial.

Aquí se ha pretendido dar un enfoque más riguroso para la detección de ejes de simetría oblicua y se ha ofrecido una formulación detallada de todo el método empleado. Para ello se ha recogido información de su utilización en otras aplicaciones y se han adaptado y adecuado al tipo de dibujos de entrada de que disponemos. Por último, y aunque no es el objetivo de la presente tesis, también se ha ofrecido una formulación clara y sencilla para la utilización de esta información en el proceso de reconstrucción mediante optimización.

En efecto, la reconstrucción a partir de ejes de simetría oblicua no es el objetivo final de la tesis; siendo la detección de ejes dichos de simetría oblicua un paso previo para la detección de una regularidad mucho más potente y general. De dicha regularidad se hablará ampliamente en el siguiente capítulo.





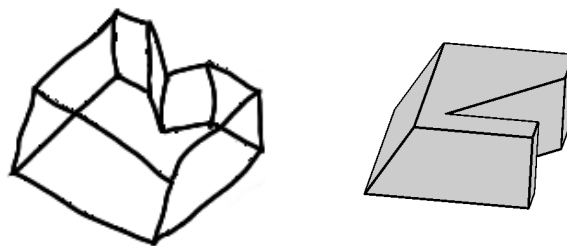
# Capítulo 6. Simetría de Modelos 3D detectados en el Espacio 2D

---

## 6.1. INTRODUCCIÓN

---

Muchos de los objetos fabricados son simétricos. Esto se debe a que la simetría hace que los objetos sean más fáciles de interpretar y de fabricar, y por otra parte, suele ser un requerimiento funcional o estético. Algunos estudios han mostrado que una larga proporción de componentes industriales tienen cierto grado de simetría [LMM01]. Las mismas razones que hacían que la simetría facial oblicua fuera una cualidad deseable en los objetos, hacen que un modelo simétrico sea tanto o más deseable. En otras palabras, la simetría es para el diseñador una característica deseable y explícita de diseño; por lo tanto, un sistema que intenta detectar y hacer explícitas las intenciones de diseño debe considerar la simetría.



*Figura 6-1. Boceto 2D y Modelo 3D reconstruido sin tener en cuenta la simetría.*

## Percepción Artificial de Dibujos Lineales

---

Desde otro punto de vista, la simetría es un concepto fundamental que la percepción visual humana utiliza [Pal99]. Los efectos de su ausencia pueden verse claramente en la Figura 6-1 y en el alzado, la planta y el perfil de la Figura 6-2e , que muestra el modelo 3D reconstruido mediante un proceso de optimización en el que no se ha considerado la simetría. La falta de simetría es la que hace poco plausible al objeto reconstruido.

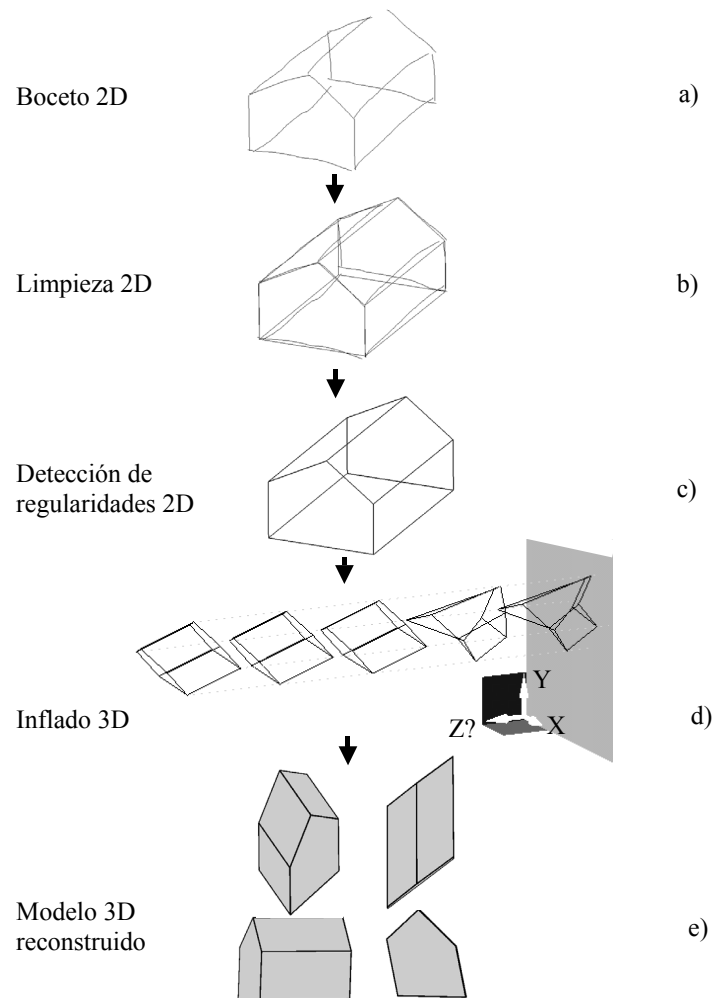


Figura 6-2.. Del boceto al modelo 3D.

Existen trabajos previos sobre detección y la aplicación de la simetría al modelado de objetos y su reconocimiento en modelos sólidos [TJ03]. Pero ninguno de ellos está

orientado hacia la reconstrucción a partir de dibujos lineales, y en ningún caso se resuelve el problema de detectar los planos de simetría en representaciones planas de los modelos que se pretende obtener.

Todo ello hace pensar que la detección de la simetría y su utilización en el proceso de reconstrucción es un problema complejo, y que su resolución podría mejorar el comportamiento de la optimización y los resultados obtenidos.

Así en este capítulo, se define un método original para la detección de planos de simetría en representaciones bidimensionales de objetos poliédricos, y su posterior utilización para la reconstrucción del modelo. En concreto, se exponen dos técnicas fundamentalmente para la aplicación de la simetría al proceso de reconstrucción, discutiéndose sus resultados. Ambas han ofrecido muy buenos resultados en la reconstrucción de modelos simétricos.

## 6.2. DETECCIÓN DE PLANOS DE SIMETRÍA EN POLIEDROS

---

En una representación bidimensional de un objeto poliédrico, simétrico y tridimensional, los únicos datos que se poseen son las aristas y vértices que delimitan las caras del poliedro que se quiere representar. A partir de estos datos se pueden obtener las caras que forman el poliedro. El problema de la detección de caras es complejo, pero algunos de los métodos más completos que existen para abordarlo ya se han tratado en el §Capítulo 4.

La intersección de un plano cualquiera con un poliedro (el modelo tridimensional) ofrecería como resultado uno o más caminos cerrados de líneas (uno o varios polígonos). El modo para reconocer un plano de simetría en el espacio bidimensional tiene como base esta misma idea: encontrar un camino cerrado de líneas que represente tal plano de simetría. A tal camino lo hemos denominado *polígono de simetría*. Para determinarlo, utilizamos técnicas semejantes a las que se han descrito en el capítulo de detección de caras, introduciendo criterios originales, los cuales hemos desarrollado a partir de un análisis de las características propias del tipo de camino buscado.

El circuito cerrado que se busca coincidiría con la intersección de dicho plano con algunas de las caras y aristas del modelo. Por lo tanto, el polígono de simetría que determina un plano de simetría en un objeto no tiene por qué estar formado únicamente por aristas del objeto, sino que también puede estar formado por ejes de simetría oblicua que se encuentran sobre las caras del polígono (por ejemplo la línea  $P_3-U_3$  de la Figura 6-3 –

## Percepción Artificial de Dibujos Lineales

---

derecha). Es decir, se trata de una simetría parcial formando parte de una simetría total. Ya se vio en el capítulo anterior como detectar estos ejes de simetría facial.

A partir de ahora, cuando se haga referencia a los lados que forman el polígono de simetría se hablará de aristas y vértices cuando se refiera a componentes del dibujo original, y se hablará de ejes cuando un lado del polígono sea un eje de simetría facial oblicua. Asimismo se hablará de puntos medios haciendo referencia a los puntos medios de aristas del dibujo original. No obstante, cuando la precisión se considere innecesaria, y a fin de no complicar en exceso la explicación, se utilizarán los términos de aristas y vértices como genéricos.

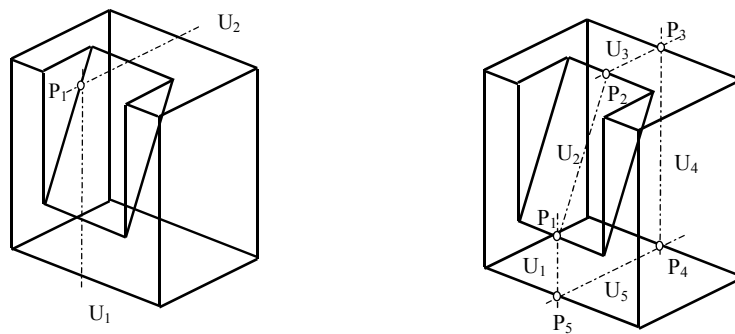


Figura 6-3. Plano de simetría de un modelo representado por un circuito 2D

Puesto que la detección del plano se realiza sobre la proyección bidimensional del objeto, los polígonos de simetría detectados representan planos reales de simetría vistos desde un punto de vista desconocido. Es por ello que a estos planos se les podría llamar *Planos Oblicuos de Simetría*.

El polígono de simetría puede estar formado no sólo por los ejes de simetría facial (aristas imaginarias), sino también por aristas del grafo inicial. Por lo tanto, las líneas que representan el plano de simetría pueden ser de diferentes formas:

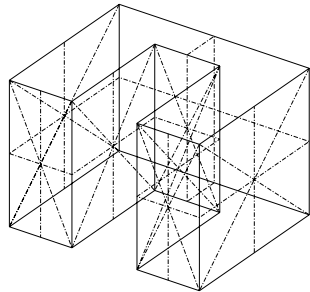
- ❑ que comiencen y acaben en el punto medio de aristas de grafo,
- ❑ que comiencen y acaben en vértices del grafo,
- ❑ que comiencen en un punto medio de una arista y acaben en un vértice (o a la inversa),
- ❑ o se trate de una de las aristas del modelo.

## Capítulo 6. Simetría de Modelos 3D detectados en el Espacio 2D

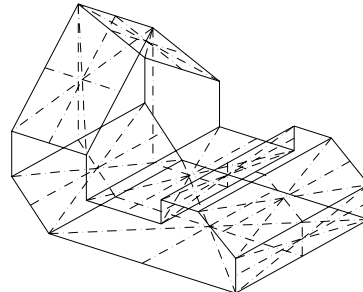
---

Como paso previo a la detección del circuito que representa al plano de simetría, se deben calcular las caras del modelo, así como los ejes de simetría oblicua que se pueden encontrar en cada una de las caras.

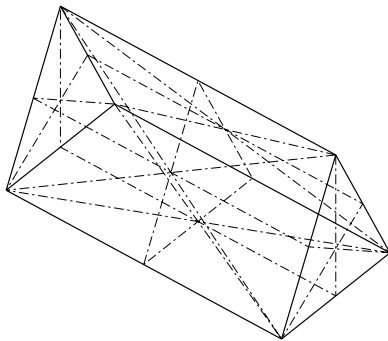
Si se puede encontrar una secuencia de ejes y aristas que concurren en sus puntos finales consecutivamente formando un polígono, entonces se considera que el plano que contiene a ese polígono es un posible plano de simetría.



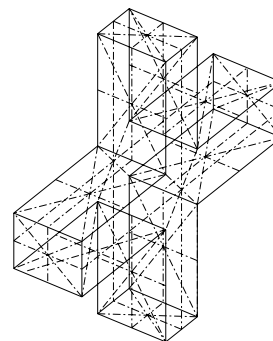
a) 24 aristas  
40 ejes de simetría facial oblicua



b) 34 aristas  
35 ejes de simetría facial oblicua



c) 9 aristas  
18 ejes de simetría facial oblicua



d) 36 aristas  
56 ejes de simetría facial oblicua

Figura 6-4. Modelos proyectados con todos sus ejes de simetría facial oblicua representados.

Pero el número de líneas candidatas a pertenecer al plano (o planos) de simetría es demasiado alto. Así por ejemplo, el dibujo lineal representado en la Figura 6-4- b) tiene 34 aristas en su grafo, y se detectan 35 ejes de simetría facial con el método expuesto en el capítulo anterior, con lo que el número de líneas candidatas es de 69. Esto hace que la tarea de encontrar un circuito cerrado, con estas líneas como candidatas, que determine un plano de simetría sea muy difícil. Es decir, el número de líneas candidatas es muy elevado para encontrar un circuito cerrado de líneas que representasen un posible plano de simetría. Incluso para un ser humano “entrenado” en la interpretación de dibujos lineales es una tarea complicada poder identificar los ejes que realmente deben pertenecer al plano y cuáles se deben descartar.

Tampoco si se quiere resolver a través de un algoritmo se encuentra una solución sencilla, puesto que el problema se puede plantear como la búsqueda de caras planas de un grafo que tiene por aristas la suma de las aristas del modelo más los ejes encontrados. Pero con esto se obtiene que el grafo ni siquiera es conexo, y que si se descompone en grafos conexos más sencillos puede que estos sean planares o que no lo sean. Por lo tanto no se podría aplicar ninguno de los métodos de detección de caras basado en la teoría de grafos, tal y como se presenta en el Capítulo 4.

Como consecuencia de todas las dificultades descritas arriba, se ha optado por buscar alguna estrategia para la reducción del número de líneas candidatas para cerrar el polígono de simetría. Es por ello que se opta por la aplicación de ciertas reglas que simplifiquen la obtención de los posibles planos de simetría oblicua.

### 6.2.1. Reglas fundamentales para el cálculo de los planos de simetría.

Se formulan tres reglas fundamentales para reducir el número de líneas candidatas a pertenecer al plano de simetría buscado:

1. Cuando un eje de simetría facial atraviesa una cara y termina en el punto medio de una arista, la cara adyacente que comparte esta arista con la primera, también debe tener un eje de simetría facial que pase por el punto medio de esta arista. Éste es el caso de los ejes  $e_1$  y  $e_2$  en la Figura 6-5 - a). Todos aquellos ejes de simetría facial que terminando en un punto medio de una arista, no cumplan esta condición serán eliminados. Con esta regla, y dependiendo del ejemplo con el que trabajemos, podemos eliminar un gran número de ejes.

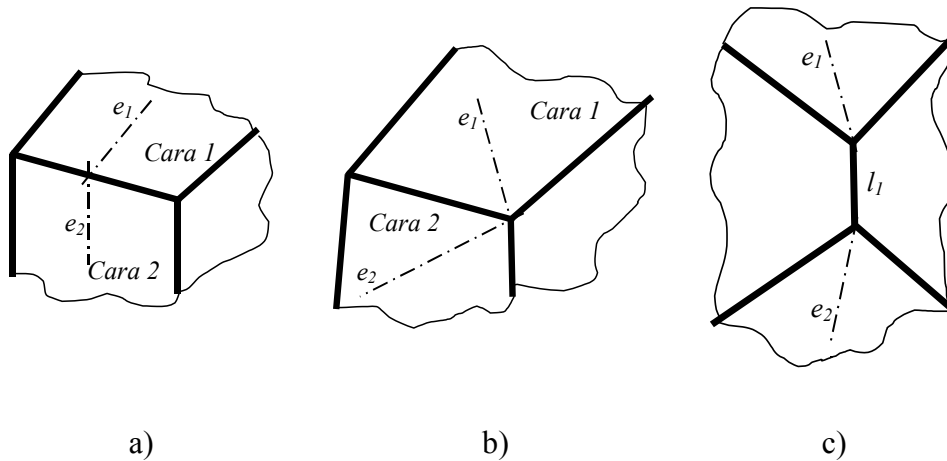


Figura 6-5. Reglas para la reducción de líneas candidatas a formar parte de los planos de simetría del modelo.

2. Aquellos ejes de simetría facial que pasan a través de un vértice pueden continuar el circuito por otro eje de simetría o por una arista del grafo. Pero dos líneas consecutivas no pueden pertenecer a la misma cara. Por lo tanto, si un eje termina en un vértice se admiten dos posibles opciones:
  - a) Que alguna cara adyacente que comparta con la primera ese vértice posea un eje de simetría facial que pase por el mismo vértice, como los ejes  $e_1$  y  $e_2$  en la Figura 6-5 - b).
  - b) Que la cadena de líneas que forman el plano de simetría continúe por cualquiera de las aristas del dibujo que terminan en ese vértice, pero no pertenecen a la cara a la cual pertenece el eje de simetría inicial (como el eje  $e_1$  y la arista  $l_1$  en la Figura 6-5 - c)). El otro extremo de la arista seleccionada como candidata debe formar parte de otro eje de simetría. Por supuesto, este segundo eje de simetría debe pertenecer a una cara diferente a las que comparten la arista candidata, y obviamente también diferente a la primera cara. Así en la Figura 6-5 - c) se estaría haciendo referencia a la secuencia  $e_1 - l_1 - e_2$ .
3. Si dos líneas candidatas consecutivas ( $l_1$  y  $l_2$ ) terminan en puntos ( $v_1$  y  $v_2$ ) de una misma cara, puesto que el plano de simetría oblicua debe estar representado por

## Percepción Artificial de Dibujos Lineales

---

una cadena cerrada de líneas (ejes o aristas), para que  $l_1$  y  $l_2$  pertenezcan a un plano de simetría, debe existir una línea (de  $v_1$  a  $v_2$ ) en esa cara que cierre el circuito. Si esa tercera línea no existe, las otras son excluidas como candidatas. Por ejemplo en la Figura 6-6, las líneas  $l_1$  y  $l_2$  no pueden pertenecer a un único plano de simetría porque no existe ninguna línea entre  $v_1$  y  $v_2$  en la cara a la cual los dos puntos pertenecen. Esta observación puede ser extendida a circuitos con mayor número de líneas.

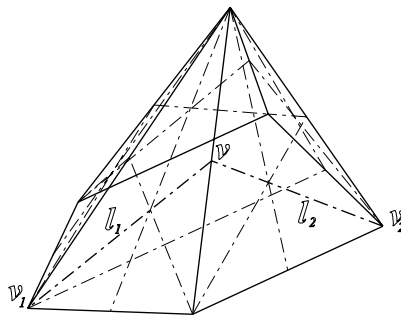


Figura 6-6 Grafo con sus ejes de simetría oblicua identificados

Cualquier eje que no cumpla cualquiera de estas tres reglas, es descartado como línea candidata a formar parte del plano de simetría. Y en la mayoría de los casos, descartar un eje implica descartar toda una cadena de ejes. Pero incluso siendo así, muy frecuentemente nos encontramos con demasiadas líneas candidatas. Por esa razón se utiliza otra condición más de naturaleza heurística elegida después de procesar un elevado número de bocetos y encontrar que ofrece buenos resultados:

4. Todos los ejes de simetría facial candidatos que conecten dos vértices en una cara, serán eliminados.

Hay que destacar, para evitar confusiones, que esta última regla se aplica sólo a ejes de simetría facial, y nunca a aristas de caras; sólo a ejes de simetría facial oblicua que comienzan y terminan en algún vértice. Con esta regla adicional, obtenemos muy buenos resultados en nuestra búsqueda de planos de simetría.

Esta regla tiene un inconveniente importante en algunos casos, como por ejemplo en la Figura 6-7, donde aplicando esta regla descartamos todos los ejes de simetría facial como



el  $e_1$ , y con esto perdemos la detección de algunos planos de simetría *verdaderos*. Pero este punto es discutido con mayor profundidad en el apartado 6.3.

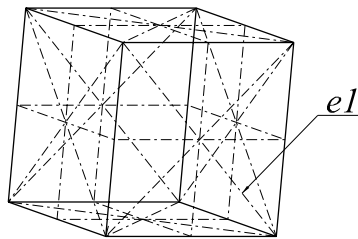


Figura 6-7. Cubo con todos sus ejes de simetría facial detectados

### 6.2.1.1. Generación de Planos de Simetría

En la sección anterior se han dado un número de reglas que permiten descartar ciertos ejes de simetría facial que no forman parte del plano de simetría. Pero con esto solo tenemos una serie de “líneas” desordenadas con las que constituir un polígono cerrado que represente uno o varios planos de simetría. Así el objetivo de esta sección es obtener listas ordenadas y cerradas de líneas que representen planos de simetría.

Se crea una matriz, que llamaremos *Matriz de Ejes*, cuyo número de columnas coincide con el número de nuevos ejes detectados. En cada una de estas columnas se almacenan los datos que de ellos se conocen, que son su inicio y final (vértice o punto medio de arista) y también a qué cara pertenece.

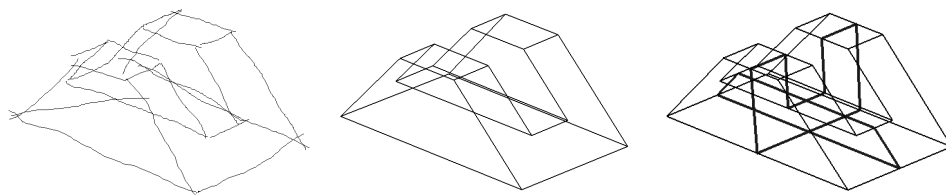


Figura 6-8. Proceso de introducción de datos y reconocimiento de los planos de simetría del modelo.

Se comienza escogiendo un eje aleatoriamente, y se busca en la matriz de ejes si hay algún otro eje que coincida en uno de sus extremos con el extremo final del eje seleccionado. Si eso ocurre, ese eje se añade como una de las aristas del polígono de

## Percepción Artificial de Dibujos Lineales

---

simetría buscado. Ese eje se convierte en el eje de referencia para continuar la búsqueda de los lados del polígono. Así la búsqueda se repite hasta que un extremo del eje seleccionado coincide con el inicial. En ese caso se habrá cerrado el polígono y el plano quedará definido por medio de él mismo. Por ejemplo en la Figura 6-8 siguiendo este método se detectan dos planos de simetría.

Esta búsqueda funciona sin mayor complicación si el plano no transcurre por ninguna de las aristas del dibujo inicial. En ese caso, como en la Figura 6-9, el extremo del último eje añadido no coincide con el de ningún otro eje. Por ello hay que buscar todas las aristas concurrentes en este vértice y comprobar si del otro extremo de esa arista parte otro eje que pueda ir cerrando el polígono. Pero esta última condición no es suficiente para aceptar a la arista como uno de los lados del polígono de simetría, especialmente cuando se trabaja con modelos que poseen numerosas caras triangulares (Figura 6-10). Es por ello que se le añade otra condición:

- el eje de simetría continuación de la arista no debe pertenecer a ninguna de las caras ya visitadas por el resto de los lados del polígono de simetría.

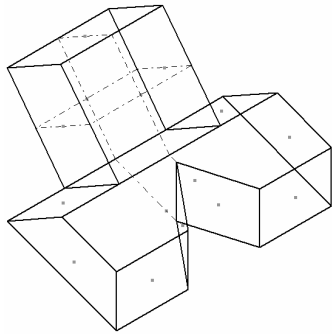


Figura 6-9. Modelo sobre el que se ha detectado un plano de simetría global y otro parcial.

