

Obtención de Modelos 3D a partir de Proyecciones Axonométricas

⁽¹⁾R. Béjar, ⁽¹⁾A. Cabiscol, ⁽¹⁾C. Fernández, ⁽²⁾M. López y ⁽¹⁾F. Manyà

⁽¹⁾ *Departamento de Informática e Ingeniería Industrial*

Universitat de Lleida, Jaume II, 69, E-25001 Lleida

e-mail: {ramon,alba,cesar,felip}@eup.udl.es

⁽²⁾ *Departamento de Expresión Gráfica en la Ingeniería*

Universitat Politècnica de Catalunya, Av. Víctor Balaguer, s/n, 08800 Vilanova i la Geltrú

e-mail:membrilla@ege.upc.es

Resumen

En este artículo presentamos un método computacional para la resolución del problema de la interpretación de una escena 3D a partir de una proyección axonométrica de la misma. Si bien este problema había sido tratado hasta ahora mediante técnicas basadas en SAT o CSPs, aquí proponemos un método basado en un tipo específico de fórmulas proposicionales multivaluadas. Nuestro método puede ser competitivo con SAT porque las fórmulas obtenidas son más compactas que las obtenidas con lógica clásica.

Palabras clave: interpretación de imágenes, proyecciones axonométricas, lógica multivaluada.

1. Introducción

En este artículo presentamos un nuevo método, basado en lógica proposicional, para interpretar y comprender dibujos de objetos poliédricos formados por vértices de tres caras. El objetivo es obtener modelos 3D a partir de proyecciones axonométricas.

La interpretación espacial de líneas de dibujo es un tema estudiado por las comunidades de Visión Artificial y Geometría Estructural. Éstas desarrollan algoritmos para automatizar la obtención de modelos 3D a partir de dibujos 2D tales que, como sucede con la visión humana, sean capaces de rechazar imágenes imposibles.

En este trabajo partiremos del sistema de etiquetado de Huffman [1] y Clowes [2], que asigna a cada línea o arista de una representación axonométrica R una etiqueta que denota si la arista es convexa, cóncava o de frontera u oclusión. El sistema asigna una

etiqueta de tipo + a aquellas aristas que son convexas y en las que las caras que concurren en la arista son visibles en la representación R; asigna una etiqueta de tipo - a aquellas aristas que son cóncavas y en las que las caras que concurren en la arista son visibles en la representación R; y asigna una etiqueta de tipo (\rightarrow , \leftarrow) a aquellas aristas en las que únicamente es visible una cara de las que concurren en la arista. Estas últimas serán las aristas frontera y las caras que concurren en la arista presentarán una oclusión.

Por otra parte, en función del número de aristas que concurren en un vértice y su posición en la representación, el sistema de Huffman y Clowes clasifica las uniones en los vértices. Las uniones pueden ser de cuatro tipos: L, Y, T y F. Además, como las aristas que concurren en un vértice tienen etiqueta, estos cuatro tipos de uniones dan lugar a un catálogo que está formado por 18 uniones que representan todas las uniones que son físicamente realizables en una escena poliédrica 3D. Si una proyección axonométrica admite un etiquetado de las aristas consistente con las uniones del catálogo de Huffman y Clowes, entonces podemos obtener un modelo 3D. En cambio, si el etiquetado es inconsistente, la proyección corresponde a un objeto poliédrico 3D físicamente irrealizable. Sin embargo, cabe mencionar que el conjunto de proyecciones axonométricas posibles con el catálogo de Huffman y Clowes no es completo, en el sentido que se han estudiado extensiones del mismo [3] que hacen válidas ciertas proyecciones que con el catálogo de Huffman y Clowes no son válidas.

Dada una representación axonométrica R de un objeto 3D, nuestro método empieza por representar R y las restricciones de las uniones por una fórmula F de la lógica proposicional multivaluada, de manera que R corresponde a un objeto 3D si, y sólo si, F es satisfactible (es decir, no es contradictorio). A continuación, determina la satisfactibilidad de F utilizando MV-Satz [4], que es un algoritmo de satisfactibilidad que incorpora técnicas de inteligencia artificial. Finalmente, si F es satisfactible, construye un etiquetado correcto del objeto 3D a partir del modelo lógico de F que ha derivado MV-Satz.