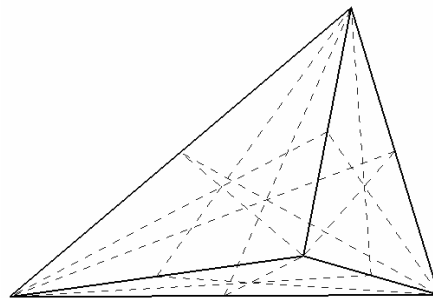


Figura 6-10. Modelo de caras triangulares con todos los planos de simetría representados.

En el caso de que la cadena de líneas que representan el plano de simetría recorra una arista, las aristas contiguas pertenecerán a un plano que no coincida con ninguno de los dos que la arista es intersección. Sólo cuando el circuito llega a una arista y la recorre, o llega a un vértice y hay que decidir por cuál de las caras continúa el circuito, puede que el método sea más complejo. En los casos en los que el vértice es de grado 3, el problema

está completamente acotado, puesto que sólo tendrá un plano por el que discurrir el plano de simetría. Pero cuando el grado del vértice es mayor, la complejidad de la búsqueda del posible plano de simetría aumenta.

En algunos casos los resultados del método expuesto hasta el momento pueden ofrecer planos de simetría parciales. En poliedros formados por componentes que poseen simetría por separado, se pueden encontrar estos planos parciales de simetría. Así por ejemplo en la Figura 6-9 de los dos planos de simetría encontrados uno de ellos es total y un segundo es parcial. Este segundo en esta primera aproximación se debería descartar. Para ello únicamente contabilizando el número de vértices que quedan a un lado del circuito y al otro (ver el apartado 6.8.1. Identificación de los dos lados de un modelo simétrico), o el número de caras a ambos lados, se tendría identificado si se trata de un plano parcial o global.

### **6.3. LIMITACIONES DEL MÉTODO DE DETECCIÓN DE PLANOS DE SIMETRÍA EN REPRESENTACIONES 2D DE POLIEDROS**

---

A pesar de que el método explicado en la sección anterior está ofreciendo muy buenos resultados, con un coste computacional reducido en más de 150 bocetos evaluados, el método tiene diversas limitaciones.

En primer lugar, podemos observar que nuestro método de búsqueda de planos de simetría en grafos bidimensionales depende directamente de la correcta detección de caras y de la detección de ejes de simetría facial en ellos. Aunque son problemas resueltos por otros autores y también en esta misma tesis (ver Capítulo 4 y Capítulo 5), cualquier comportamiento anómalo de estos métodos afectaría a nuestro método.

En segundo lugar, la mayor limitación que posee el método se debe a la última regla utilizada para eliminar ejes candidatos en la búsqueda de los planos de simetría. Esta regla es muy útil en la mayoría de los ejemplos estudiados, pero puede eliminar algunos planos de simetría reales. Así por ejemplo, de los 9 planos de simetría que posee un cubo, con la aplicación de esta regla eliminamos 6 de ellos. Sin embargo, el programa aún encuentra los tres planos de simetría más obvios (Figura 6-11). Parece lógico decir que éstos son los tres planos principales de simetría del modelo tridimensional. Por lo tanto, el método detecta todos los planos de simetría más relevantes e importantes (ver la Figura 6-11).

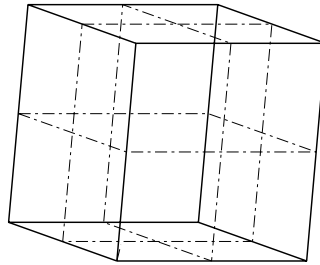


Figura 6-11. Cubo con sus tres planos de simetría principales detectados

El problema es importante en ciertos ámbitos. Por ejemplo, si la detección de los planos de simetría se utilizase en un programa de análisis, la detección de estos tres planos de simetría serviría para reducir enormemente el número de variables de cálculo. Sin embargo, veremos que para el propósito de esta tesis, la limitación no es importante. En cambio, la reducción de coste computacional que se consigue al aplicar la regla que impide encontrar estos planos de simetría sí que es importante.

El mismo caso se observa en la Figura 6-4 – d) y el resultado en la Figura 6-12 – d), donde aplicando esta regla, perdemos dos planos de simetría (los que pasan por los vértices  $s-q-u-v$  y  $p-r-w-t$ ). Pero, sin embargo encontramos tres planos de simetría, que psicológicamente podrían considerarse los planos fundamentales de simetría del modelo.

Una tercera limitación es la falta de detección de aquellos planos de simetría que intersectan a la pieza exacta y únicamente por aristas del grafo inicial. Puesto que dos aristas consecutivas forman un plano, si éstas forman parte del plano de simetría, la cadena de líneas que lo forman van a pertenecer necesariamente al mismo plano. Un ejemplo de ello podría ser la Figura 6-13, donde se ha señalado un plano de simetría ( $l_1-l_2-l_3-l_4$ ) formado únicamente por aristas del modelo, y donde no interviene ningún eje de simetría facial. Hay que destacar que ese mismo ejemplo tiene también otros dos planos de simetría más con la misma característica. El método propuesto impide encontrar ninguno de los tres planos de simetría mencionados, pero simplifica la detección de otros planos de simetría de mayor relevancia que estos (los dos indicados en la Figura 6-13-b)).

Así se cree que las limitaciones no son tan importantes como en un principio pudieron parecer. En la mayoría de los casos en los que la aplicación de la última regla tiene como consecuencia la falta de detección de alguno de los planos de simetría, facilita y acelera la detección de otros planos de simetría que perceptualmente son más relevantes. De hecho,

durante las pruebas no ha sido fácil encontrar ejemplos en los que no se detecten planos de simetría por esta razón.

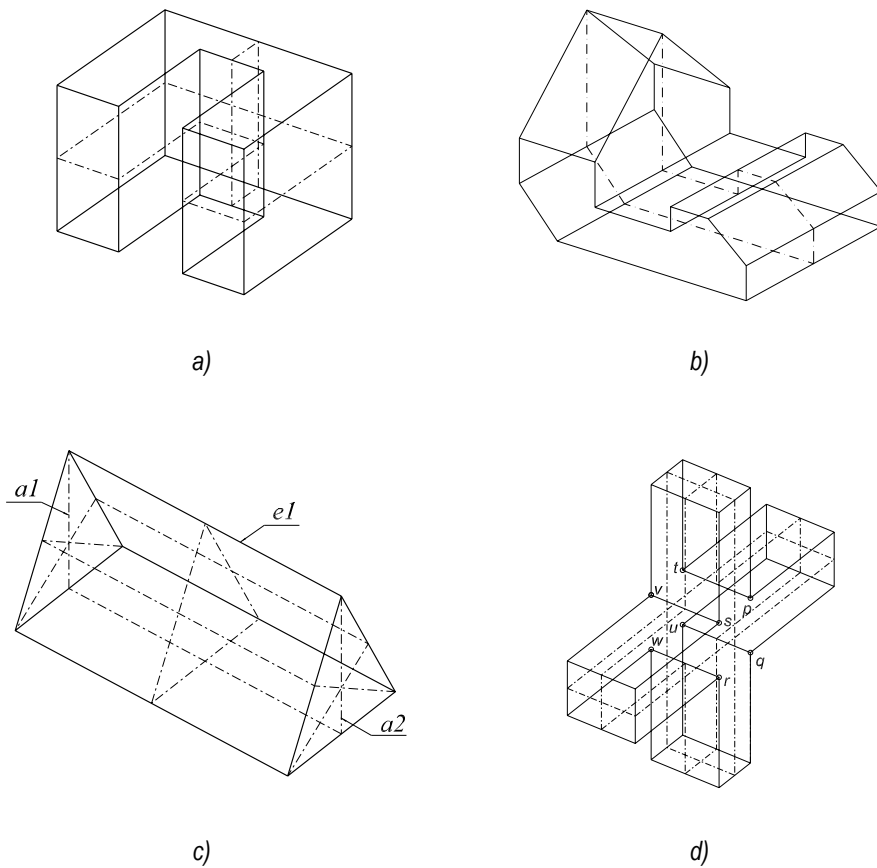


Figura 6-12. Modelos proyectados con todos los planos de simetría que detecta el método propuesto.

En cuanto a la complejidad en la búsqueda de la cadena cerrada de ejes y aristas que representen un plano, cabe decir que es reducida. En los casos en los que el eje de simetría acaba en el punto medio de alguna arista, la determinación de la presencia de alguna línea que continúe el circuito o no lo haga es inmediata: sólo si hay otro eje de

simetría perteneciente a la cara contigua, que pase por ese punto medio, será aceptado y en caso contrario, los dos ejes serán rechazados.

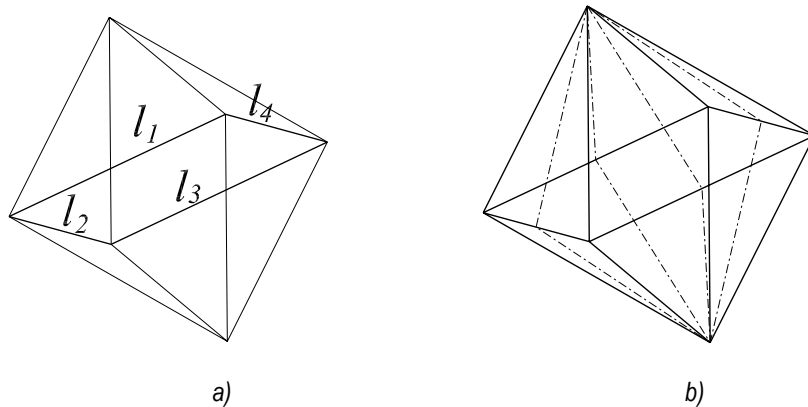


Figura 6-13. Modelo que contiene planos de simetría cuyo polígono de simetría está formado exclusivamente por aristas del modelo.

Con los planos de simetría identificados en el dibujo lineal de entrada, ahora es el momento de utilizar su conocimiento como parte activa dentro de la reconstrucción tridimensional.

## **6.4. PRIMERAS APROXIMACIONES A LA UTILIZACIÓN DE LOS PLANOS DE SIMETRÍA**

---

### **6.4.1. Planicidad del plano de simetría**

La detección de un plano de simetría en el grafo bidimensional nos ofrece mucha información sobre el modelo tridimensional que el grafo representa. Así, una primera apreciación y aplicación en el uso de los planos de simetría, como regularidad dentro de un proceso de reconstrucción basado en la optimización, sería que el plano de simetría debe ser tratado como una cara plana. Se trata de exigir la planicidad del plano de simetría, como si se tratara de una cara virtual, al igual que se puede exigir la planicidad de todas las caras del modelo detectadas.

Para ello se busca los coeficientes  $a$ ,  $b$ ,  $c$  y  $d$  de la ecuación del plano más probable (siendo  $ax + by + cz + d = 0$ ) formado por los vértices contenidos en el polígono de

simetría. Estos se calculan como el área del polígono proyectada sobre un plano coordenado. De esa forma,  $a$  se calculará como el valor unitario del área del polígono proyectada sobre el plano coordenado  $YZ$ ,  $b$  sobre el plano  $XZ$  y  $c$  sobre el  $XY$ . El coeficiente  $d$  se calculará mediante la sustitución del plano por uno de los vértices del plano. La formulación de este cálculo se muestra con más detalle en el capítulo anterior en §3.5.2.

El coste a minimizar en la función objetivo será la distancia entre los vértices que componen el polígono de simetría ( $v_k$ ) al propio plano de simetría, que de forma simplificada podría ser calculado como:

$$\sum_{k=m}^{n-1} |ax_k + by_k + cz_k + d|$$

Ecuación 6-1

Siendo  $k=m, \dots, n-1$  cada uno de los vértices y/o puntos medios que determinan el polígono de simetría asociado al plano de simetría considerado.

Esta condición se cumple trivialmente en el plano del dibujo, por lo tanto no será capaz de reconstruir ningún modelo por sí sola puesto que no hará que la figura escape del plano 2D. Sin embargo, la condición se debería cumplir y con ello minimizar su valor en todos los modelos en los que los planos de simetría hayan sido calculados correctamente. Por lo tanto si se quiere aplicar esta formulación, debe ser acompañada por un método de inflado previo o junto a otras regularidades que consigan que el modelo escape del óptimo trivial.

### 6.4.2. Distancia de los vértices del modelo al plano de simetría

Un segundo acercamiento al problema fue añadir a la formulación anterior la idea de que la distancia de dos de los puntos presuntamente simétricos deben tener la misma distancia respecto del plano de simetría en el modelo 3D. Por lo tanto, en el caso de que el conjunto los  $n$  vértices del modelo sean simétricos deben cumplir la siguiente ecuación:

$$\sum_{i=0}^{n-1} \text{distancia}(v_i - \text{plano}) = 0$$

Ecuación 6-2

Siendo  $i=0, \dots, n-1$  cada uno de los vértices del modelo considerado, y siempre que las distancias se consignent con su signo.

Pero los resultados no ofrecen resultados tan buenos como se esperaban. Esto es debido a que para una misma distancia dada habrá dos planos posibles sobre los que se encuentren los dos puntos simétricos, uno en el sentido positivo y otro en el negativo del plano. Es decir, esta condición no es suficiente: todos los puntos de los dos planos paralelos distarán lo mismo del plano de simetría, por lo tanto cualquiera de sus puntos cumplirían la condición de distancia. Como consecuencia, el resultado probablemente nos ofrezca la mayoría de sus aristas oblicuas con respecto a la solución esperada, y, además, con independencia entre ellas, lo que acaba ofreciendo una imagen sin ningún tipo de simetría, ni siquiera oblicua.

### 6.4.3. Ortogonalidad respecto al plano de simetría

Una solución que ofrezca simetría oblicua se podría subsanar contabilizando en la función objetivo el grado de perpendicularidad entre la línea determinada por un vértice y su presuntamente simétrico respecto al plano de simetría.

Esta última condición es fácil de comprobar para todos aquellos puntos finales de aristas cuyo punto medio forman parte del plano de simetría. Pero para el resto de los vértices, si realmente queremos que se cumpla la condición de simetría, deberíamos encontrar la correspondencia entre vértices simétricos: una tarea que en principio no es obvia. Para paliar parcialmente esta dificultad, se decidió aplicar la ortogonalidad a aquellas aristas atravesadas por el polígono de simetría por su punto medio, cuyos extremos sin ningún otro cálculo deberían ser claramente simétricos.

### 6.4.4. Aplicación.

Basada en este primer método de utilización de la simetría del modelo, se decidió formularla como una regularidad compuesta por tres sumandos; las tres condiciones anteriormente expuestas. Para que las medidas de planicidad, distancia y ortogonalidad tuvieran la misma importancia durante el proceso de optimización, se decidió normalizar sus valores.

Con la agrupación de estas tres condiciones, el sistema es capaz de reconstruir una pieza simétrica desde su representación 2D a un modelo tridimensional únicamente mediante



optimización de esta regularidad; sin necesidad de utilización de un inflado previo o de la combinación con cualquier otra regularidad que nos haga salir del mínimo local.

Pero los resultados nos ofrecen tiempos de cálculo demasiado elevados y unos modelos no siempre aceptables (ver el ejemplo de la Figura 6-14). Por ello se requiere de una nueva estrategia, tal y como se describe en las siguientes secciones de este mismo capítulo.

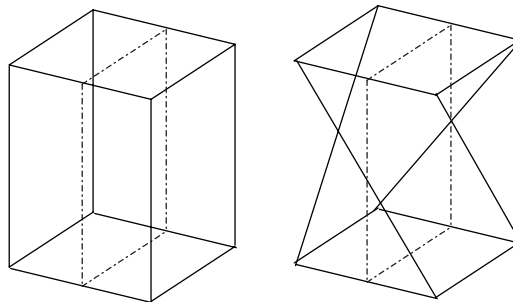


Figura 6-14. Modelo reconstruido inaceptable de un bloque prismático.

## 6.5. GENERALIDADES DE LOS NUEVOS MÉTODOS.

---

Una vez el plano de simetría se ha detectado, para conseguir tratar con simetrías, es conveniente introducir un nuevo sistema de coordenadas, que se ha venido a llamar *Sistema de Simetría*, y está definido en la siguiente sección. El cambio de sistema de referencia hace que la condición de simetría pueda ser formulada de forma más sencilla y compacta.

Además, para poder obtener la máxima información del modelo simétrico los vértices simétricos deben *emparejarse* adecuadamente. Sin embargo, establecer la relación de emparejamiento de vértices simétricos no es una tarea trivial teniendo como partida un dibujo en dos dimensiones. Únicamente se trataría de una tarea simple en el caso de que todos los vértices del primer subgrupo estuvieran conectados a su pareja del segundo subgrupo mediante una arista cuyo punto medio se sabe que pertenece al plano de simetría. Por ejemplo, las aristas  $l_1$ ,  $l_2$  y  $l_3$  en la Figura 6-15-a) ó las  $l_4$  y  $l_5$  en la Figura 6-15-b).

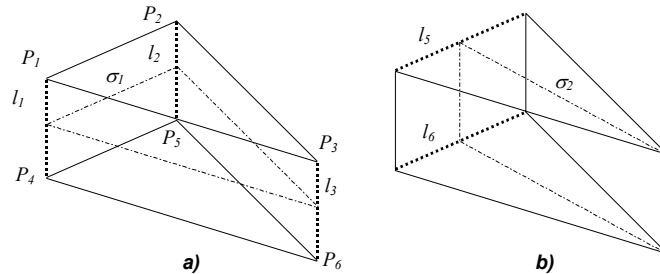


Figura 6-15. Determinación de subgrupos de vértices simétricos.

Sin embargo, el problema general de *Emparejamiento* es más complicado, y se debe resolver para obtener una formulación práctica de la *regularidad de simetría bilateral*.

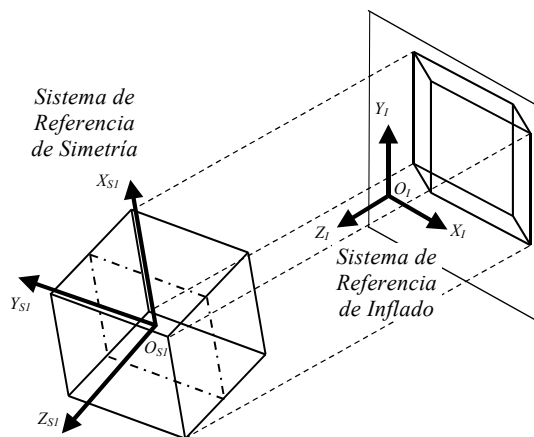


Figura 6-16. Sistemas de referencia utilizados en la aplicación de los planos de simetría en la Reconstrucción.

La segunda consideración general que se hizo fue, que puesto que un objeto simétrico tiene la misma información topológica por duplicado, un plano de simetría podría reducir el tamaño del problema de la reconstrucción haciendo reconstruir únicamente la mitad del objeto. Esta consideración fue el germen para desarrollar el segundo método para la reconstrucción de los modelos simétricos. Cuando se detectan múltiples planos de simetría, una mayor reducción del tamaño del problema puede obtenerse.

Así en las próximas secciones tienen como objetivo detallar dos nuevos métodos de utilización de la simetría en la reconstrucción. Un primer método formulando la simetría

como regularidad en un proceso de optimización. Y un segundo método basado en reducir el tamaño del problema exigiendo la simetría como restricción.

## 6.6. SISTEMAS DE SIMETRÍA

Llamamos *Sistema de Referencia de Simetría* o *Sistema de Simetría* a un nuevo sistema de referencia que se introduce para cada plano de simetría, de forma que el plano de simetría se haga coincidir con el plano  $XY$  de coordenadas. En la Figura 6-16 se representa un *sistema de simetría*  $(O_{SI} X_{SI} Y_{SI} Z_{SI})$ , junto con un sistema de inflado  $(O_I X_I Y_I Z_I)$ . Puesto que el centroide del polígono con el que se ha conseguido representar el plano de simetría permanecerá invariante tras la reconstrucción, el origen del nuevo sistema de coordenadas se sitúa en él. Los ejes  $X_{SI} Y_{SI}$  se sitúan sobre el plano de simetría y el eje  $Z_{SI}$  sale perpendicularmente del plano orientado según un sistema de referencia a derechas.

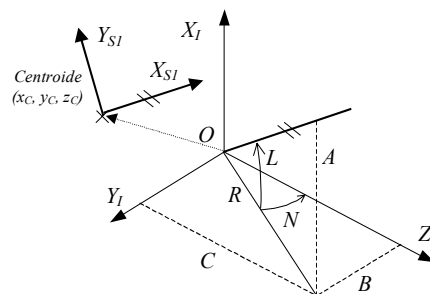


Figura 6-17. Transformación al sistema de referencia de simetría.

Denotemos  $(x_p, y_p, z_p)$  a las coordenadas de un vértice  $P$  en el modelo cuando lo referimos al sistema de referencia del inflado (el sistema inicial plano) y  $(x'_p, y'_p, z'_p)$  cuando lo referimos al sistema de simetría. De este modo la transformación encargada de realizar el cambio de coordenadas queda:

$$\begin{bmatrix} x'_p \\ y'_p \\ z'_p \\ 1 \end{bmatrix} = \begin{bmatrix} R & -u_1 u_2 / R & -u_1 u_3 / R & -x_c R + \frac{u_1 u_2}{R} y_c + \frac{u_1 u_3}{R} z_c \\ 0 & u_3 / R & -u_2 / R & -\frac{u_3}{R} y_c + \frac{u_2}{R} z_c \\ u_1 & u_2 & u_3 & -u_1 x_c - u_2 y_c - u_3 z_c \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix}$$

Ecuación 6-3

según la notación dada en el capítulo anterior (§ 5.5.1).

## 6.7. CONDICIÓN DE SIMETRÍA

---

Los  $n$  vértices de un modelo simétrico pueden ser reagrupados sin pérdida de generalidad de modo que satisfagan:

$$\begin{aligned} (1, 2, \dots, n) &= (1, 2, \dots, m), & \text{con } z'_1, \dots, z'_m > 0 \\ & (m+1, m+2, \dots, 2m), & \text{con } z'_{m+1}, \dots, z'_{2m} < 0 \\ & (2m+1, 2m+2, \dots, n) & \text{con } z'_{2m+1}, \dots, z'_n = 0 \end{aligned}$$

Ecuación 6-4

En otras palabras,  $m$  puntos se encuentran a un lado del plano de simetría, otros  $m$  puntos caen en el lado opuesto, y  $n-2m$  son los puntos que se encuentran contenidos en el propio plano de simetría ( $m$  podría ser nulo). Los puntos del primer subconjunto tienen su simétrico en los puntos del segundo subconjunto, de modo que, de nuevo sin pérdida de generalidad, los puntos pueden reordenarse de manera que el punto  $1$  tiene su correspondiente simétrico en el punto  $m+1$  y así sucesivamente. Esto da las siguientes condiciones:

$$\begin{aligned} x'_1 &= x'_{m+1}, \dots, x'_m = x'_{2m} \\ y'_1 &= y'_{m+1}, \dots, y'_m = y'_{2m} \\ z'_1 &= -z'_{m+1}, \dots, z'_m = -z'_{2m} \end{aligned}$$

Ecuación 6-5

Esta es la forma más sencilla de la formulación de la condición de simetría para los vértices, y es la formulación básica de la *regularidad de simetría bilateral*.

## 6.8. EMPAREJAMIENTO DE VÉRTICES SIMÉTRICOS.

---

Una vez detectado el plano de simetría, si se pretende llegar a una formulación completa de la simetría en el modelo, se debe resolver el problema del emparejamiento para que la Ecuación 6-5 tenga sentido. Es decir, se debe encontrar para cada vértice su correspondiente simétrico. Para ello hay que establecer una búsqueda minuciosa de los mismos.

El objetivo del algoritmo propuesto era la división de los vértices en los grupos descritos en la Ecuación 6-4, de forma que se identificasen aquellos vértices que se encontraban a un lado u otro del plano de simetría. Además, se sabía que era conveniente encontrar la "pareja" simétrica de cada vértice, aunque esto último parecía más difícil.

El algoritmo desarrollado comienza con la construcción de una *Matriz de Parejas Simétricas*,  $MS$  de tamaño  $2 \times n$ , donde  $n$  es el número de vértices del grafo inicial. En el elemento  $MS(1, i)$  se almacena el índice del vértice simétrico con respecto al vértice  $i$  (su pareja). En la segunda fila  $MS(2, i)$  se almacena una etiqueta que indica cuál es la posición relativa del vértice  $i$  con respecto al plano de simetría.

### 6.8.1. Identificación de los dos lados de un modelo simétrico.

Los vértices del dibujo lineal pueden tener en el modelo 3D tres posiciones relativas con respecto al plano de simetría:

- estar en uno de los dos lados en los que el plano divide el espacio, o
- estar contenidos en el propio plano de simetría.

Así el objetivo inicial es identificar para cada vértice su posición relativa. El modo elegido para señalar la posición relativa es añadir a cada vértice una etiqueta, la cual tendrá valor: +1 si pertenece a uno de los dos lados en los que el plano de simetría divide a la pieza, -1

si pertenece al otro lado, y 0 si se trata de un vértice contenido en el propio plano de simetría.

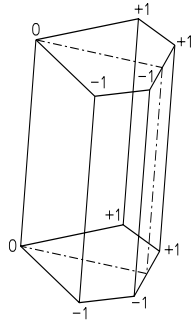


Figura 6-18. Vértices etiquetados en función de un plano de simetría

Para la identificación de cada uno de los vértices se comienza por una de las caras que son divididas por el plano de simetría, es decir, que contiene a uno de los lados del polígono de simetría. Se toma una de ellas al azar, y por ser la primera cara se le asigna arbitrariamente a un lado la etiqueta positiva y al otro lado la negativa.

Una vez se tiene la primera cara evaluada, se recorren las caras adyacentes a ésta que también son atravesadas por el plano de simetría. Este tipo de caras también tendrán sus vértices divididos a un lado y otro del plano de simetría. Además, al ser adyacentes a la primera, los vértices comunes a las dos son utilizados para garantizar la coherencia en la propagación del etiquetado.

Un tercer paso consiste en extender el etiquetado al resto de las caras. Las caras que no tienen ningún punto en común con el plano de simetría se irán escogiendo de forma que la cara escogida tenga al menos una arista en común con una cara ya evaluada. Este tipo de caras tendrá todos sus vértices etiquetados con un mismo valor (+1, o -1).

Por último aquellos vértices que se encuentran en el polígono de simetría son visitados y etiquetados con 0.

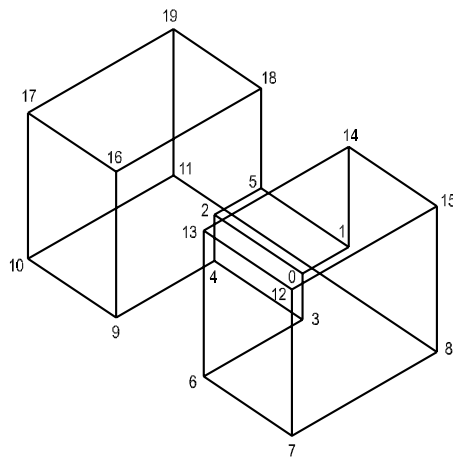
### 6.8.2. Emparejamiento de vértices simétricos.

Del mismo modo que para etiquetar la posición relativa de los vértices respecto al plano de simetría, para relacionar las parejas de vértices simétricos en primer lugar se recorren las caras por las que el plano de simetría secciona al modelo. Sus vértices son emparejados

## Capítulo 6. Simetría de Modelos 3D detectados en el Espacio 2D

mediante la ordenación simultáneamente de los mismos en sentido horario y anti-horario. Así por ejemplo, en la cara 11 de la Figura 6-20, si se comienza por el eje  $a$  que corta a la arista 11-8, ya se sabe que los vértices 11 y 8 son simétricos. Los vértices de media cara son recorridos en sentido horario (8-7-6-3) y la otra media cara en sentido anti-horario (11-10-9-4), así las dos listas resultantes se emparejan dando como resultado todas las parejas de vértices simétricos de esa cara.

El mismo proceso se realiza para el resto de caras cortadas por el plano de simetría (Ver Figura 6-20).



<i>Cara n°</i>	<i>Vértices en caras</i>
<i>Cara 0</i>	0, 2, 5, 1
<i>Cara 1</i>	2, 0, 3, 4
<i>Cara 2</i>	18, 16, 17, 19
<i>Cara 3</i>	6, 13, 12, 7
<i>Cara 4</i>	7, 12, 15, 8
<i>Cara 5</i>	0, 1, 14, 13, 6, 3
<i>Cara 6</i>	12, 13, 14, 15
<i>Cara 7</i>	17, 16, 9, 10
<i>Cara 8</i>	5, 2, 4, 9, 16, 18
<i>Cara 9</i>	17, 10, 11, 19
<i>Cara 10</i>	15, 14, 1, 5, 18, 19, 11, 8
<i>Cara 11</i>	11, 8, 7, 6, 3, 4, 9, 10

$$MS = \begin{pmatrix} 2 & 5 & 0 & 4 & 3 & 1 & 9 & 10 & 11 & 6 & 7 & 8 & 17 & 16 & 18 & 19 & 13 & 12 & 14 & 15 \\ +1 & +1 & -1 & +1 & -1 & -1 & +1 & +1 & +1 & -1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 & -1 \end{pmatrix}$$

Figura 6-19

Para todas esas caras la correspondencia es más sencilla de encontrar que para el resto, puesto que los dos vértices simétricos pertenecen a la misma cara y ambos están conectados a otros dos ya evaluados.

Una vez evaluadas todas las caras que poseen algún eje de simetría coincidente con el plano de simetría hay que evaluar el resto. En este caso, la simetría de vértices se establece entre caras diferentes, de modo que en primer lugar habrá que buscar la correspondencia entre estas caras. Así el objetivo es buscar caras simétricas.

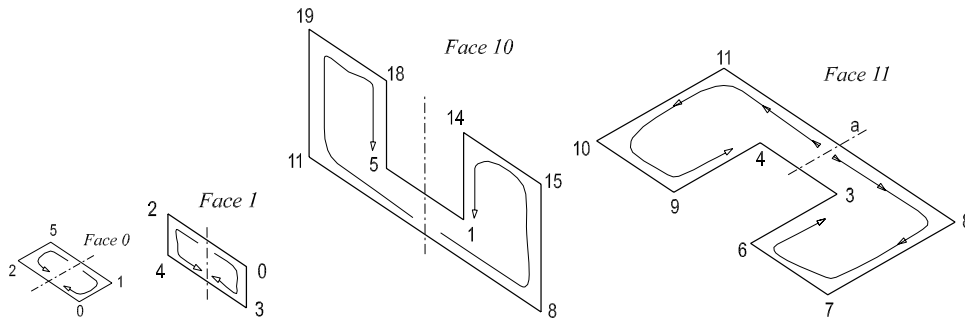


Figura 6-20. Caras atravesadas por el plano de simetría

La lista de caras que no han sido evaluadas por completo son recorridas en busca de al menos tres vértices evaluados en una misma cara ( $v_i, v_j, v_k \subset F_m$ ). Tres puntos no colineales determinan un único plano, de modo que los tres vértices simétricos a ( $v_i, v_j, v_k$ ) determinan su cara simétrica ( $v'_i, v'_j, v'_k \subset F'_m$ ). Pero aún cuando la cara simétrica ha sido encontrada, queda por determinar cuál es la correspondencia entre sus vértices. El orden de la correspondencia entre vértices viene dada por una arista de la cara, de modo que dos de los vértices conocidos deben ser consecutivos.

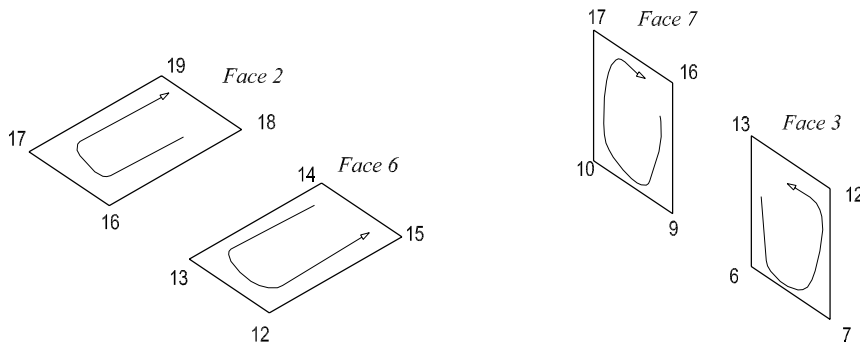


Figura 6-21. Caras Simétricas cuya relación de simetría de sus vértices se establece en sentido inverso.

Para ello se ordenan las dos caras de modo que comiencen por dos de estos vértices (una arista) y ellos son los que marcan la dirección en la que son ordenados los vértices de la cara (por ejemplo, en la Figura 6-21 los vértices 6-7 de la cara 3 y los 9-10 de la cara 7). Cabe hacer notar que la dirección relativa de las caras para obtener sus vértices simétricos no es obvia, puesto que las caras simétricas pueden establecer su correspondencia girando



en la misma dirección o en direcciones opuestas, y ésta debe ser determinada. Así, por ejemplo en la Figura 6-21 la correspondencia entre los vértices de las caras 3 y 7 se encuentra según direcciones opuestas (9-10-17-16 y 6-7-12-13). Sin embargo, en la Figura 6-22 las caras 9 y 4 mantienen su correspondencia con los vértices ordenados en el mismo sentido.

Por lo tanto el método empleado básicamente es el siguiente:

- ❑ Se toma una de las dos caras identificadas como simétricas y se ordena ( a derechas o a izquierdas) con la condición de que comience por dos vértices evaluados.
- ❑ La cara simétrica se ordena tomando los dos primeros vértices (una arista) aquellos que son simétricos de la primera cara y luego se siguen ordenando los vértices según la dirección tomada por esa primera arista.
- ❑

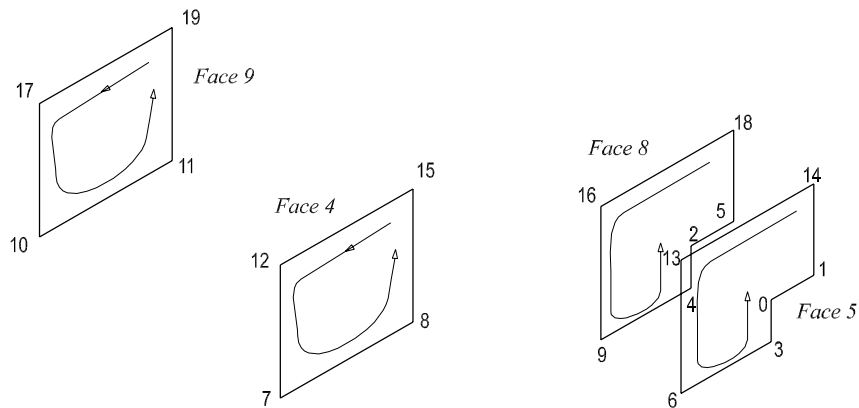


Figura 6-22. Caras Simétricas cuya relación de simetría de sus vértices se establece en el mismo sentido.

## 6.9. LA SIMETRÍA DEL MODELO COMO REGULARIDAD

Una vez el polígono de simetría se ha detectado y el emparejamiento de vértices se ha completado, la regularidad de simetría es fácil de formular. En la reconstrucción mediante

## Percepción Artificial de Dibujos Lineales

---

optimización, cada regularidad tiene un coste, y el objetivo es minimizar la suma de estos costes. El coste es cero cuando la condición deseada se ha conseguido, y difiere más de cero cuanto más nos alejamos del ideal.

La condición de simetría bilateral formulada en Ecuación 6-5 puede ser reformulada como la varianza ( $\sigma^2$ ) de la diferencia de coordenadas de todos los pares de vértices simétricos:

$$Coste_j = \sigma^2 (dif\_x') + \sigma^2 (dif\_y') + \sigma^2 (sum\_z')$$

Ecuación 6-6

$$\begin{aligned} dif\_x' &= (x'_i - x'_{m+i}) \\ dif\_y' &= (y'_i - y'_{m+i}) \\ sum\_z' &= (z'_i + z'_{m+i}) \end{aligned} \quad \text{para cada vértice } i=1 \dots m.$$

donde

$$dif\_x' = dif\_y' = 0,$$

y

$$sum\_z' = z'_i \quad \text{para cada vértice } i = 2m+1 \dots n.$$

La simetría se reformula en función de la varianza puesto que al ser una función cuadrática hace que la convergencia sea más rápida que una función lineal. Además, las desviaciones típicas se calculan asumiendo que la media sea igual a cero. Utilizando la media real no se mejoran los resultados de la reconstrucción, y hay que recordar que el objetivo de la Ecuación 6-6 no es otro que penalizar las malas soluciones y hacerlo empleando funciones que determinen regiones factibles tan convexas como sea posible. El objetivo real es encontrar variables en las cuales la media de cada sumando sea también cero, y cuando cada par de vértices simétricos cumple la condición, el coste de la regularidad alcanza el mínimo.

El coste que se define en la Ecuación 6-6 es para un solo plano de simetría. El coste total para un modelo con  $r$  planos de simetría queda definido como:

$$Coste\_de\_Simetría = \frac{1}{r} \sum_{j=1}^r Coste_j$$

Ecuación 6-7

## 6.10. RECONSTRUCCIÓN DEL MODELO PARTIENDO DE LA MITAD SIMÉTRICA

Con el objeto de agilizar el proceso de reconstrucción y conociendo las parejas de vértices simétricos, se tiene información suficiente para descomponer el dibujo de entrada en dos mitades y eliminar una de las mitades simétricas que formarían el modelo. Esto hace reducir el número de vértices, aristas y caras del dibujo a reconstruir. La reconstrucción de la mitad simétrica da como resultado un problema con menor número de variables, que la figura inicial y con ello se reducen los tiempos de cálculo del proceso de reconstrucción. Una vez se obtiene la mitad del modelo tridimensional, una función de simetría puede ser utilizada para obtener el modelo completo. Además, esta idea puede ser extendida a múltiples planos de simetría, lo que hace reducir el tamaño del problema consecutivamente.

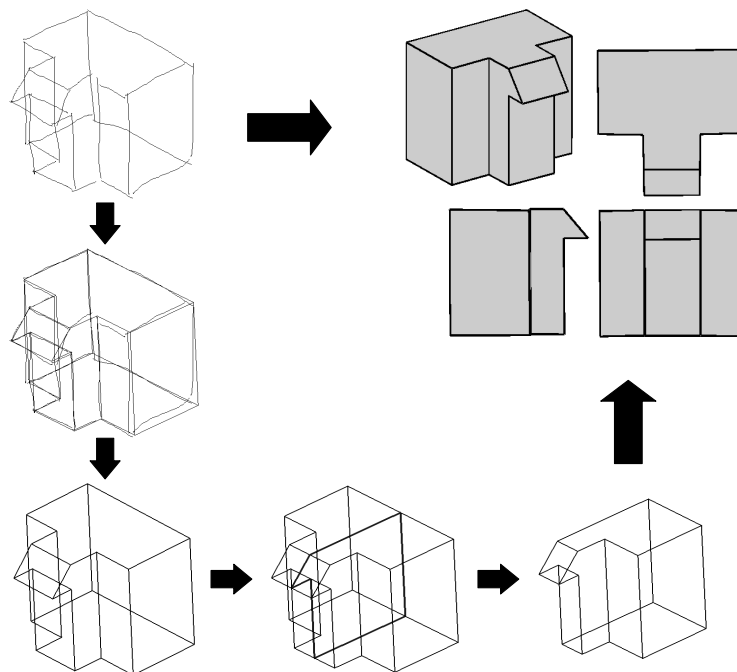


Figura 6-23. Generación de la mitad simétrica y posterior reconstrucción de la misma.

Puesto que se posee la información que relaciona cada uno de los vértices del grafo con uno de los lados simétricos (§ 6.8.1), es relativamente sencillo obtener la mitad del grafo.

Para ello hay que eliminar los vértices y aristas de una de las mitades. Al grafo resultante hay que añadirle los vértices contenidos en el plano de simetría que en el grafo anterior eran puntos medios de aristas, y si es necesario las aristas que los conectan, al mismo tiempo que se eliminan las aristas que conectan las dos mitades simétricas.

En contra de lo que se podría pensar, la información que se puede obtener de las imperfecciones del grafo inicial no se queda reducida por la eliminación de media figura. El método que se utiliza para encontrar los planos de simetría ([PCM03] y § 6.2) construye los polígonos de simetría en función de los dos lados de la figura, situando sus vértices en puntos medios de aristas o en los mismos vértices del grafo inicial. Esto hace que el polígono de simetría lleve consigo la media de los dos lados, y absorba las posibles imperfecciones que arrastran los bocetos iniciales. Así la información obtenida puede ser entendida como el promedio de ambos lados.

### 6.10.1. La función de simetría 3D.

La simetría en el espacio tridimensional se puede realizar mediante composición de transformaciones. El proceso consiste en convertir a la recta  $E$  (recta perpendicular al plano) en el eje  $Z$  y al plano de simetría en el plano  $XY$ , hacer una simetría respecto a este plano  $XY$ , y finalmente devolver al plano  $XY$  a su posición inicial.

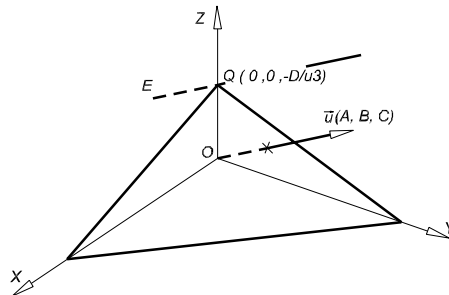


Figura 6-24. Sistema de Referencia para la transformación de Simetría

Siendo  $(a, b, c)$  el vector unitario del plano de simetría que corresponde a la ecuación  $ax+by+cz+d=0$ , las siete transformaciones que se necesitan para el cálculo de las coordenadas simétricas son:

- Una traslación del punto  $Q$  al origen  $O$ :

$$T_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & d/c & 1 \end{bmatrix}$$

- Un giro de eje  $X$  y ángulo  $N$ . (Ecuación 5-18, pág. 129)
- Un giro de eje  $Y$  y ángulo  $-L$ . (Ecuación 5-19, pág. 130)
- Una simetría respecto del plano  $XY$ :

$$S_{XY} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Un giro de eje  $Y$  y ángulo  $L$ .
- Un giro de eje  $X$  y ángulo  $-N$ .
- Una traslación que devuelva el origen  $O$  a la posición inicial del punto  $Q$ .

Así la composición de las siete transformaciones da como resultado:

$$S = T_i \cdot G_{X,N} \cdot G_{Y,-L} \cdot S_{XY} \cdot G_{Y,L} \cdot G_{X,-N} \cdot T_i^{-1} = \begin{bmatrix} 1-2a^2 & -2ab & -2ac & 0 \\ -2ab & 1-2b^2 & -2bc & 0 \\ -2ac & -2bc & 1-2c^2 & 0 \\ -2da & -2db & -2dc & 1 \end{bmatrix}$$

Ecuación 6-8

Si se tiene la mitad del modelo reconstruido y el plano de simetría identificado, aplicando a sus vértices la transformación  $S$ , obtenemos el modelo completo también reconstruido.

Realmente, como el proceso de inflado empleado para reconstruir garantiza que el modelo final mantiene las mismas coordenadas  $x$  e  $y$  que los vértices del dibujo original, únicamente se necesita hacer el cálculo de la coordenada  $z'$  de los vértices "reflejados". Siendo  $(x, y, z)$  las coordenadas de su vértice simétrico el cálculo a realizar queda en:

$$z' = -2ac \cdot x - 2bc \cdot y + (1 - 2c^2)z - 2dc$$

Ecuación 6-9

## 6.11. RESULTADOS

---

Un boceto con un único plano de simetría se utiliza para ilustrar los resultados de los diferentes métodos expuestos anteriormente. Inicialmente el dibujo mostrado en la Figura 6-25 es automáticamente limpiado o mejorado y el polígono de simetría que representa su plano de simetría es detectado.

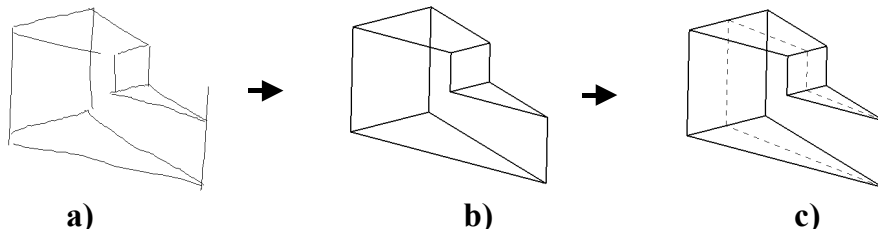


Figura 6-25. a) Boceto inicial, b) dibujo lineal, c) polígono de simetría detectado.

Si la regularidad de simetría es formulada únicamente en términos de planicidad del plano de simetría, como si se tratase de una cara plana más del modelo (§6.4.1), el resultado es un modelo con un grado de planicidad del plano de simetría muy alto, pero que no posee la forma deseada. En la Figura 6-26 se muestra el modelo en proyección axonométrica junto con tres vistas diédricas del modelo obtenido. Además, la condición se cumple para el modelo con todas las coordenadas  $z$  iguales a cero.

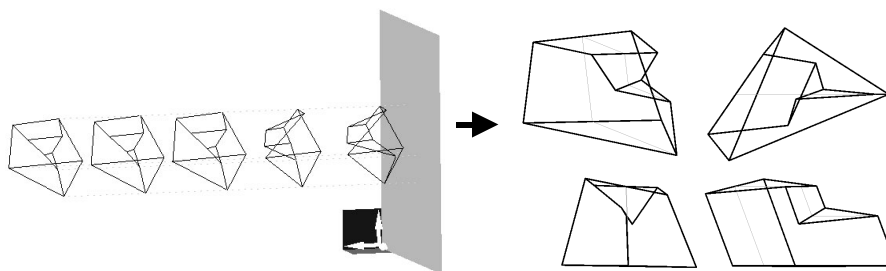


Figura 6-26. Modelo obtenido utilizando únicamente la condición de planicidad del plano de simetría como regularidad.