2. Fórmulas multivaluadas

En este trabajo proponemos la resolución del problema mediante su reducción al problema de la satisfactibilidad de las MV-fórmulas. Las MV-fórmulas son unas fórmulas que generalizan las formas normales conjuntivas (FNCs) de la lógica clásica proposicional. Se diferencian en que consideran un conjunto de valores de verdad que puede contener más de dos valores y en que podemos usar literales multivaluados que se satisfacen por más de una interpretación diferente para su variable. Además, asumen la existencia de un orden total entre los elementos del conjunto de valores de verdad. Este orden ayuda a expresar de manera más compacta algunos literales multivaluados.

Definición 1. Un conjunto de valores de verdad T es un conjunto finito $\{i_1, i_2, \dots, i_n\}$ de valores entre los que existe un orden total. Un literal multivaluado es una expresión de la forma $S:p$, donde S es un subconjunto de T y p es una variable proposicional. Si S es de la forma $\{i\}$ o de la forma $\uparrow i = \{j \mid j \geq i\}$ o bien de la forma $\downarrow i = \{j \mid j \leq i\}$ entonces diremos que $S:p$ es un MV-literal. Una MV-cláusula es una disyunción de MV-literales y una MV-fórmula es una conjunción de MV-cláusulas.

Definición 2. Una interpretación es una función del conjunto de variables proposicionales al conjunto de valores de verdad. Una interpretación satisface un MV-literal $S:p$ si el valor que asigna a p se encuentra en S , satisface una MV-cláusula si satisface al menos uno de sus MV-literales y satisface una MV-fórmula si satisface todas sus MV-cláusulas.

Ejemplo: Para expresar que $(\{2\}:p_1 \Rightarrow (\{3\}:p_2 \vee \{6\}:p_3))$, empleamos esta MV-cláusula:

$$(\downarrow 1 : p_1 \vee \uparrow 3 : p_1 \vee \{3\}:p_2 \vee \{6\}:p_3)$$

3. Definición formal del problema

El problema de interpretación de figuras tridimensionales que aquí consideramos se basa en el problema original tratado por Huffman y Clowes, aunque es posible

considerar pequeñas variaciones del mismo que pueden ser también resueltas mediante nuestro método. El problema se basa en los siguientes principios:

1. El dibujo bidimensional es una proyección axonométrica de una unión de objetos tridimensionales. Los objetos tridimensionales son objetos poliédricos, donde cada vértice es la intersección de exactamente tres planos. La figura 1 muestra proyecciones axonométricas válidas de dos objetos poliédricos.
2. Cada línea del dibujo representa una arista de uno de los poliedros. Se ignoran sombras, líneas ocultas y zonas de cambio de color.

Una interpretación particular del dibujo se obtiene al etiquetar todas las aristas del mismo de manera consistente. Este etiquetado nos informa sobre si las aristas son concavas o convexas o bien si son aristas que unen dos caras de las cuales sólo una es visible en la proyección axonométrica.