Esto mismo ocurre si se exige la ortogonalidad (§6.4.3) entre el plano de simetría y las aristas que cruzan el plano de simetría (ver Figura 6-27). Cuando se exigen las dos condiciones, la función objetivo ofrece resultados patológicos que corresponden a mínimos locales de la función objetivo, como se puede observar en la Figura 6-28.

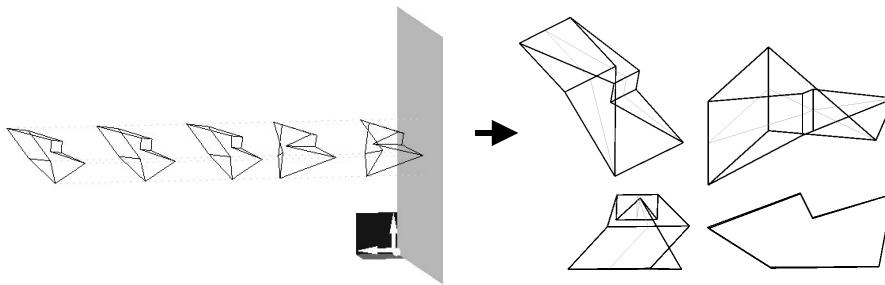


Figura 6-27. Modelo obtenido utilizando únicamente la condición de ortogonalidad de líneas que cruzan el plano de simetría con el propio plano como regularidad.

En los tres casos se ha utilizado la optimización Hill-Climbing. Además, para escapar de la solución trivial, aquella en la que todas las coordenadas  $z$  son iguales a cero, se realiza un *inflado previo aleatorio* [3.4]. La forma final del modelo depende en gran medida del modelo de partida obtenido aleatoriamente, es por ello que los resultados obtenidos en repetidas pruebas son similares pero no exactamente iguales a los mostrados en las figuras.

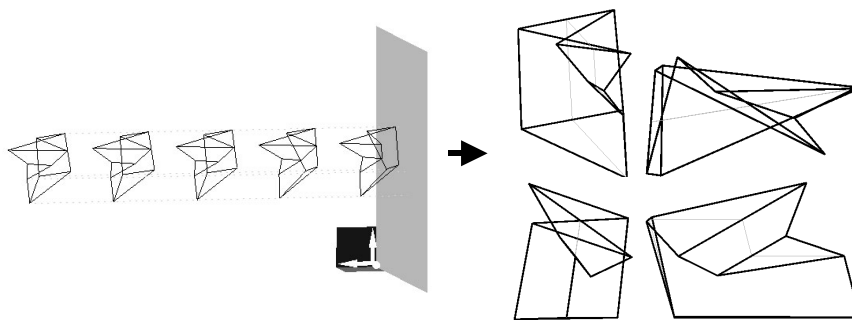


Figura 6-28. Modelo obtenido utilizando simultáneamente la planicidad del plano de simetría y la ortogonalidad entre las aristas que cruzan el plano de simetría y el propio plano de simetría como regularidad.

## Percepción Artificial de Dibujos Lineales

---

Resultados mucho mejores son los obtenidos utilizando la regularidad de simetría bilateral descrita en la sección 6.9 (Ver Figura 6-29). Además, no se necesita la utilización de un inflado previo, puesto que la regularidad de simetría en el modelo no se satisface trivialmente en la imagen de partida. La regularidad está formulada como una función cuadrática (ver Ecuación 6-6, pág. 160), lo que hace que la convergencia sea más rápida que una función lineal.

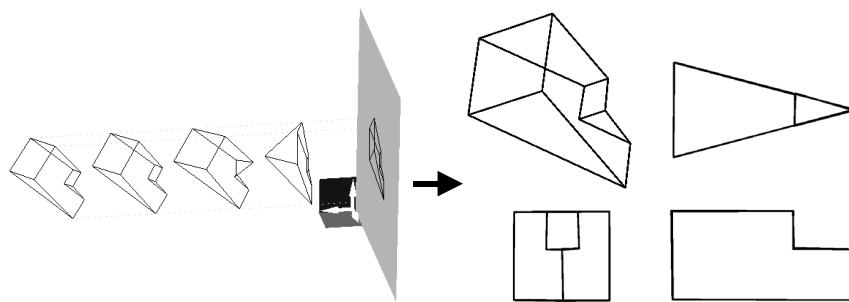


Figura 6-29. Modelo obtenido utilizando la regularidad de simetría bilateral como único método de inflado.

Además, se puede concluir diciendo que ésta es una *regularidad verdadera*, puesto que definimos como “verdaderas” a aquellas regularidades cuya utilización depende de la detección de características que posee la imagen de partida. Es decir, únicamente se utiliza para reconstruir aquellos dibujos lineales en los que se detecta la condición de simetría en el dibujo de entrada.

La simetría resuelve por sí sola la reconstrucción para los bocetos simétricos siempre que el plano o los planos de simetría se hayan detectado correctamente. Es decir, puede ser utilizada como única regularidad para conseguir la reconstrucción de bocetos en modelos 3D, sin necesidad de la utilización de ningún método de inflado adicional. Así, esta regularidad puede ser considerada como un nuevo método de inflado por sí misma.

De este modo tampoco necesita para la correcta reconstrucción de modelos de otras regularidades, posiblemente conflictivas. Esto hace de la función objetivo una función mucho más sencilla que la combinación de muchas regularidades que hace de ella una función con numerosísimos mínimos locales de los cuales es difícil salir en el proceso de optimización. La simplicidad en la función a optimizar hace del proceso de reconstrucción una herramienta computacionalmente más eficiente, reduciendo los tiempos de cálculo.



Capítulo 6. Simetría de Modelos 3D detectados en el Espacio 2D

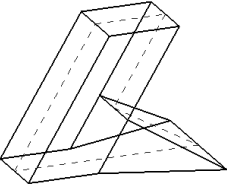
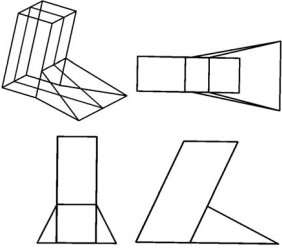
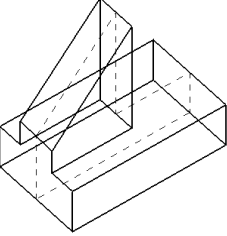
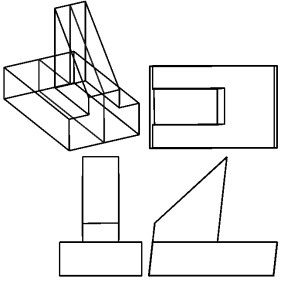
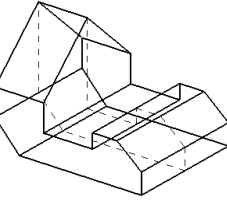
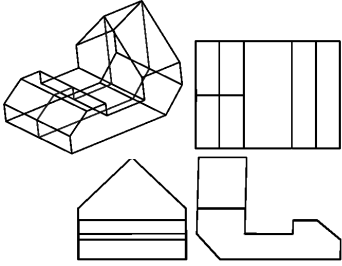
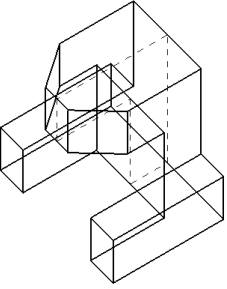
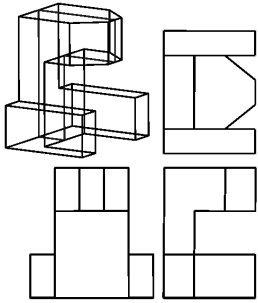
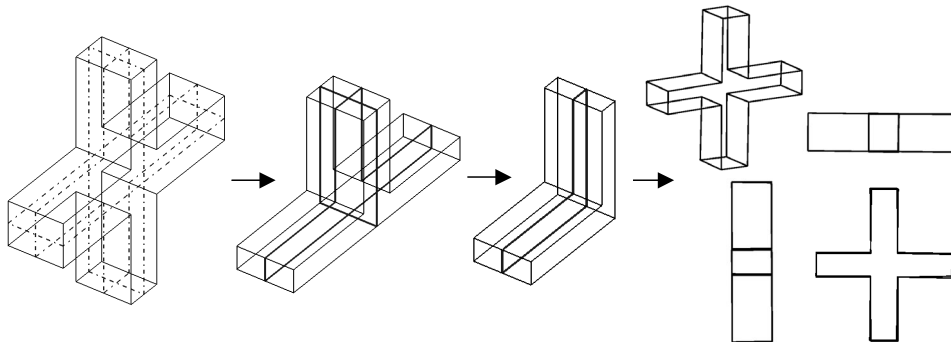
Dibujo Lineal		Modelo 3D	Proceso
	19 aristas 12 vértices		9 caras 1 plano de simetría Tiempo de reconstrucción: menor a 1''
	24 aristas 16 vértices		10 caras 1 plano de simetría Tiempo de reconstrucción: menor a 1''
	33 aristas 22 vértices		13 caras 1 plano de simetría Tiempo de reconstrucción: menor a 1''
	46 aristas 30 vértices		18 caras 1 plano de simetría Tiempo de reconstrucción: 2''

Tabla 6-1. Ejemplos simétricos reconstruidos con la regularidad de simetría bilateral.

## Percepción Artificial de Dibujos Lineales

---

En la Tabla 6-1 se ofrecen algunos ejemplos que muestran el comportamiento de este método. Cuatro ejemplos que muestran la forma obtenida tras la aplicación exclusiva de esta regularidad.

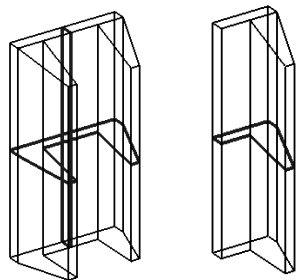


*Figura 6-30. Modelo con tres planos de simetría.*

Por último se da un ejemplo de una forma más compleja para mostrar los beneficios de la reducción del dibujo a la mitad. La Figura 6-30 muestra un dibujo lineal cuyo modelo posee tres planos de simetría. Dos de ellos son utilizados para reducir el tamaño del problema, y un último plano se utiliza para la reconstrucción del modelo parcial utilizando como método de reconstrucción la simetría bilateral. Finalmente el modelo completo es restaurado. Particularmente, en este ejemplo los tiempos de cálculo son reducidos de 3 segundos para la reconstrucción del dibujo completo a menos de 1 segundo si se reconstruye el modelo reducido.

Este nuevo método simplifica el proceso de reconstrucción mediante disminución del número de variables utilizadas en los cálculos, y tiene como una de sus consecuencias principales la reducción del tiempo empleado en el proceso completo de reconstrucción.

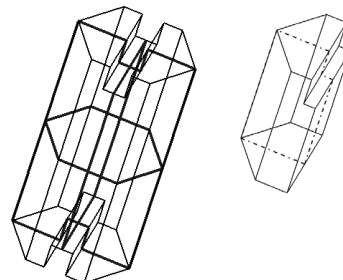
La aplicación de la simetría bilateral ofrece muy buenos tiempos de cálculo debido a la simplicidad de la función objetivo. Cuando el número de planos de simetría va aumentando la complejidad de la función objetivo aumenta con él. Así que los modelos testados han revelado que es aconsejable dejar a la simetría bilateral como única regularidad cuando sólo se tiene un plano de simetría. Cuando la simetría es múltiple es más eficiente reducir el modelo hasta que el mismo sólo posee un plano de simetría y en ese momento aplicar la regularidad de simetría, tal y como expone en el método de reconstrucción partiendo de la mitad simétrica.



a)

b)

Figura 6-31. Figura con dos planos de simetría  
Tiempo de reconstrucción partiendo de la  
imagen a) 3" y de la imagen b) menos de 1"



a)

b)

Figura 6-32 Figura con tres planos de simetría  
Tiempo de reconstrucción partiendo de la imagen  
a) 6" y de la imagen b) menos de 1"

Hay que destacar que los modelos que pueden ser construidos mediante extrusión son una excepción a esta regla. Estos tienen (al menos) un plano de simetría (ortogonal a la dirección de extrusión), pero la mitad del dibujo construido desde él tiene exactamente el mismo número de vértices, aristas y caras que el dibujo original (ver Figura 6-33). El dibujo encontrado es topológicamente exacto al anterior incluyendo también el plano de simetría. Por tanto, una comprobación adicional debe ser realizada para los modelos de extrusión para prevenir este problema. Esta comprobación consiste simplemente en comparar el número de vértices antes y después de la reducción. En el caso de que este plano sea el único plano de simetría del objeto, no tiene sentido activar la reducción. En el caso de que existan otros planos de simetría que no sean "de extrusión", deberían ser ellos los elegidos para la reducción, dejando el plano "de extrusión" para aplicar la regularidad de simetría bilateral.

Algunos otros modelos, aunque no son de extrusión también presentan la misma peculiaridad. El ejemplo mostrado en la Figura 6-34 se observa como la mitad de la pirámide es topológicamente igual al modelo que representa la figura inicial. Por lo tanto en sucesivas reducciones del grafo se obtendrán grafos equivalentes que no simplificarán el modelo ni la función objetivo. Pero como ya se ha dicho anteriormente, para la detección automática de todos estos casos (incluidos los construidos mediante extrusión) basta con comprobar el número de entidades en el grafo original y en el reducido.

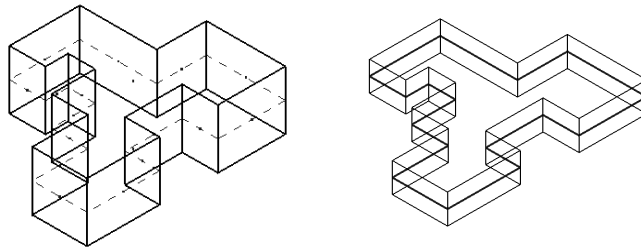


Figura 6-33. Detección de planos de simetría en figuras correspondientes a modelos extruidos.

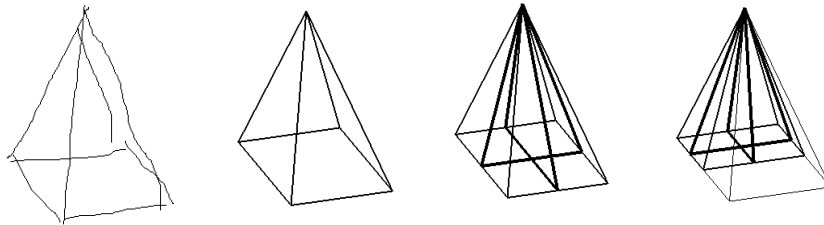


Figura 6-34. Modelos topológicamente iguales antes y después de eliminación de la mitad simétrica.

## 6.12. CONCLUSIONES

---

En este capítulo se ha realizado una nueva aportación dentro del campo de la detección de planos de simetría oblicuos de modelos poliédricos representados mediante dibujos lineales bidimensionales.

Se han presentado dos ideas novedosas para la reconstrucción de objetos. La primera consiste en aplicar la simetría como una regularidad en una reconstrucción basada en la optimización. De hecho, se demuestra que la simetría es adecuada para reconstruir modelos por sí misma, sin la necesidad de utilizar ninguna otra técnica de inflado previo u otras regularidades.

Los resultados obtenidos en las pruebas realizadas mediante la aplicación informática REFER se caracterizan por unos tiempos muy reducidos en la ejecución y un muy pequeño ratio de fallo. Sin embargo, la conclusión principal es que la regularidad de simetría bilateral oblicua permite reconstruir objetos que hasta ahora no se podían reconstruir en modo alguno; o se podían reconstruir sólo después de una cuidada selección interactiva de las

regularidades apropiadas y de los parámetros de optimización correctos. Por lo que concluimos que representa un paso más hacia la reconstrucción tridimensional.

Una segunda idea es la eliminación de la información duplicada en los modelos simétricos antes de la reconstrucción. Después de la eliminación de la mitad del dibujo, se puede realizar la reconstrucción del dibujo simplificado mediante cualquiera de los métodos de reconstrucción conocidos (y por lo tanto también con el primer método aquí explicado). El objetivo de esta técnica es disminuir el tiempo de cálculo. Este objetivo se cumple en un gran número de ejemplos, aunque no para extrusiones o algún otro caso particular, como se expone en la sección 6.11.

En definitiva, se ofrecen nuevos métodos para la reconstrucción de sólidos poliédricos partiendo de bocetos planos basados en la detección de planos de simetría bilateral en bocetos, que han sido defendidos en un congreso reciente [PCM03] y su utilización en la reconstrucción, que también ha sido defendida en un congreso reciente [PMC03]. Su principal limitación es que el funcionamiento de los métodos de reconstrucción originales expuestos en este capítulo depende estrechamente del método de detección de los planos de simetría en el boceto ( §6.3 y [PCM03] ). Dos aportaciones complementarias son la identificación de los dos lados simétricos en la representación 2D y el emparejamiento de vértices explicado en este capítulo, que son requeridas por los métodos expuestos que necesitan esta información.

En resumen, el uso de la simetría es un método útil que permite inflar formas complejas sin introducir un conjunto de regularidades heterogéneas grande que potencialmente pueden ser conflictivas. Tampoco requiere un inflado previo para escapar del mínimo local del plano. Además, ha sido utilizado con éxito para la reducción recursiva del tamaño del problema.

Finalmente se debe hacer notar que se necesita trabajo futuro para analizar los casos de simetría rotacional.



# Capítulo 7. Conclusiones y Desarrollos Futuros

---

## 7.1. CONCLUSIONES

---

La reconstrucción tridimensional de modelos virtuales a partir de bocetos planos es un campo de indudable interés para la fase de diseño conceptual. En esta tesis se ha realizado una revisión amplia del estado del arte de esta disciplina, y se ha llegado a la conclusión de que la introducción de conceptos provenientes de la teoría de la percepción, es sin duda necesaria. La reconstrucción de objetos poliédricos a partir de bocetos pseudo-axonométricos mediante técnicas de optimización matemática basada en regularidades se considera una de las técnicas más prometedoras. Se ha implementado y después de detectar algunos problemas propios de esta técnica, se han hecho contribuciones encaminadas a mejorarlos. También se ha desarrollado la regularidad de simetrías del modelo, describiendo el modo de detectarla y el de utilizarla para reconstrucción. Para ello se han debido hacer aportaciones en el campo de la detección de caras en los dibujos bidimensionales y en la detección de ejes de simetría oblicua en las mismas. El interés de estas aportaciones queda validado por las publicaciones científicas a que ya han dado lugar y que se reseñan al final del presente capítulo.

En definitiva, las aportaciones de esta tesis han mejorado los resultados obtenidos anteriormente tanto con la técnica de reconstrucción mediante optimización como con otras técnicas de reconstrucción. Además, la metodología desarrollada se ha adaptado muy bien a un entorno de “diseño mediante bocetos” que el grupo REGEO está desarrollando en la actualidad [NCC03]. Por ello, los desarrollos futuros que se proponen son realizables a corto plazo, y la contribución de esta tesis es fundamental para ello.

### 7.1.1. Reconstrucción tridimensional

Reconstruir poliedros mediante optimización es más eficiente cuando la función objetivo es más sencilla, y de ella se eliminan todos los requisitos que no corresponden con características reales del modelo. El cúmulo de regularidades falsas en la función objetivo hace lento el proceso de optimización, y lo complica aumentando el número de mínimos locales en ella, de los cuales es difícil salir en el proceso de reconstrucción. La simplicidad en la función objetivo a optimizar hace del proceso de reconstrucción una herramienta computacional más eficiente, reduciendo los tiempos de cálculo.

En mi opinión, para que las técnicas de optimización puedan hacer de la reconstrucción tridimensional un método eficiente hay que elegir de modo eficiente las regularidades en cada caso particular.

Por lo tanto hay que poder detectar el número máximo de regularidades verdaderas y elegir correctamente aquellas que se cumplen en el boceto. Pero el campo de la detección de regularidades y su formulación e interpretación, -al que la presente tesis ha hecho alguna modesta aportación, - está abierto a estudio, y aún quedan muchas regularidades por detectar y formular.

Por todo lo visto, es interesante seguir trabajando en las regularidades para poder cambiar las antiguas falsas regularidades, por otras verdaderas y fiables.

Esta tesis ha contribuido al estudio de las regularidades ofreciendo una formulación más rigurosa de algunas de las ya estudiadas por otros autores, y aportando una nueva: la simetría bilateral del modelo.

### 7.1.2. Detección de Caras

Un punto importante en la reconstrucción de bocetos en modelos poliédricos y también en cualquier técnica de reconocimiento de figuras 2D, así como para la correcta formulación de muchas regularidades, es la detección de caras en los bocetos. El conocimiento de las caras ofrece casi tanta información como el conocimiento de los nodos y las aristas del grafo. Además de ser necesaria para aplicar muchas de las regularidades, las caras pueden ser utilizadas para la introducción de restricciones en el proceso reconstructivo y así seguir un orden en la aplicación de las mismas como describe Sugihara [Sug99].

En el método empleado en esta tesis, su detección es totalmente independiente de la información geométrica relativa en las uniones de aristas, por lo que se puede aplicar el



mismo método tanto para superficies curvas como planas. Las aportaciones encaminadas principalmente a mejorar la eficiencia del proceso de detección de caras, y a resolver algunos casos particulares que anteriormente no podían ser resueltos, o que requerían cálculos costosos que los hacían inviables para una aplicación interactiva, han sido presentadas y justificadas.

### **7.1.3. Regularidades. Simetría facial y Simetría bilateral del modelo**

La correcta formulación de las regularidades es crucial en el método de reconstrucción aquí elegido. En esta tesis se presenta una formulación rigurosa de la simetría facial oblicua, así como de la detección de todos sus ejes de simetría. De este modo se ha comprobado la detección de los ejes de simetría en todas las caras del modelo y se ha comprobado con respecto a los mismos la simetría de las caras en el modelo tridimensional.

La simetría de las caras del modelo es una regularidad importante, y la misma nos lleva a otra regularidad: la simetría del modelo. La detección de esta característica en el dibujo 2D no es obvia en absoluto. Para ello se aprovecha el cálculo de los ejes de simetría facial oblicua y se busca el polígono de simetría que es el que realmente desvelará si existe simetría en el modelo. Se ha detallado como detectarla y como aplicarla en reconstrucción. Además, la propuesta se ha validado con una implementación y unas pruebas que también se han descrito. La conclusión es que esta regularidad, que no se cumple trivialmente en el plano 2D, ofrece un fuerte potencial. La utilización de la regularidad de simetría del modelo hace de la reconstrucción mediante optimización un proceso más rápido debido a la mayor simplicidad de la función objetivo, resultado de la eliminación de otras regularidades posiblemente conflictivas. Esto también conlleva una reducción de los mínimos locales, y con ello disminuye la probabilidad de caer en uno de ellos y aumenta la probabilidad de obtener los modelos finales deseados o muy próximos a ellos.

Además, en los casos en los que se detecta más de un plano de simetría se ha comprobado que es aconsejable reducir la figura (eliminando una mitad) de modo recursivo hasta que sólo posea un plano de simetría. El procedimiento utilizado para la reducción del boceto también se ha detallado. El boceto reducido se reconstruye con la regularidad de simetría, aplicada al único plano de simetría remanente, y posteriormente partiendo del modelo reducido se reconstruye el modelo completo correspondiente al boceto inicial.

La evaluación de la técnica en un conjunto amplio de ejemplos ha mostrado resultados muy satisfactorios. Pero, además, la detección de simetría ha abierto la puerta a nuevas técnicas, distintas de la reconstrucción basada en optimización, que se describen en los desarrollos futuros.

### 7.1.4. Aplicación Docente

De la construcción de una aplicación informática con la que constatar resultados y validaciones, surgió la idea de completar una versión docente que llevar a las aulas ( Ver Anexo I).

Las asignaturas gráficas en los primeros cursos de ingeniería tienen como uno de sus primeros objetivos el desarrollo de la visión espacial de los alumnos. Para ello se utilizan diferentes técnicas mediante las cuales los alumnos deben proyectar modelos y reconstruir mentalmente modelos proyectados. Manejar dibujos bidimensionales de modelos es una tarea en un principio enrevesada. Sin embargo, manejar modelos con herramientas de visualización dinámicas y con representaciones realistas hace el trabajo mucho más sencillo.

Por otra parte la reconstrucción inmediata de los bocetos, hace de la herramienta una fuente de autoaprendizaje en los alumnos rápida y fácil de utilizar.

Así esta aplicación docente se espera que ofrezca buenos resultados en el aula, y ayude al aprendizaje de los alumnos de forma eficiente.

## 7.2. DESARROLLOS FUTUROS

---

Dados los buenos resultados que la simetría bilateral ha dado en la reconstrucción de poliedros, parece lógico pensar que sería interesante también detectar bocetos correspondientes con modelos que posean simetría rotacional. Asimismo se necesitaría realizar un estudio de simetría facial oblicua que incluyese la simetría facial oblicua rotacional.

Dentro de la detección de la simetría bilateral del modelo, sería interesante mejorar la detección de los planos de simetría. Para ello sería necesario el desarrollo de un método que permitiera calcular todos los circuitos cuando los ejes son añadidos al grafo. Esto

podría aumentar la eficacia de la detección de los planos y en consecuencia en la reconstrucción tridimensional de modelos simétricos.

Otro desarrollo futuro bastante obvio es ampliar el universo de objetos reconstruibles. Por una parte hay que considerar los cuerpos poliédricos más complejos (agujeros, etc.) y por otra parte hay que introducir superficies curvas.

También es imprescindible, y se ha apuntado en esta tesis, implementar métodos de optimización matemática más eficientes. Entendiendo que la eficiencia supone aumentar el porcentaje de éxito en la reconstrucción, y, al mismo tiempo, reducir los tiempos de cálculo.

Por último, otros desarrollos futuros suponen nuevas líneas de trabajo, que los conocimientos adquiridos en esta tesis hacen viables a corto plazo. Entre ellas, destacamos la propuesta de reconstruir utilizando motores geométricos, que es objeto de dos proyectos de investigación en los que actualmente participo:

- “Del boceto al modelo: Nuevas interfaces de usuario para sistemas CAD. Código del Proyecto: P1-1B2002-08. *UJI-Fundació Caixa Castelló. Programa de fomento de proyectos de investigación 2002. Modalidad B. Del 15/12/02 al 15/12/04.* Investigador Principal: Dr. D. P. Company.
- “Interfaces Caligráficas para Herramientas Avanzadas de Diseño” Código del Proyecto: CTIDIB/2002/51. *Generalitat Valenciana. Ayudas para Proyectos de I+D para el ejercicio 2002, modalidad B. Del 01/01/02 al 31/12/03.* Investigador Principal: Dr. D. M. Contero.

### 7.2.1. Modelos representados mediante grafos no-conexos

Una mejora sin duda deseable sería la ampliación de los tipos de modelos con los que trata el reconstructor REFER. Por una parte su ampliación a aquellos modelos cuya representación puede tratarse como la representación de dos grafos no-conexos. Por ejemplo la Figura 7-1 y la Figura 7-2 muestran dos ejemplos sencillos de este caso. Para la reconstrucción empleada, cada uno de los dos grafos se consideraría como un modelo distinto, y las posiciones relativas entre ellos o las operaciones booleanas que existan entre ellos tampoco quedan definidas.

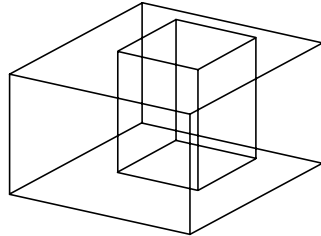


Figura 7-1. Modelo con un agujero pasante.

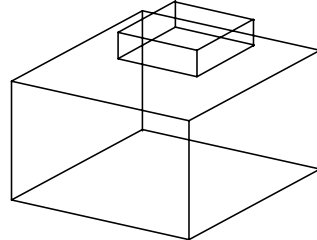


Figura 7-2 Modelo formado por dos bloques

Este tema es bastante complejo. Por ejemplo, perceptualmente la Figura 7-1 representa claramente un bloque con un agujero prismático pasante en su interior, y la Figura 7-2 representa un bloque mayor que tiene una protusión de un bloque más pequeño con la misma forma. Sin embargo, el problema no tiene una solución clara en la Figura 7-3: no es tan fácil determinar si el bloque de menor tamaño está sobre el mayor o es un agujero del mismo.

Para poder ampliar la reconstrucción a este tipo de dibujos, hay que dedicar un estudio intensivo a las características de este tipo de modelos y al mismo tiempo adaptar las herramientas, como por ejemplo las regularidades a las nuevas topologías.

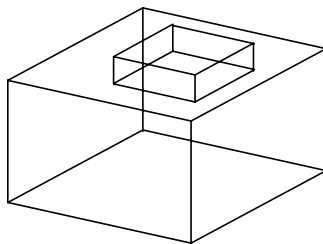


Figura 7-3. Modelo representado mediante dos grafos no-conexos

### 7.2.2. Modelos con Caras Curvas

Otra ampliación a otro tipo de modelos sería la aceptación de objetos con caras curvas. Existe un caso particular que se puede incluir fácilmente, porque todos los objetos con caras curvas que puedan representarse por medio de grafos isomorfos a los que

representan poliedros como los que se han tratado en esta tesis, pueden resolverse de la misma forma que los cuerpos poliédricos. No se han considerado estos cuerpos para la presente tesis, porque aun no hemos implementado la interfaz caligráfica que gestiona el bocetado de aristas y contornos curvos. Este es el único obstáculo, porque, para obtener el grafo isomorfo no es necesario eliminar la curvatura de sus aristas ni contornos, puesto que la geometría de esas aristas no afecta al proceso.

Lo que es necesario es incluir aquellas “aristas” que corresponden a las fronteras de aquellas superficies contiguas que no generan aristas en sentido estricto porque no determinan cambios bruscos en la dirección de la normal a la superficie. Son las producidas por transiciones “suaves”, del tipo de las que corresponden a conexiones tangentes. Es decir, cuando hay *continuidad tangencial* ( $C^2$ ), hay continuidad posicional, y, además, la dirección de la tangente es la misma para el punto común, tanto si se le considera como punto de una superficie, como si se le considera punto de la segunda superficie.

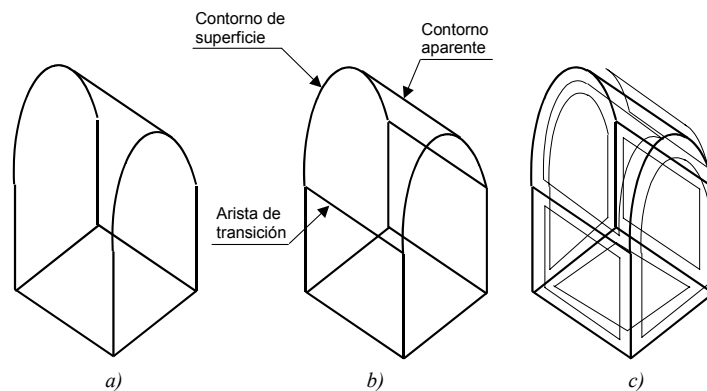


Figura 7-4. a) Modelo con superficies curvas representado mediante aristas y contornos;  
b) el mismo modelo con las curvas necesarias para delimitar caras y  
c) representación de las caras determinadas.

En definitiva, hay que incluir aquellas curvas que delimitan el contorno de las “caras” no planas (*curvas de contorno de superficie*), aquellas curvas que delimitan el contorno aparente del objeto (*curvas de contorno aparente*, o “silhouette” curves) y las curvas de transición entre superficies con continuidad  $C^2$  o superior (*curvas o aristas de transición*).

En el caso de superficies de simple curvatura, o superficies regladas, el procedimiento de añadir las curvas de contorno es suficiente para generar un grafo isomorfo al de un objeto

poliédrico topológicamente equivalente. Por tanto, el proceso de detección de caras se puede realizar del modo descrito anteriormente.

En los casos de superficies de doble curvatura, el procedimiento anterior da lugar a grafos incompletos y se necesitaría un estudio en mayor profundidad para conseguir su introducción en un programa de reconstrucción tridimensional.

### 7.2.3. PSO

La optimización en enjambre de partículas (PSO, o “Particle Swarm Optimisation”) fue introducida en 1995 por los trabajos desarrollados por Eberhart y Kennedy [EK95] y [KE95]. Los fundamentos del método pueden encontrarse en [Hu03].

En la búsqueda de un método de optimización que mejore los tiempos de cálculo y que permita escapar de mínimos locales, se cree que la filosofía de este método de optimización puede mejorar el algoritmo de optimización en estos aspectos.

Respecto a los tiempos de cálculo, es probable que la metodología PSO acelere la convergencia, en comparación con los métodos simples (que no utilizan información de gradientes), que son los que mayoritariamente se han empleado hasta la fecha para reconstrucción mediante optimización. Además, la metodología PSO parece apropiada para integrar todos los métodos de inflado previo desarrollados hasta la fecha por el grupo REGEO. Además, la metodología PSO supone una alternativa a los criterios heurísticos que empleamos hasta la fecha para elegir el método de inflado previo más apropiado para cada caso.

### 7.2.4. Motores Geométricos.

La forma de trabajar de los actuales programas de modelado está basado en un primer boceto que representa una sección plana con la que posteriormente se pueden realizar diferentes operaciones, como un barrido o una extrusión y así obtener un modelo 3D. En ese primer dibujo, el programa es capaz de detectar ciertas características como perpendicularidad y paralelismo entre líneas. Estos mecanismos se sirven de un gestor de modelos basados en restricciones paramétricas o variacionales para realizar estas detecciones [D-Cub].

Los modeladores geométricos tridimensionales tiene como principal objetivo crear y manipular modelos simbólicos de formas geométricas. Los modeladores tridimensionales

están implantados en mucho software comercial con el que se tiene que representar, mostrar o manipular formas tridimensionales. Sin embargo, el número de modeladores con kernel abierto está muy limitado.

La utilización de uno de estos gestores de modelos podría ayudar a la reconstrucción tridimensional. Así, el método propuesto parte de la idea de explorar las posibilidades que ofrecen los núcleos geométricos comerciales ACIS [ACIS], D-Cubed [D-Cub] y otros (Parasolid [Paras] y OpenCASCADE [OpenC]) para la gestión de modelos obtenido al reconstruir el futuro desarrollo del sistema. Administrador de modelos, concentrando nuestros esfuerzos en detectar, evaluar y formular regularidades que contengan la información de cómo el diseñador ve un modelo a través de un boceto.

La disponibilidad de dichos motores geométricos hace viable la exploración de dichas estrategias. En concreto, se ha utilizado la versión sin coste del programa para Universidades del motor geométrico “ACIS 3D geometric modeling” [Cor01], [ACIS]. ACIS suministra un núcleo de modelado sólido de tecnología sofisticada para algunas de las aplicaciones CAD más extendidas, tales como AutoCAD, Mechanical Desktop, e Inventor (Autodesk), CADdy++ Mechanical Design (Ziegler), CADKEY (CADKEY), GRADE/Cube II (Hitachi Zosen), IronCAD (Alventive), y Vellum Solids (Ashlar).

ACIS está escrito en C++ y consiste en un conjunto de clases y funciones en C++. Los desarrolladores de aplicaciones utilizan las clases y funciones para crear una aplicación que gestiona el modelo 3D. ACIS complementa las aplicaciones ofreciendo un entorno unificado para el modelado de curvas, superficies y sólidos.

ACIS es un modelador de tipo B-rep, lo que significa que define la frontera que separa al material (sólido) del espacio vacío. Esta frontera está construida con un conjunto cerrado de superficies. No obstante, ACIS integra modelos de alambre (“wireframe”), de superficies, y modelos sólidos permitiendo que coexistan en una estructura de datos unificada, la cual es implementada en una jerarquía de clases C++.

### **7.3. PUBLICACIONES RELACIONADAS CON ESTA TESIS**

---

Fruto de esta tesis se han obtenido hasta la fecha, las siguientes publicaciones:

- [PMC03] Piquer, A.; Martin, R.R.; Company, P. "Using skewed mirror symmetry for optimisation-based 3D line-drawing recognition" GREC'2003, Fifth IAPR International Workshop on Graphics Recognition, Barcelona, Spain. Proceedings of the Fifth IAPR International Workshop on Graphics Recognition (ISBN 84-932156-6-X), pp. 182-193, 2003.
- [PCM03] Piquer Vicent, A.; Company Calleja, P.; Martin, R.R. "Skewed Mirror Symmetry in the 3D Reconstruction of Polyhedral Models." The 11-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2003 (WSCG'03). *Journal of WSCG*. Vol. 11, No. 3, (ISSN 1213-6972), pp. 504-511. Plzen (República Checa), 2003.
- [PC 03] Piquer, A.; Company, P. "Simetría Facial Oblicua en Reconstrucción Tridimensional de Dibujos Lineales". *MetalUnivers*. (ISSN 1578-8873), nº 12, diciembre, Pág. 88, 2002.
- [PC 02] Piquer Vicent, A.; Company Calleja, P. "Simetría Facial Oblicua en Reconstrucción Tridimensional de Dibujos Lineales." XV Congreso Nacional de Ingeniería Mecánica. Actas del XV Congreso de Ingeniería Mecánica. Cádiz, 2002.
- [CCP03a] Company, P, Contero M., Piquer A., Aleixos N., Conesa J. y Naya F. "Aplicación docente de un programa de modelado 3D mediante bocetos axonométricos." XIII ADM – XV INGEGRAF International Conference on Tools and Methods Evolution in Engineering Design. Napoli, Salerno (Italia), 10 páginas, formato CD (resumen en página 28 del libro de resúmenes), 2003.
- [CC03b] Conesa, J; Company, P; Piquer, A. y Aleixos, N. "Ampliación del método de inflado axonométrico para la reconstrucción de poliedros." XIII ADM – XV INGEGRAF International Conference on Tools and Methods Evolution in Engineering Design. Napoli, Salerno (Italia), 10 páginas, formato CD (resumen en página 65 del libro de resúmenes), 2003.



# Glosario

---

**2-Variiedad (2-Manifold):** cuerpos en los que en cualquier punto de su superficie se puede construir un disco, de tamaño arbitrariamente pequeño, que sea topológicamente equivalente a un disco sobre un plano.

**Árbol (Tree) :** Grafo conexo y sin ciclos.

**Árbol CSG (CSG tree):** Se trata de una estructura jerárquica que representa la composición de un determinado *modelo* a partir de una serie de *primitivas* definidas y un conjunto de transformaciones. Cada rama del árbol representa un *primitiva* y cada *nodo* interno representa la operación de transformación a la que es sometido el *modelo* o *modelos* conectados al nodo.

**Árbol Expandido BFS (Breath-First Search):** Árbol cuya principal propiedad es que la distancia de cualquiera de sus vértices al vértice de partida es mínima.

**Árbol Generador (Spanning Tree):** Subgrafo que es *árbol* y contiene a todos los *vértices* del grafo.

**Arista (Edge):** Segmento de recta que define parte del contorno de un objeto. Una arista queda definida por la intersección de dos *caras* en un *modelo*. Pares no ordenados de *vértices*.

**Arista de Corte (Cut-Edge):** Arista que por sí misma desconecta el grafo.

**Arista Oculta (Hidden edge):** Cualquier *arista* del *modelo* que en la *proyección* dada (en la *imagen*) queda oculta por alguna de las *caras* del *modelo*.

**Arista Visible (Visible Edge):** Cualquier *arista* del *modelo* que en la *proyección* dada (en la *imagen*) no queda oculta por ninguna *cara* del *modelo*.

**Arreglo (Beautification):** Modificación realizada sobre el dibujo lineal para que sea una representación consistente.

**Articulación (Articulation-Vertex):** *Vértice de Corte.*

**Base de un Espacio de Circuitos (Base of a Circuits' Space):** Conjunto de *circuitos fundamentales* de un *grafo* cuya combinación genera todas las *caras potenciales* del *modelo*.

**Boceto (Sketch):** 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.

**B-Rep (B-Rep):** *Representación B-Rep.*

**Camino (Path):** *Paseo* en el cual cada *arista* y cada *vértice* se recorre una sola vez.

**Caminos Disjuntos (Edge-Disjoint Paths):** *caminos* que enlazan cualquier pareja de *vértices* sin tener ninguna *arista* en común.

**Cara (Face):** *Circuito cíclico* de un *grafo* que define una *cara* en el *modelo* que representa.

**Caras Potenciales (Potential Faces):** conjunto de *circuitos cíclicos* de un *grafo* que son candidatos a ser *caras* del *modelo* tridimensional.

**Centroide (Centroid):** Centro de Gravedad o Centro de Masas.

**Ciclo (Cycle):** *Camino* cerrado.

**Cifra de Mérito (Figure of Merit):** Valor cuantitativo que permite valorar, de la manera más objetivamente posible, todos los *modelos* posibles de acuerdo con un mismo criterio. En otras palabras, la cifra de mérito permite cuantificar la bondad de las distintas *soluciones* posibles.

**Circuito (Circuit):** *Ciclo*.

**Circuito Cíclico (Cyclic Circuit):** *Circuito simple* con la características de que el primer y último *nodo* son el mismo.

**Circuito Simple (Simple Circuit):** Se define como *circuito simple* a aquel *camino* en el que no se repite ningún *nodo* del *grafo*.

**Circuitos Fundamentales (Fundamental Circuits):** Conjunto de *circuitos cíclicos* tal que todos ellos poseen un *enlace* que no se encuentra en el resto de los *circuitos* y entre todos recorren todas los *enlaces* del *grafo*.

**Conectividad de Aristas (Edge-Connectivity):** Número mínimo de *aristas* que deben ser eliminados para obtener un *grafo* no conexo, o un *grafo* con una única *arista*.

**Conectividad de un Grafo (Connectivity):** número mínimo de vértices que deben ser eliminados para obtener un *grafo* no conexo, o un *grafo* con un único vértice.

**Conjunto de Corte (Cut-Set):** conjunto mínimo de *aristas* cuya eliminación desconecta un *grafo*.

**Contorno Aparente (Apparent Contour):** Es el conjunto de *enlaces* y *nodos* que representa el *circuito* exterior de un *grafo*. Desde el punto de vista de la geometría se define como la *proyección* sobre el plano de la imagen del cilindro o cono circunscrito a un *modelo*.

**Croquis (Sketch):** Dibujo que permite 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 (CSG):** Representación CSG.

**Curvas de Contorno Aparente (Silhouette Curves):** Curvas que delimitan el contorno aparente del objeto.

**Curvas de Contorno de Superficie (Boundary Curves):** Curvas que delimitan el contorno de las caras no planas.

**Curvas de Transición (Transition Curves):** Curvas de transición entre superficies con continuidad  $C^2$  o superior.

**Dibujo Lineal (Line Drawing):** Dibujo que expresa de forma estricta la geometría de una *imagen*.

**Diferencia Simétrica (Symmetric Difference):** Dado dos circuitos es el resultado de la unión de los dos conjuntos de aristas menos su intersección.

**Direcciones Principales (Main Directions):** Aquellas que se repiten con mayor frecuencia en una *imagen*.

**Eje Conjugado (Conjugate Axis):** Línea que pasa por el centroide de una figura simétrica y cuya dirección es la misma a la de correspondencia entre los puntos de una figura simétrica.

**Eje de Oblicuidad (Axis of Skew or Skewed Transverse Axis):** Eje conjugado.

**Eje de Simetría (Axis of Symmetry):** por un eje coplanario con una figura que la divide en dos partes, tales que a cada punto de una de las dos partes le corresponde un punto de la otra parte y se cumple que ambos puntos equidistan del eje y determinan una recta perpendicular al mismo.

**Ejes de Simetría Oblicua (Axes of Skewed Symmetry):** Eje de simetría de una simetría oblicua.

**Ejes Tentativos de Simetría Oblicua (Tentative Axis of Skew Symmetry):** Posibles ejes de simetría oblicua, que hay que testar aplicando la función inversa de oblicuidad para comprobar que lo son.

**Emparejamiento de Vértices Simétricos (Symmetrical Vertices Matching):** Técnica para encontrar en una figura 2D las parejas de vértices que en el modelo 3D correspondiente serían simétricos respecto a uno de sus planos de simetría.

**Enlace (Link):** Son las líneas que muestran las conexiones entre dos *nodos* distintos.

**Espacio de Circuitos (Space of Circuits):** Conjunto de *caras potenciales* definidas en un grafo.

**Espacio de Soluciones (Space of Solutions):** Conjunto infinito de vectores  $n$ -dimensionales que representan *soluciones* al problema de reconstrucción.

**Espacio del Gradiente (The Gradient Space):** Geométricamente corresponde con la proyección de los elementos del grafo dual  $(a_x, a_y, a_z)$  sobre un plano  $a_z = 1$ , con el centro de proyección en el origen de coordenadas del espacio dual y utilizando  $(0,0,1)$  como origen del espacio del gradiente.

**Etiquetado (Labelling):** Procedimiento basado fundamentalmente en tener un diccionario de uniones, de forma que se comparan las uniones del dibujo lineal con las del diccionario, y dependiendo del tipo de unión de la que se trate, las líneas que convergen en esta unión serán etiquetadas de un modo u otro.

**Etiquetado lógico (Logic Labelling):** etiquetado que asigna una configuración que esté comprendida en el diccionario de etiquetas, asegurando que las dos posibles uniones de un segmento produzcan la misma etiqueta para el mismo.

**Extensión Ortográfica (Orthographic Extension):** Conjunto infinito de objetos tridimensionales cuya proyección ortogonal coincide con el dibujo lineal dado.

**Extensión ortográfica (Orthographic Extension):** Conjunto infinito de *modelos* geoméricamente válidos obtenidos a partir de una *imagen*. Se considera que un *modelo* es válido si su *proyección* coincide con la *imagen* de partida.