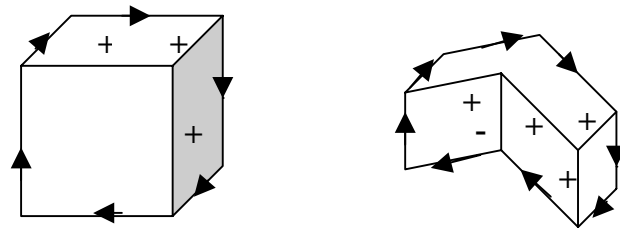


Figura 1. Proyecciones axonométricas de objetos tridimensionales

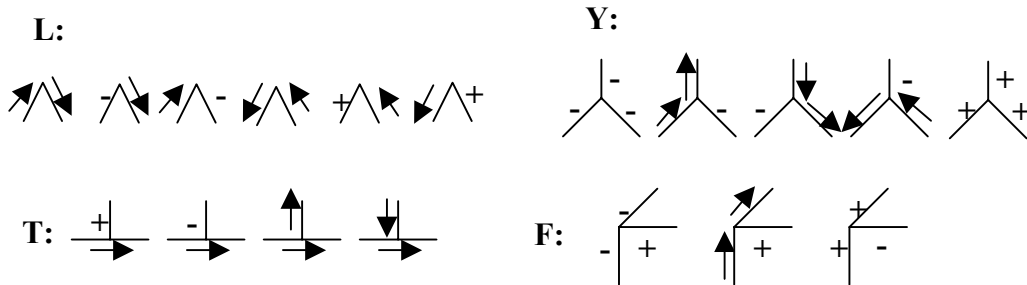


Figura 2. Catálogo Huffman-Clowes de etiquetados válidos para aristas de vértices triédricos.

Después de considerar las diferentes formas en las que podemos formar vértices triédricos y los diferentes puntos de vista desde donde pueden ser observados, Huffman y Clowes obtuvieron el catálogo de 18 vértices posibles que pueden aparecer en una

proyección axonométrica (ver la figura 2). Dentro de un vértice, numeramos sus aristas empezando por la que se encuentre en su posición superior izquierda, y avanzando en el sentido de las agujas del reloj.

4. Codificación mediante una MV-fórmula

Consideramos que como entrada se nos da el conjunto total de vértices (V) y aristas (A) de la proyección axonométrica del objeto. Cada arista se describe mediante sus dos vértices (v_1 y v_2) y mediante dos números (v_{1i} y v_{2i}) que indican, dentro de los dos vértices, que dos aristas están realmente fusionadas en una sola. Para cada vértice tenemos una etiqueta que indica su tipo, de entre los cuatro tipos diferentes (L,Y,T,F). A partir del grafo (V,A), construimos una MV-fórmula de la siguiente manera:

1. El conjunto de valores de verdad, T , es un conjunto con los seis primeros números naturales $\{ 1,2,3,4,5,6 \}$. Para cada uno de los cuatro tipos de vértices, cada valor de verdad representa una de las diferentes formas en las que ese vértice podría aparecer etiquetado. Obsérvese en la figura 2, que como máximo necesitamos seis identificadores de etiquetados válidos para cualquier tipo de vértice.
2. El conjunto de variables proposicionales, Var , es el conjunto $\{ v \mid v \in V \}$. Cada variable representa un vértice distinto de la proyección axonométrica.
3. El conjunto de MV-cláusulas de la fórmula es la unión de diferentes conjuntos de MV-cláusulas. Una interpretación que satisfaga la fórmula nos dará un etiquetado válido para los vértices de la proyección. Las MV-cláusulas aseguran que el etiquetado asignado a un vértice es consistente con el de sus vértices vecinos, según el catálogo de la figura 2. Como la especificación completa de las cláusulas es muy larga, aquí nos limitamos a describir un subconjunto particular de las mismas. Consideremos el caso donde queremos asegurar que el etiquetado para un vértice v_1 , que es de tipo F, sea consistente con el del vértice v_2 , que es de tipo Y, porque se encuentran conectados mediante una arista común en las posiciones de las aristas v_{1i} y v_{2i} de los vértices v_1 y v_2 , respectivamente. Tendremos un conjunto diferente de tres MV-cláusulas, en función de las posiciones de la arista en los dos vértices (v_{1i} y v_{2i}). Estos son los casos posibles:

$$\begin{aligned}
V_{ii} = 1 \text{ y } v_{2i} = 1: & (\{1\}: V_1 \Rightarrow (\{1\}: V_2 \vee \{4\}: V_2)) \wedge (\{2\}:v1 \Rightarrow \{3\}:v2) \wedge (\{3\}:v1 \Rightarrow \{5\}:v2) \\
V_{ii} = 1 \text{ y } v_{2i} = 2: & (\{1\}: V_1 \Rightarrow (\{1\}: V_2 \vee \{2\}: V_2)) \wedge (\{2\}:v1 \Rightarrow \{4\}:v2) \wedge (\{3\}:v1 \Rightarrow \{5\}:v2) \\
V_{ii} = 1 \text{ y } v_{2i} = 3: & (\{1\}: V_1 \Rightarrow (\{1\}: V_2 \vee \{3\}: V_2)) \wedge (\{2\}:v1 \Rightarrow \{2\}:v2) \wedge (\{3\}:v1 \Rightarrow \{5\}:v2) \\
V_{ii} = 2 \text{ y } v_{2i} = 1: & (\{1\}: V_1 \Rightarrow \{5\}:V_2) \wedge (\{2\}:v1 \Rightarrow \{5\}:V_2) \wedge (\{3\}: V_1 \Rightarrow (\{1\}: V_2 \vee \{4\}: V_2)) \\
V_{ii} = 2 \text{ y } v_{2i} = 2: & (\{1\}: V_1 \Rightarrow \{5\}:V_2) \wedge (\{2\}:v1 \Rightarrow \{5\}:V_2) \wedge (\{3\}:V_1 \Rightarrow (\{1\}:v2 \vee \{2\}: V_2)) \\
V_{ii} = 2 \text{ y } v_{2i} = 3: & (\{1\}: V_1 \Rightarrow \{5\}: V_2) \wedge (\{2\}:v1 \Rightarrow \{5\}:v2) \wedge (\{3\}:v1 \Rightarrow (\{1\}:v2 \vee \{3\}:v2)) \\
V_{ii} = 3 \text{ y } v_{2i} = 1: & (\{1\}: V_1 \Rightarrow (\{1\}: V_2 \vee \{4\}: V_2)) \wedge (\{2\}:v1 \Rightarrow \{2\}:v2) \wedge (\{3\}:v1 \Rightarrow \{5\}:v2) \\
V_{ii} = 3 \text{ y } v_{2i} = 2: & (\{1\}: V_1 \Rightarrow (\{1\}: V_2 \vee \{2\}: V_2)) \wedge (\{2\}:v1 \Rightarrow \{3\}:v2) \wedge (\{3\}:v1 \Rightarrow \{5\}:v2) \\
V_{ii} = 3 \text{ y } v_{2i} = 3: & (\{1\}: V_1 \Rightarrow (\{1\}: V_2 \vee \{3\}: V_2)) \wedge (\{2\}:v1 \Rightarrow \{4\}:v2) \wedge (\{3\}:v1 \Rightarrow \{5\}:v2)
\end{aligned}$$

Además de dos cláusulas adicionales que aseguran que los vértices v_1 y v_2 , tipo F e Y respectivamente, sólo tomen valores de verdad que representen etiquetados válidos:

$$(\downarrow 3: V_1) \wedge (\downarrow 5: V_2)$$

5. Complejidad

El procedimiento ideado por Waltz [5], basado en CSPs, es capaz de etiquetar escenas “típicas” con un tiempo promedio lineal en el número total de aristas de la escena. Sin embargo, sabemos que en el peor caso, para algunas escenas, la resolución de este problema no puede llevarse a cabo con tiempo polinomial [6]. Nuestro objetivo en el futuro es estudiar el rendimiento de nuestro método para escenas típicas.

6. Referencias

1. D.A. Huffman, “Impossible objects as nonsense sentences”, B. Meltzer y D. Michie Ed., Machine Intelligence 6, Edinburgh University Press, 295-323, Edinburg, 1971.
2. M.B. Clowes, “On seeing things”, Artificial Intelligence 2: 79-116, 1971.
3. C. Ansótegui et al., “Resolution methods for many-valued CNF formulas”, SAT Conference, 2002
4. T. Kanade, “A theory of Origami world”, Artificial Intelligence 64: 147-160, 1980.
5. D.Waltz, “Understanding line-drawings of scenes with shadows”, in: The Psychology of Computer Vision, McGraw-Hill, 19-91, 1975.
6. L.M. Kirousis, C.H. Papadimitriou, “The complexity of recognizing polyhedral scenes”, Journal of Computer and System Sciences 37: 14-38, 1988.