**Figura Imagen (Image Figure):** Es la imagen obtenida como consecuencia de la sección producida por un plano a la *figura proyección*. Usualmente se denomina simplemente *imagen*.

**Figura Proyección (Projection Figure):** Es la figura obtenida como consecuencia de la proyección de un *modelo*. Usualmente se denomina simplemente *proyección*.

**Género de una Superficie (Genus):** Una superficie es de género  $g$  si es topológicamente equivalente a una esfera con  $g$  agujeros.

**Geometría Frontal (Frontal Geometry):** Geometría de la parte vista del objeto representado en la imagen.

**Grado de un Vértice (Degree of a Vertex):** número de *aristas* que concurren en un *vértice*.

**Grafo (Graph):** par  $G=(V,A)$ , donde  $V$  es un conjunto finito no vacío (a cuyos elementos se les llama *vértices*) y  $A$  es una familia finita de pares no ordenados de *vértices* de  $V$  (cuyos elementos son llamados *aristas*).

**Grafo Conexo (Connected Graph):** Un *grafo* en el que para cada par de *vértices*  $u$  y  $v$  existe un *camino* de  $u$  a  $v$ . Un *grafo* conexo siempre representa a un *poliedro euleriano*.

**Grafo  $k$ -conexo ( $k$ -connected):** si su *conectividad* es mayor o igual que  $k$ .

**Grafo Planar (Planar Graph):** Aquel que es isomorfo a un grafo plano.

**Grafo Plano (Plane Graph):** Grafo dibujado en un plano de forma que las *aristas* sólo se intersectan en los *vértices*.

**Grafos Isomorfos (Isomorphic Graphs):** Aquellos que poseen una correspondencia uno a uno entre sus *vértices*, con la propiedad de que el número de *arista* que unen cualquiera dos *vértices* de un *grafo* es igual al número de *aristas* que unen los *vértices* correspondientes en el segundo *grafo*. Es decir, mantienen la adyacencia.

**Grafos Isomorfos (Isomorphic Graphs):** grafos cuyos *vértices* tiene una correspondencia uno a uno, y además cumplen la propiedad de que el número de *arista* que unen

cualquiera dos vértices de un grafo es igual al número de aristas que unen los vértices correspondientes en el segundo grafo.

**Imagen (Image):** Figura imagen.

**Imagen Matricial (Bitmap Image):** Almacenamiento de una *imagen* mediante un mapa de píxeles.

**Imagen Vectorial (Vectorial Image):** Almacenamiento de una *imagen* mediante elementos primitivos tales como puntos, líneas y curvas.

**Mano Alzada (Freehand):** Estilo de dibujo sin ayuda de instrumentos de delineación. La ausencia de instrumentos hace que no se pueda garantizar la validez geométrica de lo dibujado. Por tanto, son dibujos que no respetan las leyes de la geometría métrica.

**Mapa de Bits (Bitmap):** Representación de una *imagen* discretizada como una matriz rectangular de elementos pictóricos simples.

**Matriz de Circuitos (Cycle Matrix):** Matriz donde se registra la relación entre las aristas que pertenecen a cada circuito, de forma que  $c(i, j) = 1$  si la arista  $j$  es una de las que componen el circuito  $i$ , y cero en caso contrario.

**Matriz de Ejes (Axis Matrix):** Matriz cuyo número de columnas coincide con el número de ejes de simetría facial oblicua detectados y en ellas se almacenan los datos que de ellos se conocen.

**Matriz de Incidencia (Incidence Matrix):** Matriz donde cada fila y cada columna representa a un vértice y el contenido de cada posición  $x_{ij}$  será la unidad cuando entre el vértice  $i$  y el  $j$  exista en el grafo un enlace que los una, y cero en caso contrario.

**Matriz de Interacciones (Interaction Matrix):** Matriz donde cada fila y cada columna representa un circuito y el contenido de cada elemento  $(i, j)$  es el número de aristas que tienen en común los circuitos  $i$  y el  $j$ . Y en los elementos de la diagonal se guarda número de aristas que contiene cada circuito.

**Método de Inflado (Inflation Approach):** Consiste en mantener las coordenadas (X,Y) de todos los vértices del *modelo* iguales a las coordenadas del correspondiente *nodo* de la *imagen*. Las coordenadas Z de los vértices del *modelo* constituyen las *variables de reconstrucción*.

**Método de Inflado Aleatorio (Random Inflation Method):** Método de inflado que consiste en dar a cada vértice una coordenada  $z$  aleatoria.

**Modelo (Model):** Representación tridimensional de un objeto. Todos los vértices que definen el *modelo* tienen asignadas coordenadas  $x$ ,  $y$  y  $z$ .

**Modelo Alámbrico (Wire-Frame):** Representación en la que las paredes del objeto se consideran transparentes.

**Modelo B-Rep (B-Rep Model):** Representación B-Rep.

**Modelo origami (Origami model):** Modelo no sólido. Las paredes que definen el contorno del *modelo* se suponen de espesor despreciable.

**Modelo Psicológicamente Aceptable (Psychologically plausible model):** Modelo obtenido a partir de una *imagen* mediante procesos de *reconstrucción* que concuerda con el *modelo* interpretado por ser humano cuando visualiza una *imagen*. En otras palabras, el *modelo* obtenido automáticamente mediante *reconstrucción* sería el mismo *modelo* que un ser humano diseñaría en un sistema CAD cuando visualiza una *imagen*.

**Nodo (Node):** Vértice.

**Normalon (Normalon):** Polígono que tienen la propiedad de que todos los ángulos entre dos *aristas* concurrentes son de  $90^\circ$ .

**Número Ciclomático (Cyclomatic Number):** Número de *circuitos fundamentales* de un *grafo* que queda definido por la expresión  $E-N+1$ ; siendo  $E$  el número de *enlaces* del *grafo* y  $N$  el número de *nodos* en el *grafo*.

**Objeto Poliédrico (Polyhedral Object):** Modelo geométrico limitado por regiones planas. Cada una de dichas regiones identifica una *cara* del poliedro. Una *arista* queda definida por la intersección de dos *caras* y un *vértice* queda definido por la intersección de dos o más *aristas*.

**Óptimo Global (Global Optimum):** Solución de un espacio  $n$ -dimensional donde la función objetivo alcanza su coste mínimo.

**Óptimo Local (Local Optimum):** Solución de un espacio  $n$ -dimensional donde la función objetivo alcanza un coste mínimo en un volumen más o menos extenso alrededor de la *solución*.

## Percepción Artificial de Dibujos Lineales

---

**Paseo (Walk):** Sucesión finita de *vértices* y *aristas*, donde se van enlazando sucesivamente *vértices* con *aristas*.

**Píxel (Pixel):** Elemento pictórico. Mínima unidad de representación. Cada *pixel* lleva asociado características tales como color, luminosidad y contraste.

**Planos Oblicuos de Simetría (Skewed Symmetry Planes):** planos representados mediante un polígono de simetría representando planos reales de simetría vistos desde un punto de vista desconocido.

**Poliedro (Polyhedron):** cuerpo geométrico limitado por un conjunto de polígonos planos, o caras.

**Poliedro Euleriano (Eulerian Polyhedrons):** 2-variedad de género cero. *Poliedro Simple*.

**Poliedro Simple (Simple Polyhedron):** 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.

**Poliedros Triédricos (Threehedral poliedra):** poliedros en cuyos vértices concurren exactamente tres caras.

**Polígono de Simetría (Symmetry Polygon):** es el polígono resultante de la sección de un modelo simétrico por su plano de simetría, formado por la unión de todas las aristas producidas por el corte.

**Politopos:** Subconjuntos del espacio euclídeo que se construyen con elementos rectos.

**Primitivas (Primitives):** Conjunto de entidades simple. Dichas *primitivas* se corresponden con puntos, rectas y curvas en una *Representación Brep* y con modelos definidos y parametrizados en *Representaciones CSG*.

**Proyección (Projection):** *Figura proyección*.

**Proyección Oblicua (Oblique Projection):** *Proyección paralela* en la que los rayos proyectantes no son perpendiculares al plano que contiene la *imagen*.

**Proyección Ortogonal (Orthogonal Projection):** *Proyección paralela* en la que los rayos proyectantes son perpendiculares al plano que contiene la *imagen*.

**Proyección Paralela o Cilíndrica (Parallel or Cylindrical Projection):** *Proyección* en la que los rayos proyectantes del *modelo* son paralelos a una dirección dada (centro de proyección impropio).



**Proyección Perspectiva o Central (Perspective or Central Projection):** Proyección en la que los rayos proyectantes del *modelo* pasan por un punto fijo (centro de proyección propio).

**Punto de Vista General (General Point of View):** Punto de vista arbitrario desde el cual se proyecta el *modelo* sin que existan coincidencias accidentales de líneas o vértices.

**Rango de Arista (Range of Edges):** Número de *caras* de las que forma parte una *arista*.

**Rango de Vértices (Range of Vertices):** Número de *caras* del que forma parte un *vértice*.

**Reconstrucción Geométrica (Geometrical Reconstruction):** Determinación de las relaciones topológicas y geométricas de todas las partes atómicas de un objeto a partir de una o varias *imágenes* planas.

**REFER (REFER):** Software de reconstrucción tridimensional basada en optimización en el que se han probado los resultados de esta tesis.

**REGEO (REGEO):** Grupo de investigación de Reconstrucción Tridimensional de Sólidos.

**Reglas de Percepción (Perception Rules):** Conjunto de reglas que el ser humano utiliza intuitivamente en la interpretación de *imágenes* bidimensionales.

**Regularidad (Regularity):** Expresión analítica de las *reglas de la percepción* visual humana que rigen la interpretación de *dibujos lineales*.

**Regularidad Verdadera (True Regularity):** aquella regularidad cuya utilización está fundada en la existencia de la característica a la que hace referencia en la imagen 2D.

**Representación B-Rep (Boundary Representation):** Representación de un *modelo* por un conjunto finito de superficies delimitadas por un conjunto de *aristas*.

**Representación CSG (CSG Representation):** Representación de un *modelo* a partir de un conjunto de *primitivas*, una serie de transformaciones y un conjunto de operaciones booleanas.

**Señales o Características Perceptivas (Perceptual Cues):** Reglas sencillas que componen otras reglas más complejas, según las cuales se rige la percepción humana.

**Separador (Separating Set):** conjunto de *vértices* cuya eliminación produce un *grafo no conexo*, o un *grafo* con un único *vértice*.

**Simetría de Rotación o Simetría Polar (Rotational Symmetry):** una figura tiene simetría rotacional cuando se puede dividir en partes que pueden ser permutadas y se repiten a sí mismas giradas un ángulo constante.

**Simetría Especular, o Simetría Reflectiva (Reflectional Symmetry):** Simetría formada por un eje que divide a la figura en dos partes tales que a cada punto de una de las dos partes le corresponde un punto de la otra parte y se cumple que ambos puntos equidistan del eje y determinan una recta perpendicular al mismo.

**Simetría oblicua (Skewed symmetry):** Simetría cuya dirección de correspondencia entre puntos no es ortogonal al eje de simetría.

**Sistema de Referencia de Inflado (Inflation Reference System):** Sistema de referencia cuyos ejes  $X$  e  $Y$  se encuentran sobre el plano del dibujo 2D y el eje  $Z$  es perpendicular a éste, orientado según un sistema de referencia a derechas.

**Sistema de Referencia de Simetría (Symmetry Reference System):** Sistema de referencia cuyo origen se encuentra en el centroide del polígono de simetría, los ejes  $X$  e  $Y$  se alinean con dos de los ejes del sistema del modelo y el eje  $Z$  sale perpendicularmente al plano orientado según un sistema de referencia a derechas.

**Sistemas CAD (CAD systems):** Sistemas de diseño asistido por ordenador.

**Solución (Solution):** Vector  $n$ -dimensional que representa un valor de las *variables de reconstrucción*.

**Solución Trivial (Trivial solution):** *Solución* definida por el vector cuyas componentes son todas nulas (= 0).

**Subespacio de Soluciones (Subspace of Solutions):** 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 *soluciones*, ya que la representación completa resulta imposible.

**Topología Trasera (Backside topology):** topología de la parte no vista del objeto debido a que por ser un objeto opaco él mismo la está ocultando.

**Valencia (Valency):** *Grado*.

**Valor de Referencia (Reference Value):** Valor numérico tomado como referencia para medir las desviaciones de una determinada magnitud.

**Variables de Reconstrucción (Reconstruction Variables):** Conjunto de variables que permiten definir los distintos *modelos* reconstruidos a partir de una *imagen*. El método propuesto, las coordenadas “Z” de cada *vértice* del *modelo* constituyen las variables de reconstrucción.

**Vértice (Vertices):** Es el punto tridimensional definido por la intersección de dos o más *aristas*.

**Vértice de Corte (Cut-Vertex):** *Vértice* cuya eliminación produce un *grafo* no *conexo*.

**Visión Artificial (Artificial Vision):** *Visión por ordenador*

**Visión por Ordenador (Computer Vision):** Conjunto de todas las herramientas y estrategias empleadas para hacer “ver” a los ordenadores.



# Bibliografía

---

- [ACIS] <http://www.spatial.com>
- [AJM97] Alcañiz, M.; Juan M.C.; Monserrat, C., and Contero, J.G. "Algoritmo Robusto para la Discriminación Texto-Gráfico. Aplicación a la interpretación de planos de ingeniería digitalizados." *XI Congreso Nacional de Ingeniería Mecánica*, pp. 621-628, 1997.
- [Ald83] Aldefeld, B. "On Automatic Recognition of 3D Structures from 2D Representations" *Computer Aided Design* vol.15, No.2, pp.59-64, 1983.
- [AMW88] Alt, H.; Mehlhorn, K.; Wagener, H. & Welzl, E. "Congruence, Similarity, and Symmetries of Geometric Objects" *Discrete & Computational Geometry* No. 3, pp. 237-256, 1988.
- [And00] Andreu Hernández, J.V. "Desarrollo, implementación y prueba de un algoritmo de reconstrucción de objetos a partir de una representación axonométrica, utilizando técnicas de optimización" Proyecto Final de Carrera, Universidad Politécnica de Valencia, Junio de 2000.
- [AP00] Ablameyko, S. & Pridmore, T. "Machine Interpretation of Line Drawing Images: Technical Drawings, Maps and Diagrams" Springer. ISBN: 3-540-76207-8, 2000.
- [APG02] Aleixos, N., Piquer, A., Galmes, V. y Company, P. "Estudio comparativo de aplicaciones CAD de modelado". *Actas del XIV Congreso internacional de Ingeniería Gráfica* (ISBN 84-699-8559-0), pp. 91-100, 2002.

## Percepción Artificial de Dibujos Lineales

---

- [AR84] Aldefeld, B. and Richter, H. "Semiautomatic Three-Dimensional Interpretation of line Drawings" *Computer & Graphics*, vol.8, No4, pp.371-380, 1984.
- [AT95] Ah-Soon, C. & Tombre, K. "A Step Towards Reconstruction of 3-D CAD Models" *Proceedings of Third International Conference on Document Analysis and Recognition*, Montréal (Canadá), pp. 331-334, 1995.
- [Ata85] Atallah, M.J. "On Symmetry Detection" *IEEE Transactions on Computers*. Vol. C-34, No. 7, Julio, pp. 663-666, 1985.
- [BCN94] Branco, V. Costa, A, and Nunes,F. "Sketching 3D models with 2D interaction devices" *Eurographics 94*, Vol. 13, No. 3, pp. C-489-C-502, 1994.
- [Ber96] G.R. Bertoline, E.N. Wiebe, C.L. Miller y L.O. Nasman. *Fundamentals of Graphics Communication*. Irwin Graphics Series, 1996. (*Dibujo en Ingeniería y Comunicación Gráfica*. Ed. Mc Graw Hill, 1999)
- [BET95] Bülthoff, H. H.; Edelman, S.Y.; Tarr, M.j. "How are three-dimensional objects represented in the brain?" *Cerebral Cortex*, vol. 5, pp. 247-260, 1995.
- [BT81] Barrow, H.G. & Tenenbaum, J.M. "Interpreting Line Drawings as Three-Dimensional Surfaces" *Artificial Intelligence*, Vol.17, pp. 75-116. August, 1981.
- [BW91] Braid, L. & Wang, P."Three-dimensional object recognition using gradient descent and the universal three-dimensional array grammar." *Proc. of SPIE. Conference on Intelligent Robots and Computer Vision X: Algorithms and Techniques*, Vol. 1607, pp. 711-718, 1991.
- [BW95] Baird, L. and Wang, P. "3D Object Perception Using Gradient Descent." *Int. Journal of Mathematical Imaging and Vision (IJMIV)*, Vol. 5, pp. 111-117, 1995.
- [BW96] Brown, E.W. & Wang, P.S. "Why We See 3-Dimensional Objects (Recovery From 2D): Another Approach". *Proceedings of SPIE'95, Robotic Vision*, 1996.
- [BZ 98] Bloomenthal K., Zeleznik R.C. et al. "SKETCH-N-MAKE: Automated machining of CAD sketches." *Proceedings of ASME DETC'98*, pp. 1-11, 1998.

- [CB 86] Courter, S.M.; Brewer, J.A. "Automated conversion of curvilinear wire-frame models to surface boundary models; a topological approach " *International Conference on Computer Graphics and Interactive Techniques*, pp.171-178, 1986.
- [CC 94] Cham, T.J.; Cipolla, R. "A local approach to recovering global skewed symmetry" *Proceedings of the International Conference on Pattern Recognition*. Pp. 222-226, Jerusalem, Israel 1994.
- [CC 95] Cham, T.J.; Cipolla, R. "Symmetry detection through local skewed symmetries" *Image and Vision Computing*. Vol. 13-No. 5, pp. 439-450, 1995.
- [CCC00] Conesa, J.; Contero, M. & Company, P. "Comportamiento de los algoritmos de optimización en la reconstrucción geométrica de sólidos" *Anales de Ingeniería Mecánica. XIV Congreso de Ingeniería Mecánica*, vol. 1, pp. 425-430, 2000.
- [CCG98] Conesa, J., Company, P. y Gomis, J.M., "La estructura del aprendizaje en el contexto de los sistemas CAD". *Actas del X Congreso internacional de Ingeniería Gráfica. Volumen III*, (ISBN 84-89791-08-2), 1998, pp. 41-55.
- [CCG99] Conesa Pastor, J.; Company Calleja, P.,and Gomis Martí, J.M. "Initial modeling strategies for geometrical reconstruction optimization-based approaches" *International Conference on Design Tools and Methods in Industrial Engineering. 11th ADM Conference Proceeding*, pp.161-171, 1999.
- [CCP03a] Company, P, Contero M., Piquer A., Aleixos N., Conesa J. y Naya F. "Aplicación docente de un programa de modelado 3D mediante bocetos axonométricos." *XIII ADM – XV INGEGRAF International Conference on Tools and Methods Evolution in Engineering Design*. Napoli, Salerno (Italia), 10 páginas, formato CD (resumen en página 28 del libro de resúmenes), 2003.
- [CCP03b] Conesa, J; Company, P; Piquer, A. y Aleixos, N. "Ampliación del método de inflado axonométrico para la reconstrucción de poliedros." *XIII ADM – XV INGEGRAF International Conference on Tools and Methods Evolution in Engineering Design*. Napoli, Salerno (Italia), 10 páginas, formato CD (resumen en página 65 del libro de resúmenes), 2003.

## Percepción Artificial de Dibujos Lineales

---

- [CGC98] Company, P.; Gomis, J.M. And Contero, M. "An Optimization-Based Algorithm to Reconstruct 3D Models from Single Line Drawings" *II Seminario Italo-Spagnolo*, vol. 2, pp.952-958, 1998.
- [CGC99] Company, P.; Gomis, J.M., and Contero, M. "Geometrical Reconstruction from Single Line Drawings Using Optimization-Based Approaches" *WSCG99 The 7-th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media'99*, 1999.
- [CLD96] Chen, Y.; Langrana, N.A., and Das, A.K. "Perfecting vectorized Mechanical drawings" *Computer Vision and Image Understanding*, Vol. 63, nº 2, pp.273-286, March 1996.
- [Clo71] Clowes, M-B. "On seeing things." *Artificial Intelligence*, 2(1), pp.79-112, 1971.
- [CO 93] Chartrand, G. & Oellermann, O.R. *Applied and Algorithmic Graph Theory*. McGraw-Hill, Inc. ISBN: 0-07-557101-3, 1993.
- [Com90] Company, P. "Una propuesta de lógical como soporte a la Docencia de la Expresión Gráfica" *Actas del II Congreso de Expresión Gráfica en la Ingeniería*, pags. 1-15, (1990).
- [Com97] Company, P. "Integrating Creative Steps in CAD Process", *Actas del Congreso Internacional sobre principios y métodos del diseño en Ingeniería*, vol.1, pp. 295-322, 1997.
- [Con01] Conesa Pastor, J. "Reconstrucción Geométrica de Sólidos Utilizando Técnicas de Optimización" Tesis Doctoral, Universidad Politécnica de Cartagena, 2001.
- [Cor01] Corney, J. *3D modeling with ACIS*. Dun Eaglais Saxe-Coburg, 2001.
- [CP88] Chen, Z.& Perng, D. "Automatic reconstruction of 3-D solid objects from 2-D orthographic views" *Pattern Recognition*, 21, pp. 439-449, 1988.
- [CPC92] Chen, Z.; Perng, D.; Chen, C and Wu C. "Fast reconstruction of 3D mechanical parts from 2D orthographic views with rules" *Computer Integrated Manufacturing*, vol. 5, No.1, pp. 2-9 1992.



- [Cug91] Cugini U. "The problem of user interface in geometric modelling." *Computers in Industry*, vol. 17, pp. 335-339, 1991.
- [D-Cub] <http://www.d-cubed.co.uk/index.htm>
- [Deo76] Deo, N. "Note of Hopcroft and Tarjan's Planarity Algorithm" *Journal of the Association for Computing Machinery*. Vol. 23, No. 1, pp.74-75, 1976.
- [DET94] Di Battista, G.; Eades, P.; Tamassia, R.; Tollis, I.G. "Algorithms for Drawing Graphs: an Annotated Bibliography" *Computational Geometry*, Vol.4, No.5, pp. 235-282, 1994.
- [DL97] Das, A.K., and Langrana, N.A. "Recognition and integration of dimension sets in vectorized engineering drawings" *Computer Vision and Image Understanding*, Vol. 68, n° 12, p. 90-108, October 1997.
- [Dor92] Dori, D. "Dimensioning Analysis. Toward Automatic Understanding of Engineering Drawings" *Communications of the ACM*, vol.35, n°10, pp.92-102, 1992.
- [Dor95a] Dori, D. "Vector-based arc segmentation in the machine drawing understanding system environment" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, n° 11, Noviembre 1995.
- [Dor95b] Dori, D. "Representing pattern recognition-embedded systems through object-process diagrams: the case of the machine drawing understanding system" *Pattern Recognition Letters*, 16, pp. 377-384, 1995.
- [Dor95c] Dori D. "From engineering drawings to 3D CAD models: are we ready now?" *Computer Aided Design*, vol. 27, No. 4, pp 243-254, 1995.
- [DV98] Dori, D and Velkovitch, Y "Segmentation and Recognition of Dimensioning Text from Engineering Drawings" *Computer Vision and Image Understanding* vol. 69, 2, pp.196-201, 1998.

## Percepción Artificial de Dibujos Lineales

---

- [EHB97] Egli, L., Hsu, C.-Y., Bruederlin, B. D. & Elber, G. "Inferring 3D Models from Freehand Sketches and Constraints." *Computer-Aided Design*, Vol. 29, No. 2, pp. 101-112, 1997.
- [EK95] Eberhart, R. C. and Kennedy, J. A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micromachine and Human Science*, Nagoya, Japan. pp. 39-43, 1995.
- [Fer92] Ferguson E.S. *Engineering and the Mind's Eye*. Ed. MIT Press, 1992.
- [Fri86] Friedberg, S.A. "Finding Axes of Skewed Symmetry" *Computer Vision, Graphics and Image Processing*, Vol. 34, pp. 138-155, 1986.
- [FZ91] Forsyth, D. & Zisserman, A. "Reflections on shading" *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 13(7), pp.671-679, Julio 1991.
- [GB 91] Gross, A.D.; Boulton, T.E. "Syman: a symmetry analyzer" *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, Maui, pp. 744-746, 1991.
- [GB 94] Gross, A.D. ; Boulton, T.E. "Analyzing Skewed Symmetries" *International Journal of Computer Vision*, 1994.
- [GCG98] Gomis, J.M.; Company,P and Gil, M.A. "Vectorization in Recoverig Engineering Drawings" *Il Seminario Italo-Spagnolo*, 1998.
- [Gib85] Gibbons, A. *Algorithmic Graph Theory*. Cambridge University Press, 1985.
- [GM95] 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.
- [GM96] 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
- [GN89] Gujar, U.G. and Nagendra, I.V. "Construction of 3D solid objects from orthographic views" *Computer & Graphics*, vol. 13, No. 4, pp. 505-521 1989.

- [Gri97] Grimstead, I.J. "Interactive Sketch Input of Boundary Representation Solid Models" *Tesis Doctoral*, Cardiff University, 1997.
- [GRL97] Grabowski, H.; Rude, S. Liu, C.; Karlsruhe; Klose, J.;Röhmer, S. & Hahn, E. and Dresden "Drawing Reconstruction. Drawing recognition software generates STEP files" *Product Data Journal*, vol. 4, pp. 30-32, 1997.
- [GTS86] Gu, K.; Tang, Z and Sun, J. "Reconstruction of 3D Objects from Orthographic Projections" *Computer Graphics Forum*, vol. 5, pp. 317-324, 1986.
- [GU 83] Ganter, M.A. & Uicker, J.J. "From Wire-Frame to Solid Geometric: Automated Conversion of Data Representations". *Computer in Mechanical Engineering* , Vol. 2 (2), pp.40-45, Sep. 1983.
- [Guz68] Guzmán, A. "Decomposition of a visual scene into three-dimensional bodies" In *AFIPS American Federation of Information Proceedings Fall Joint Computer Conference*, Vol. 33, pp. 291-304, 1968.
- [HF94] Han, C.C., and Fan, K.C. "Skeleton generation of engineering drawings via contour matching" *Pattern Recognition*, vol. 27, n°2, pp. 261-275, 1994.
- [Hig86] Highnam, P.T. "Optimal Algorithms for finding the Symmetries of a Planar Point Set" *Information Processing Letters*, vol. 22, No. 5, Abril, pp. 219-222, 1986.
- [Hof00] Hoffman D.D. "Inteligencia Visual: Cómo creamos lo que vemos" Ediciones Paidós Ibérica. ISBN:84-493-0918-2, 2000.
- [Hoh56] Hohenberg F. *Konstruktive Geometrie für Techniker*. Wien, Springer. 1956. (*Geometría constructiva aplicada a la técnica*). Ed. Labor, Barcelona. ISBN 0-12-355571-X, 1965.
- [Hol98] Holliday-Darr, K. *Applied Descriptive Geometry*. Delmar Publishers ,2nd edition, 1998. (*Geometría descriptiva aplicada*. International Thomson Editores, 2000).
- [HQ82] Haralick, R.M. & Queeney, D. "Understanding Engineering drawings" *Computer Graphics and Image Processing*, 20, 244-258, 1982.

- [HT 74] Hopcroft, J.E.; Tarjan, R.E. "Dividing a graph into triconnected components" 1974.
- [Hu03] Hu X. "Particle Swarm Optimization (PSO) and Computational Intelligence." <http://web.ics.purdue.edu/~hux/tutorials.shtml>. Visited on august 2003.
- [Huf71] Huffman, D.A. "Impossible objects as nonsense sentences. In *Machine Intelligence*, Vol. 6, eds. B. Meltzer and D. Michie. Edinburgh University Press, Edinburgh, pp.259-323, 1971.
- [Ide73] Idesawa, M. "A system to generate a solid figure from three views. *JSME Bulletin*, 16, pp. 216-225, 1973.
- [IMK97] Igarashi, T.; Matsuoka, S.; Kawachiya, S. and Tanaka, H. "Interactive Beautification: A Technique for Rapid Geometric Design." *Proceedings of the ACM Symposium on User Interface Software and Technology UIST'97*, pp. 105-114, 1997.
- [IMT99] Igarashi T., Matsuoka S., Tanaka H. "Teddy: a sketching interface for 3D freeform design." *ACM SIGGRAPH 99 Conference Proceedings*, pp. 409-416, 1999.
- [JM92] Jenkins, D.L. and Martin, R.R. "Applying constraints to enforce users' intentions in free-hand 2-D sketches." *Intelligent Systems Engineering, Vol. 1, Ncoo. 1*, pp. 31-49, 1992.
- [Joh63] Johnson, T.E. "Sketchpad III, Three Dimensional Graphical Communication with a Digital Computer" *Tesis Doctoral, MIT*, 1963.
- [JV97] Janssen, R.D.T., and Vossepoel, A.M. "Adaptative vectorization of line drawing images" *Computer Vision and Image Understanding*, vol. 65, nº 1, pp. 38-56, January 1997.
- [Kan80] Kanade, T. "A theory of origami world" *Artificial Intelligence*, 13(3), pp. 279-311. 1980.
- [Kan81] Kanade, T. "Recovery of the three-dimensional shape of an object from a single view." *Artificial Intelligence*. Vol. 17, pp. 409-460, 1981.

- [Kan86] Kanatani, K.I. "The Constraints on Images of Rectangular Polyhedra" *IEEE transactions on pattern analysis and machine Intelligence* Vol. 8, No. 4, pp. 456-463, 1986.
- [KE95] Kennedy, J. and Eberhart, R. C. "Particle swarm optimization." *Proceedings of IEEE International Conference on Neural Networks*, Piscataway, NJ. pp. 1942-1948, 1995.
- [Kuo98a] Kuo, M. H. "Reconstruction of quadric surface solids from three-view engineering drawings" *Computer Aided Design*, vol. 30, No. 7, pp. 517-527, 1998.
- [Kuo98b] Kuo, M. H. "A Systematic Approach Towards Reconstructing 3D Curved Models from Multiple 2D Views" In Karl Tombre and Atul K. Chhabra, editors, *Graphics Recognition---Algorithms and Systems, Second International Workshop (GREC'97)* vol. 1389 of *Lecture Notes in Computer Science*, pp. 265-279, Springer, Berlin 1998.
- [LB90] Lamb, D. & Bandopadhyay, A. "Interpreting a 3D object from a rough 2D line drawing" In *Proceedings of Visualization'90*, pp.59-66, 1990.
- [LDK95] Lysak, D.B.; Devaux, P.M. & Kasturi, R. "View labeling for automated interpretation of engineering drawings" *Pattern Recognition*, Vol. 28, No.3, pp.393-407, 1995.
- [LF92] 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.
- [Lip98] Lipson, H. "Computer Aided 3D Sketching for conceptual design" Tesis Doctoral. Technion - Israel Institute of Technology – Haifa, 1998.
- [LK94] Lai, C.P and Kasturi, R. "Detection of Dimension Sets in Engineering Drawings" *IEEE transactions on pattern analysis and machine Intelligence*, vol. 16, No. 8, pp. 848-855, 1994.

## Percepción Artificial de Dibujos Lineales

---

- [LMM01] Langbein, F.C.; Mills, B.I.; Marshall, A.D. & Martin, R.R. "Recognizing Geometric Patterns for Beautification of Reconstructed Solid Models" *Proc. of the International Conference on Shape Modelling and Applications*, Genova 2001.
- [LS02] Lipson, H. and Shpitalni, M. "Correlation-Based Reconstruction of a 3D Object From a Single Freehand Sketch" *AAAI Spring Symposium Series- Sketch Understanding*, 2002.
- [LS95] 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.
- [LS96] 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. 51-663, 1996.
- [LS97] Lipson, H. and Shpitalni, M. "Conceptual design and analysis by sketching" *Artificial Intelligence in Design and Manufacturing (AIEDAM)*, 1997.
- [LLR00] Long A.C., Landay J.A., Rowe L.A. y Michiels J. "Visual Similarity of Pen Gestures." *Proceedings of Human Factors in Computer Systems (SIGCHI)*, pp. 360-367, 2000.
- [Mac73] Mackworth, A.K., "Interpreting pictures of polyhedral scenes. *International Journal of Computer Vision*, 1, pp.73-103, 1987.
- [Mal87] Malik, J. "Interpreting line drawings of curved objects" *International Journal of Computer Vision*, 1, pp. 73-103, 1987.
- [Mar89] Marola, G. "On the Detection of the Axes of symmetry of Symmetric and Almost Symmetric Planar Images". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 11, No. 1, pp. 104-108, 1989.
- [Mar91] 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.

- [Mas] [http://www.trl.ibm.com/projects/s7340/modeling/ort\\_e.htm](http://www.trl.ibm.com/projects/s7340/modeling/ort_e.htm)
- [MLM01] Mills, B.J.; Langbein, F.C.; Marshall, A.D.; Martin, R.R. "Approximate Symmetry Detection for Reverse Engineering" *ACM Symposium of Solid Modelling and Applications*, Michigan, 2001.
- [MN97] Masuda, H and Numao, M. "A cell-based approach for generating solid objects from orthographic projections" *Computer Aided Design*, vol. 29, No. 3, pp. 177-187, 1997.
- [MP93] Meeran, S and Pratt, M.J. "Automated feature recognition from 2D drawing" *Computer Aided Design*, vol. 25, No. 1, pp.7-17, 1993.
- [MRC86] Meyer, G.W; Rushmeier, H.E.; Cohen, M.F.; Geenberg, D.P. & Torrance, K.E. "An experimental evaluation of computer graphics imagery" *ACM Transactions on Graphics*, 5(1), pp. 30-50, 1986.
- [MRL93] Martí, E.; Regincós, J, López\_Krane, J and Villanueva, J.J. "Hand line drawing interpretation as three-dimensional objects" *Signal Processing*, vol. 32, pp. 91-110, 1993.
- [NCC03] Naya, F.; Conesa, J.; Contero, M.; Company, P. and Jorge, J. "Smart Sketch System for 3D Reconstruction Based Modeling". *3rd International Symposium on Smart Graphics 2003*, (SG 2003 Heidelberg, Germany, pp. 58-68, 2003.
- [NG88] Nagendra I.V. and Gujar U.G. "3-D Objects from 2-D Orthographic Views - A survey" *Computer & Graphics*, vol. 12, No. 1, pp. 111-114, 1988
- [NIK95] Nayar, S.K.; Ikeuchi, K. & Kanade, T. "Surface reflection: Physical and geometrical perspectives". *IEEE Transactions on Pattern Analysis and Machine Intelligence.*, 13(7), pp. 611-634, Julio 1991.
- [NJC02] Naya, F., Jorge, J., Conesa, J., Contero, M. and Gomis, J.M. "Direct modeling: from sketches to 3D models." *Proceedings of the 1st Ibero-American Symposium in Computer Graphics SIACG*, Guimãraes, Portugal, pp. 109-117, 2002.

## Percepción Artificial de Dibujos Lineales

---

- [NL90] Nagasamy, V., and Langrana, N.A. "Engineering Drawing processing and vectorization system" *Computer Vision, Graphics and Image Processing*, vol. 49, 1990.
- [OAT88] Oh, W.G.; Asada, M. & Tsuji, S. "Model-based matching using skewed symmetry information" *Proceedings of the International Conference on Pattern Recognition*, pp. 1043-1045, 1988.
- [OK01] Oh, B.S. & Kim, C.H. "Progressive 3D reconstruction from a sketch drawing." *Proceedings of the 9<sup>th</sup> Pacific Conference on Computer Graphics and Applications*, pp. 108 -117, 2001.
- [OpenC] <http://www.opencascade.com>
- [OpenG] <http://www.opengl.com>
- [PAG02] Piquer, A., Aleixos, N., Galmes, V. y Company, P. "Estudio comparativo de aplicaciones de delineación por ordenador". *Actas del XIV Congreso internacional de Ingeniería Gráfica.*, 2002 (ISBN 84-699-8559-0), 10 páginas, formato CD (presentación póster y resumen en página 858).
- [Pal99] Palmer, S.E. *Vision Science. Photons to Phenomenology*. The MIT Press, Cambridge, 1999.
- [Par96] Parodi, P. "The complexity of understanding line drawing of Origami scenes" *International Journal of Computer Vision*, 18(2), pp.139-170, 1996.
- [Paras] <http://www.parasolid.com>
- [Pat69] Paton, K. "An Algorithm for finding a Fundamental Set of Cycles of a Graph". *Communications of ACM*, Vol. 12 (9), pp. 514-518, Sep 1969.
- [PC 02] Piquer Vicent, A.; Company Calleja, P. "Simetría Facial Oblicua en Reconstrucción Tridimensional de Dibujos Lineales." XV Congreso Nacional de Ingeniería Mecánica. *Actas del XV Congreso de Ingeniería Mecánica*. Cádiz, 2002.



- [PC 03] Piquer, A.; Company, P. "Simetría Facial Oblicua en Reconstrucción Tridimensional de Dibujos Lineales" *Metal Univers.* (ISSN 1578-8873), nº 12, diciembre, pp. 88, 2002.
- [PCM03] Piquer, A., Company, P. & Martin, R.R. "Skewed mirror symmetry in the 3D reconstruction of polyhedral models". *WSCG'2003 (The 11-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision 2003, Journal of WSCG.* Vol. 11, No. 3, (ISSN 1213-6972), pp. 504-511, 2003.
- [PJB00] Pereira J., Jorge J., Branco V. & Nunes F. "Towards calligraphic interfaces: sketching 3D scenes with gestures and context icons." *WSCG'2000. Conference Proceedings*, Prague 2000.
- [PLC98] Philips I.T.; Liang, J; Cahhabra, A.K., and Haralick R. "A performance Evaluation Protocol for Graphics Recognition Systems" In Karl Tombre and Atul K. Chhabra, editors, *Graphics Recognition---Algorithms and Systems*, Second International Workshop (GREC'97) vol. 1389 of *Lecture Notes in Computer Science*, pp. 372-389, Springer, Berlin 1998.
- [PMC03] Piquer, A.; Martin, R.R.; Company, P. "Using skewed mirror symmetry for optimisation-based 3D line-drawing recognition" *GREC'2003, Fifth IAPR International Workshop on Graphics Recognition*, Barcelona, Spain. Proceedings of the Fifth IAPR International Workshop on Graphics Recognition (ISBN 84-932156-6-X), pp. 182-193, 2003.
- [Pos92] Posch, S. "Detecting Skewed Symmetries" *Proceedings of the International Conference on Pattern Recognition*, pp.602-606, 1992.
- [Pre84] Preiss, K. "Constructing the solid representation from engineering projections" *Computer & Graphics*, vol. 8, No. 4, pp. 381-389 1984
- [QWJ00] Qin S.F., Wright D.K., and Jordano I.N. "From On-line Sketching to 2D and 3D Geometry: A System Based on Fuzzy Knowledge." *Computer-Aided Design*. Vol. 32, No. 14, pp. 851-866, 2000.
- [REG00] <http://www.tec.uji.es/d/regeo>

- [RK88] Robinson J.; Kim, C.E. "Point Symmetry of Convex Digital Regions" *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pp. 604-609, 1988.
- [RM95] Rössli, M and Monagan, G. "A High Quality Vectorization Combining Local Quality Measures and Global Constraints" *Proceedings of the Third International Conference on Document Analysis and Recognition (ICDAR '95)*, 1995.
- [Rob63] Roberts, L.G. "Machine Perception of Three-Dimensional Solids" Tesis de Massachusetts Institute of Technology Certified by Peter Elias (Thesis Supervisor) June, 1963
- [Rob65] Roberts, L.G. "Chapter 9: Machine Perception of three-dimensional solids" *Optical and Electro-Optical Information processing*, 1965.
- [Rob96] Robinson J.J. "Line Symmetry of Convex Digital Regions" *Computer Vision and Image Understanding*, vol. 64, No. 2, Septiembre, pp. 263-285, 1996.
- [Rub92] Rubine D. "Combining gestures and direct manipulation." *Proceedings ACM CHI'92 Conference Human Factors in Computing Systems*, pp. 659-660, 1992.
- [Sak83] Sakurai "Solid Model Input through orthographic views" *Computer and Graphics*, 17(3):243-252,1983
- [Sel88] Sellarès I. & Chiva, J.A. *Fundamentos de los gráficos con ordenador. Principios geométricos y algoritmos básicos*. Edunsa. Ediciones y Distribuciones Universitarias. 1988, ISBN: 84-7747-012-X
- [SG98] Schweikardt, E. & Gross, M.D. "Digital Clay: deriving digital models from freehand sketches." *ACADIA '98*, Quebec City, Canada, pp. 202-211, 1998.
- [Shi95] Shimshoni, I. "Interpreting Images of polyedral objects in the presence of uncertainly" Tesis Doctoral, Urbana, Illinois, 1995.
- [SIT01] Shen, D.; Ip, H.H.S. & Teoh, E.K. "Robust detection of skewed symmetries by combining local and semi-local affine invariants" *Pattern Recognition*, vol. 34, pp. 1417-1428, 2001.

- [SK 01] Saunders, J.A. & Knill, D.C. "Perception of 3D surface orientation from skew symmetry" *Vision Research*, Vol. 41, No. 24, pp. 3163 – 3183, 2001.
- [SKK99] Suh, T.J.; Kim, W.S. & Kim, C. H. "Two Phase 3D Object Reconstruction from Two-View Drawings" *Image Processing, Computer Graphics and Pattern Recognition*, Vol. E82-D, No.7, pp.1093-1100, 1999.
- [SL96] Shpitalni, M & Lipson, H. "Identification of faces in 2D line drawing projection of a wireframe object" *IEEE transactions on pattern analysis and machine Intelligence*, 1996.
- [SLY01] Shum, S.S.P.; Lau, W.S.; Yuen, M.M.F. & Yu, K.M. "Solid reconstruction from orthographic views using 2-stage extrusion" *Computer Aided Design*, Vol. 33, pp. 91-102, 2001.
- [SLY97] Shum, S.S.P.; Lau, W.S.; Yuen, M.M.F. & Yu, K.M. "Solid Reconstruction from Orthographic Opaque Views Using Incremental Extrusion" *Computer & Graphics* Vol. 21, No.6, pp.787-800, 1997
- [SP94] Shimshoni, I. & Ponce, J. "Recovering the shape of Polyhedra Using Line-Drawing Analysis and Complex reflectance Models" *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 514-519, 1994.
- [SS98] Shin, B.S & Shin, Y. G. "Fast 3D solid model reconstruction from orthographic views" *Computer Aided Design*, vol. 30, No. 1, pp. 63-76, 1998.
- [SSM99] Sastry, D.S.; Sasmal, N. & Mukerjee, A. "Efficient Categorization of 3D Edges from 2D Projections" *Graphics Recognition Conference GREC'99*, Jaipur, India,1999.
- [ST 94] Sugimoto, K. & Tomita, F. "Detection of Skewed-symmetrical shape" *Proceedings of the International Conference on Image Processing*. Vol. 1, pp. 696-700, 1994.
- [STV99] Sheynin, S.; Tuzikov, A. & Volgin, D. "Computation of Symmetry Measures for Polygonal Shapes" *Computer Analysis of Images and Patterns: Proceedings of 8th International Conference*, Ljubljana, 1999.- *Lecture Notes in Computer Science*, vol. 1689, Springer, p. 183-190, 1999.

## Percepción Artificial de Dibujos Lineales

---

- [Sug78] Sugihara, K. "Picture language for skeletal polyhedra" *Computr Graphics Image Processing*, vol.8, pp.382-405, 1978.
- [Sug82] Sugihara, K. "Mathematical structures of line drawings of polyhedrons-towards man machine communication by means of line drawings". *IEEE Transactionson Pattern Analysis and Machine Intelligence*, no. 4, pp. 458-469, 1982.
- [Sug86] Sugihara, K. "Machine Interpretation of Line Drawing" *MIT Press* ISBN: 0-262-19254-3, 1986.
- [Sug99] Sugihara, K. "Resolvable representation of polyhedra" *Discrete & Computational Geometry*, Vol. 21, pp. 243-255, 1999.
- [Sut63] Sutherland, I.E. "Sketchpad, A Man-Machine Graphical Communication System" *Tesis Doctoral, MIT*, 1963.
- [TIH98] Tanaka, M, Iwama, K., Hosoda, A., and Watanabe, T. "Descomposition of a 2D assembly drawing into 3D part drawings" *Computer Aided Design*, Vol. 30, No. 1, pp. 37-46, 1998.
- [TJ03] Tate S.J. and Jared G.E.M. "Recognising symmetry in solid models." *Computer-Aided Design*, Vol. 35, pp 673-692, 2003.
- [Ull02] Ullman D.G. "Toward the ideal mechanical engineering design support system". *Research in Engineering Design*. No 13. pp 55-64, 2002.
- [UN93] Ulupinar, F. & Nevatia, R. "Perception of 3-D Surfaces from 2-D Contours" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, No. 1, pp. 3-18, 1993.
- [UN95] Ulupinar, F. & Nevatia, R. "Shape from Contour: Straight Homogeneous Generalized cylinders and Constant Cross Section Generalized Cylinders" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.17-nº2, pp.120-135, 1995.
- [UWC90] Ullman D.G, Wood S. & Craig D. "The Importance of Drawing in the Mechanical Design Process." *Computers & Graphics*, Vol 14, No.2, pp. 263-274, 1990.

- [VM00a] Varley, P.A.C., and Martin R.R. "A System for Constructing Boundary Representation Solid Models from a Two-Dimensional Sketch -- Frontal Geometry and Sketch Categorisation" *1st Korea-UK Joint Workshop on Geometric Modeling and Computer Graphics*, April 2000.
- [VM00b] Varley, P.A.C., and Martin R.R. "A 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*, April 2000.
- [VM00c] Varley, P.A.C., and Martin R.R. "A System for Constructing Boundary Representation Solid Models from a Two-Dimensional Sketch -- Geometric Finishing" *1st Korea-UK Joint Workshop on Geometric Modeling and Computer Graphics*, April 2000.
- [VM00d] Varley, P.A.C., and Martin R.R. "The junction catalogue for labelling line drawings of polyhedra with tetrahedral vertices" *International Journal of Shape Modeling*, 2000.
- [VT95] Vaxivière, P. & Tombre, K. "Subsampling: A structural approach to technical document vectorization" In D. Dori and A. Bruckstein, editors, *Shape, Structure and Pattern Recognition*. World Scientific, pp. 323-332, august 1995.
- [Wal75] Waltz, D.L. "Understanding line drawings of scenes with shadows" In P.H. Winston editor. "The Psychology of Computer Vision" pp. 19-91. McGraw-Hill, New York, 1975.
- [Wan92] Wang, W.D. "On the automatic reconstruction of a 3D object's constructive solid geometry representation from its 2D projection line drawing." D.Sc Thesis, University of Massachusetts, Lowell, MA, 1992.
- [WD98] Wenyin, L and Dori, D. "A Generic Integrated Line Detection Algorithm and Its Object-Process Specification" *Computer Vision and Image Understanding*, vol.70, 3, pp. 420-437, 1998.
- [Wei87] Wei, X. "Computer Vision Method for 3D Quantitative Reconstruction from a Single Line Drawing", *PhD Thesis, Departament of Mathematics, Beijing*

## Percepción Artificial de Dibujos Lineales

---

*University, China* (escrita en Chino, para una revisión en inglés ver el artículo de Wang y Grinstein de 1993).

- [Wey51] Weyl, H. *Simetría*. Princenton University Press. ISBN: 84-7615-564-6, 1951.(*Versión Traducida al Español de 1991*).
- [WG89] Wang, W.D. & Grinstein, G. "A polihedral object's CSG-Rep reconstruction from a single 2-D line drawing" In *Proceedings of 1989 SPIE Intelligence Robots and Computer Vision III: Algorithms and Techniques*, Vol.1192, pp. 230-238, 1989
- [WG93] 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.
- [Wil86] Wilson, R.J. *Introduction to Graph Theory*. Longman Scientific & Technical, 1986.
- [WM80] Wesley, M.A. & Markowsky, G. "Fleshing out wire frames" *IBM Journal of Research and Development*, 24(5), pp. 582-597, 1980.
- [WM81] Wesley, M.A. & Markowsky, G. "Fleshing Out Projections" *IBM Journal of Research and Development*, 25(6), pp. 934-954, 1981.
- [WY 95] Wen, W. & Yuan, B. "Skewed Symmetry detection of closed contours based on their geometric properties" *Proceedings of the Computer Architectures for Machine Perception*, pp. 161-168, 1995.
- [YCT94] Yan, Q.W. ;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.
- [YTL94] Yip, R.; Tam, P. & Leung, D. "Application of elliptic fourier descriptors to symmetry detection under parallel projection" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, No. 3, pp. 277-286, 1994.
- [YY96] You, C.F. & Yang, S.S. "Reconstruction of Curvilinear Manifold Objects from Orthographic Views" *Computer & Graphics*, Vol. 20, No. 2, pp. 275-293, 1996.

- [ZHH96] Zeleznik, R.C.; Herndon, K.P. & Hughes, J.F. "SKETCH: An interface for sketching 3D scenes." *SIGGRAPH'96 Conference Proceedings*, pp. 163-170, 1996.
- [ZW 94] Zabrodsky, H. and Weinshall, D. "Three dimensional symmetry from two dimensional data" *European Conference on Computer Vision*, Stockholm, pp. 403-410, 1994.





# **Anexo I: Aplicación Docente de la Reconstrucción**

---

## **I.1. INTRODUCCIÓN**

---

Un objetivo importante de los cursos básicos de gráficos de ingeniería es la adquisición de “visión espacial”. Esta capacidad se puede describir como la habilidad de imaginar en la mente formas tridimensionales. Adquirir esta habilidad es importante para un futuro ingeniero [Fer92]. El proceso de adquisición de la visión espacial es muy complejo cuando las formas tridimensionales se manipulan a través de dibujos bidimensionales, que, por añadidura, son estáticos. Manipular modelos es mucho mejor, aun cuando los modelos sean “virtuales”, o informáticos, y se visualicen a través de representaciones bidimensionales. En efecto, dichas representaciones tienen dos ventajas fundamentales. La primera es que pueden llegar a un mayor grado de realismo (utilizando sombras, texturas, etc.) con un coste de ejecución reducido. La segunda ventaja es que el modelo puede manejarse “como si lo tuviéramos en la mano”, es decir, que la imagen del objeto puede variar dinámicamente simulando una rotación, un alejamiento o un acercamiento.

Por ello el diseño mediante prototipos virtuales se está introduciendo en los cursos avanzados. Pero, en los cursos básicos se mantiene el diseño mediante dibujos, aunque puede realizarse introduciendo también las nuevas herramientas de delineación asistida [CCG98]. Esto se justifica porque la mayoría de la comunidad académica entiende que la transición debe ser lenta, dado que en ciertos sectores industriales poco punteros, y en

pequeñas y medianas industrias, el diseño mediante dibujos sobrevivirá durante bastante tiempo. La consecuencia es que en estos cursos se intenta conjugar el aprendizaje de los fundamentos de los sistemas de representación con la adquisición y/o consolidación de la visión espacial, pero sin la ayuda de las aplicaciones CAD de modelado. No se recurre a estas aplicaciones porque no tienen interfaces muy amigables, al menos para la fase de diseño conceptual, en la que se manejan ideas incompletas y ambiguas [Com97]. Es decir, que mientras los bocetos permiten a los diseñadores gestionar este proceso de un modo eficiente, las “encorsetadas” ordenes y esquemas de trabajo de los sistemas CAD de modelado actuales tienen la finalidad de permitir la creación de modelos completos y consistentes, e *impedir* la creación de modelos ambiguos. La situación deseable, que ha sido expuesta en trabajos muy notables [UII02], es bien distinta. Un breve estudio de los principales antecedentes, así como nuestra propia interpretación de la situación actual y las tendencias a corto y medio plazo se puede encontrar en [CCP03].

La relación de éste tema con la presente tesis proviene de que la aplicación informática que se ha desarrollado para validar la metodología propuesta en esta tesis tiene también una clara utilidad docente, que sirve para romper con la situación descrita arriba.

En efecto, la versión “de investigación” del programa, es capaz de arreglar automáticamente el boceto, llegando a obtener un modelo final válido; pero aún no es amigable, porque requiere de un cierto grado de interacción con el usuario y como su nombre indica es de investigación. Sin embargo, se ha desarrollado también una versión docente de la aplicación. En ella, el usuario dispone de un “papel virtual” sobre el que dibuja a mano alzada con un “lápiz virtual”. En este caso, se ha limitado el universo de formas reconstruibles a aquellas que no precisan intervención del usuario, de modo que la reconstrucción funciona como una “caja negra”, es decir, que para el usuario, la aplicación dispone de un analizador que reconoce y reconstruye *automáticamente* el modelo tridimensional bocetado por el usuario, y se lo muestra.

La capacidad de visión espacial se potencia, puesto que el usuario puede ver y manipular el modelo tridimensional en todo momento. Por ello, los errores cometidos durante el bocetado de la representación axonométrica (tales como los errores comunes de “olvidar” ciertas aristas) se ponen rápidamente de manifiesto cuando el analizador indica al usuario que no puede generar ningún modelo 3D geoméricamente válido. O bien, cuando el analizador genera un modelo, perfectamente coherente con el boceto, pero discrepante con la imagen mental que el usuario tiene del objeto que ha pretendido bocetar.

## I.2. BOCETADOR/MODELADOR DOCENTE

En esta primera versión, el boceto que introduce el usuario es una representación pictórica de una forma poliédrica. Es decir, una representación pseudo-axonométrica del modelo. Este tipo de representación favorece el aprendizaje de la capacidad de visión espacial del alumno. En efecto, es comúnmente aceptado que las representaciones ortográficas (tales como las vistas diédricas) son apropiadas para “medir”, mientras que las axonométricas y perspectivas son más apropiadas para “ver”.

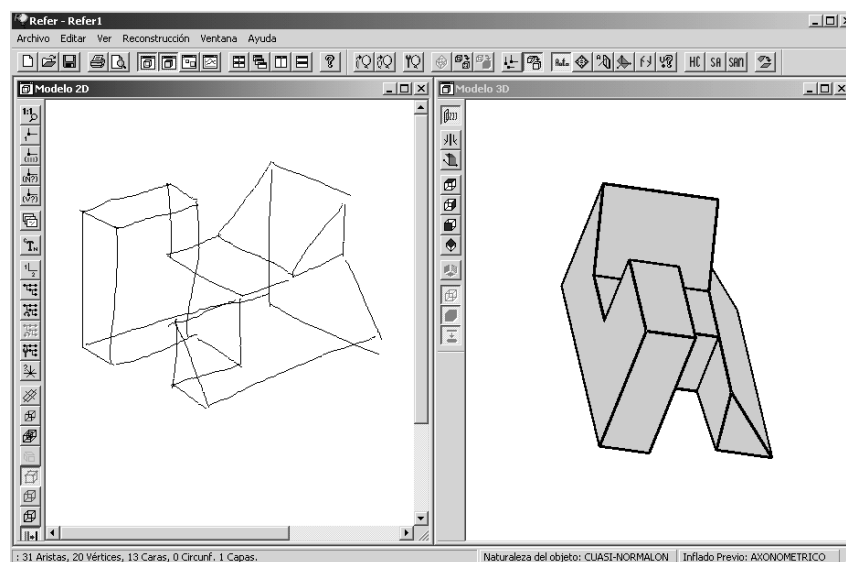


Figura I-1. Sesión de bocetado en REFER

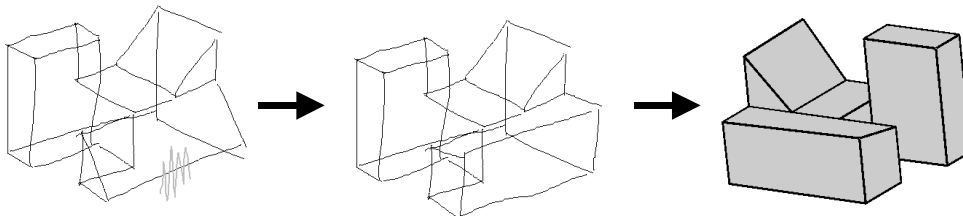
Por otra parte, hay que destacar que utilizar un bocetador axonométrico en lugar de utilizar los instrumentos virtuales de soporte a la delineación de axonométricas presentes en las aplicaciones comerciales de delineación por ordenador potencia la creatividad; dado que dichas herramientas ponen el énfasis en conseguir delinear representaciones axonométricas geoméricamente rigurosas (es decir, representaciones cuyas medidas longitudinales y angulares cumplen las leyes de la geometría proyectiva correspondientes a las proyecciones axonométricas). Al contrario que los programas CAD comerciales, el bocetador pone el énfasis en controlar la topología del objeto diseñado, y se desentiende de sus dimensiones finales.

## Percepción Artificial de Dibujos Lineales

---

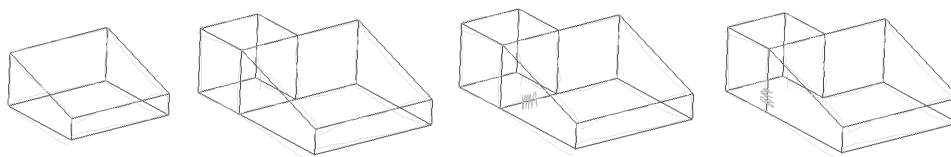
Con todo, lo que entendemos que es el elemento fundamental de la aplicación propuesta, es que dispone de una segunda ventana en la que se visualiza el modelo tridimensional del objeto bocetado (Figura I-1).

Es decir, que el usuario/diseñador puede ver el modelo tridimensional mientras él está bocetando una representación axonométrica del mismo. El modelo aparece representado con caras opacas, o por medio de una representación alámbrica (a voluntad del usuario), y puede ser rotado y escalado dinámicamente por el usuario. La modificación interactiva del boceto, repercute automáticamente en el modelo (Figura I-2)



*Figura I-2. Modificación interactiva del boceto, y actualización automática del modelo*

La facilidad de borrar también ayuda a bocetar apoyándose en líneas auxiliares que pueden ser fácilmente eliminadas (Figura I-3).



*Figura I-3. Proceso de bocetado*

El usuario también puede elegir vistas preprogramadas de dicho modelo. A saber, las vistas ortográficas principales (alzado, planta y perfil) y las dos representaciones axonométricas estandarizadas (isométrica y dimétrica). Las aristas ocultas correspondientes a las vistas seleccionadas, también pueden ser obtenidas a voluntad del usuario.

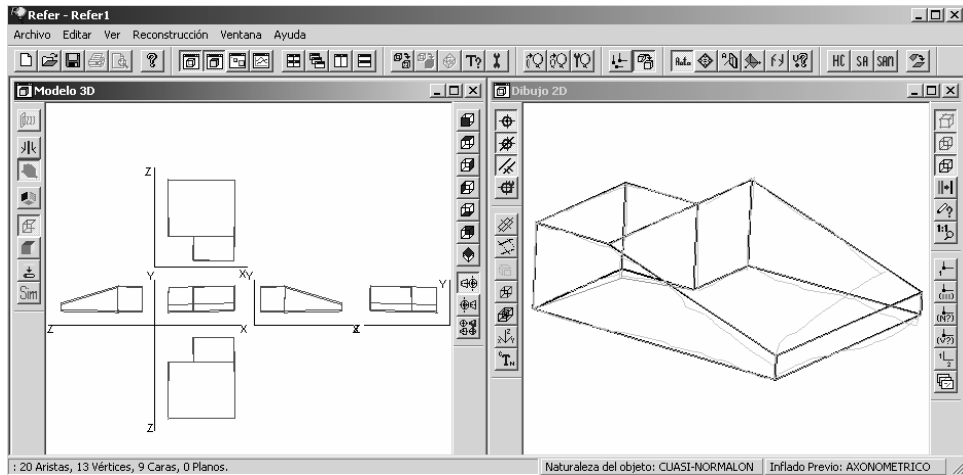


Figura I-4. Vistas ortográficas de un modelo.

Esta facilidad se ha introducido para simplificar el aprendizaje del sistema multivista. Se pretende que el alumno vea cuáles son las vistas que el sistema ha generado automáticamente a partir del modelo tridimensional que él ha bocetado. La presencia de los ejes del sistema de referencia ayuda a incrementar la capacidad de visión espacial y fomenta la interrelación entre las representaciones de tipo axonométrico y ortográfico.

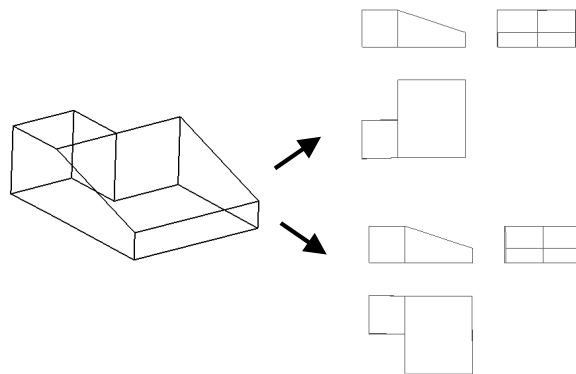


Figura I-5. Inversas de Eckhart.

La ambigüedad de las representaciones alámbricas que son compatibles con dos modelos sólidos "inversos" también se ha tenido en cuenta. El usuario puede conmutar a voluntad para mostrar lo que hemos denominado "inversas de Eckhart", tal como se muestra en la Figura I-5.

Otro tipo de características propias de la aplicación pueden utilizarse para plantearle al alumno preguntas que pueden ser contestadas con ayuda del sistema. ¿Cuántos planos de simetría tiene la pieza bocetada en la mitad izquierda de la Figura I-6? La pregunta es fácil de contestar con ayuda del detector de planos de simetría incorporado en la aplicación, que permite la visualización instantánea de la solución (tal como muestra la mitad derecha de la Figura I-6). Este tipo de cuestiones tiene un gran valor formativo para alumnos que, precisamente por sus carencias en cuanto a visión espacial, aún no tienen la soltura necesaria para percibir con claridad este tipo de propiedades. Por ejemplo, no es frecuente que los alumnos adivinen que hay cuatro planos de simetría asociados a la cuña representada en la parte izquierda de la Figura I-6. Lo normal es que *inconscientemente* descarten dos de esos planos ( $\sigma_3$  y  $\sigma_4$ ) por no cumplir percepciones métricas ( semejanza de proporcionalidad longitudinal y angular de las dos mitades en que dividen al objeto). Es decir, los alumnos no son capaces de discriminar las características topológicas y métricas de los objetos. Incluso es frecuente que no identifiquen  $\sigma_2$  como plano de simetría, por ser un plano que no aporta información nueva y relevante para ellos sobre la forma de la pieza. En otras palabras, no son capaces de analizar el objeto para extraer toda la información geométrica del mismo. Sólo son capaces de extraer aquella información que es relevante para saber cómo es el objeto.

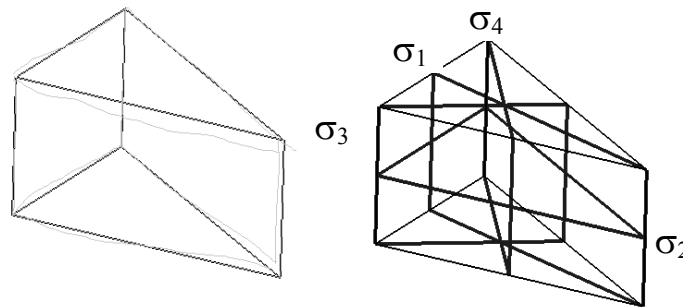


Figura I-6. Planos de simetría detectados por REFER

## I.3. DISCUSIÓN

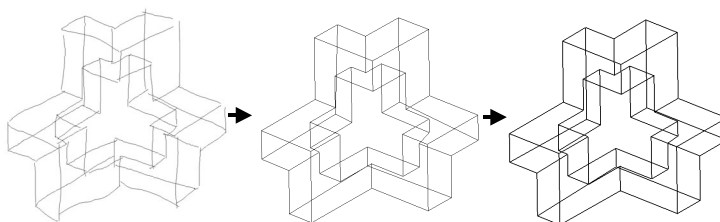
---

Al utilizar esta aplicación, la sensación de que bocetar es no es un trabajo provechoso desaparece, puesto que el alumno comprueba que tras bocetar el modelo que ha

concebido, obtiene automáticamente tanto un modelo tridimensional como distintas representaciones bidimensionales normalizadas del mismo. En otras palabras, tras concebir, mediante un boceto, la forma concebida se genera automáticamente y no hay que construirla por segunda vez.

Obviamente, la capacidad de visión espacial se potencia, tal como ha quedado sobradamente justificado en el apartado anterior.

La principal limitación es que el bocetador sólo es muy simple e intuitivo cuando se representan objetos poliédricos de complejidad media o baja, pero requiere mucha habilidad en delineación y una gran capacidad de visión espacial cuando se pretende generar formas complejas. Ejemplos como el de la Figura I-7 muestran que sólo un experto puede llegar a representaciones tan sofisticadas, o, lo que es lo mismo, que la herramienta no tiene utilidad práctica para un diseñador experto que concibe formas sofisticadas.



*Figura I-7. Boceto complejo y dibujo lineal “arreglado” mediante reconstrucción 2D.*

Es cierto que ante estos casos complejos, la versión “de investigación” del programa es capaz de arreglar automáticamente el boceto, llegando a obtener un modelo final válido (Figura I-8); pero aun no es amigable, porque requiere tiempo de espera y, principalmente, interacción con el usuario.

Pese a que el sistema aún no es fácil de usar cuando se trata de bocetar formas complejas como la descrita arriba, el planteamiento de partida de la versión docente se ha cumplido con creces en la fase de auto-experimentación a que ha sido sometido. Esperamos que su experimentación en el aula avale estas buenas expectativas y aporte información relevante para mejorarlo.

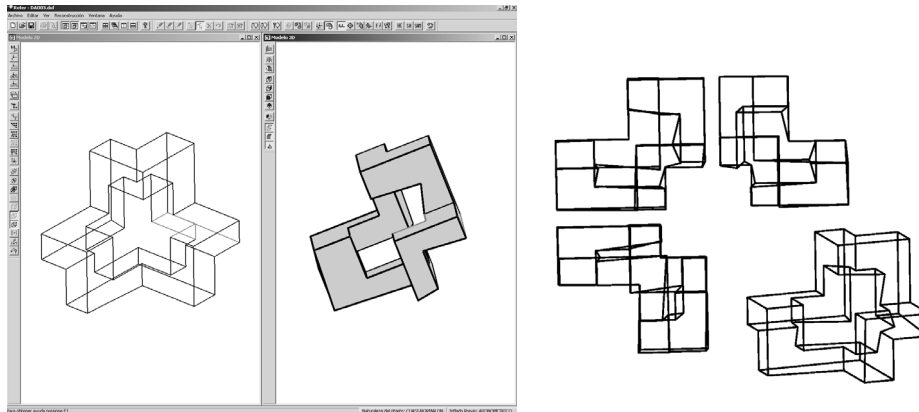


Figura I-8. Boceto “arreglado” y modelo obtenido.

## I.4. CONCLUSIONES Y TRABAJOS FUTUROS

---

El bocetador es una herramienta sencilla y fácil de incorporar a un currículo de diseño mediante dibujos. Su principal utilidad es potenciar la capacidad de visión espacial al inicio del curso. Es especialmente atractiva porque en ese momento el alumno aún no tiene la suficiente habilidad en el manejo de los programas de CAD 2D comerciales, y porque las herramientas para delinear axonometrías que incluyen dichos programas son muy limitadas y obligan a un estilo de trabajo poco natural y nada amigable.

En segundo lugar, el bocetador hace más atractivo para el alumno el aprendizaje necesario para bocetar, puesto que obtiene un resultado tangible directamente del boceto, y es una ayuda muy importante en el proceso de aprender a reconocer formas 3D.

El programa será más potente y amigable si se introducen algunas operaciones habituales, además de la de dibujar aristas y borrar arista. Operaciones tales como la extrusión, la revolución y la simetría ayudarán a generar dibujos complejos (como el de la Figura I-7 y la Figura I-8) con menos esfuerzo y más precisión, ampliando así el conjunto de formas directamente construibles a partir de boceto. También es de esperar que la introducción en el aula de “lápices virtuales” más naturales (tales como los punteros de los tablet PC) incremente aun más la amigabilidad de este entorno de trabajo.



## **Anexo I: Aplicación Docente de la Reconstrucción**

---

Por último, la introducción de los tablet PC's con conexiones infrarrojas a los cañones de video, supondrá que herramientas como las descritas en este trabajo se convertirán en elementos fundamentales del "teachware" vinculado a la enseñanza de los gráficos de ingeniería.

