

SISTEMA DE RECUPERACIÓN DE INFORMACION 3D EN ESCENAS DINAMICAS

Antonio Adán

E. S. Informática, UCLM, Paseo de la Universidad 4, 13071 Ciudad Real, Antonio.Adan@uclm.es

Fernando Molina

E. S. Informática, UCLM, Fernando.Molina@uclm.es

Luis Morena

E. S. Informática, UCLM, Lmorena@proyectos.inf-cr.uclm.es

Javier García

E. S. Informática, UCLM. jgarciabe@soluziona.com

Arturo Jiménez.

E. S. Informática, UCLM. ajimenezd@soluziona.com

Resumen

En este artículo se presenta una nueva técnica de luz estructurada color basada en una estrategia de codificación desordenada. El objetivo de este método es la recuperación de información 3D en escenas dinámicas de tal forma que el problema de correspondencia sea fácil y robustamente resuelto.

El patrón consiste en una malla de círculos de color con topología de conectividad seis, donde un círculo es codificado a través de su color y el de sus vecinos. En esta codificación se permite repetición de color y desorden lo que implica que se incremente la distancia de Hamming entre códigos contiguos. Como consecuencia, las pérdidas de códigos debidas a sombras u oclusiones en la escena se pueden tratar eficientemente. Por otro lado, el coste computacional en la fase de recuperación de códigos se reduce significativamente.

El sistema presentado ha sido implantado y probado en laboratorio en escenas dinámicas bajo especificaciones de resolución media y movimiento lento dando resultados satisfactorios.

Palabras Clave: Luz estructurada codificada, patrón desordenado.

1 INTRODUCCIÓN

1.1 MÉTODOS DE CODIFICACIÓN EN SISTEMAS DINÁMICOS DE LUZ ESTRUCTURADA.

Las técnicas de luz estructurada son una conocida alternativa a las técnicas clásicas de estereovisión en

aplicaciones de composición (integración) de superficies. En esta clase de técnicas, un proyector de patrones de luz sustituye a una cámara en la escena. Obviamente, el difícil problema de correspondencia estéreo se convierte en un problema de correspondencia entre píxeles de la imagen y puntos del patrón proyectado. Dependiendo del tipo de proyección y de la estructura de los patrones, existen diferentes aplicaciones en este campo. Recientemente Salvi et al. han presentado una revisión en [13].

Las técnicas de multiplexado temporal, donde un conjunto de patrones se proyectan sucesivamente, se utilizan para escenas estáticas. Por el contrario, los métodos en los que se proyecta un solo patrón son adecuados para escenas dinámicas. En este artículo, se presenta una nueva estrategia para la recuperación de superficies tridimensionales en ambientes dinámicos.

Dentro de los métodos dinámicos, la diferencia esencial consiste en cómo los nodos (o puntos) del patrón se codifican y en la forma de establecer la correspondencia cámara-proyector. Los párrafos siguientes se dedican a describir los avances más importantes logrados en este campo.

Existen algunas técnicas que codifican los puntos a través de características locales, por ejemplo, nivel de gris [7] o color [15]. En este caso, un amplio grupo de características se deben incluir en el patrón para identificar un punto de la imagen. Además, aparecen varios problemas de ambigüedad para píxeles contiguos. Consecuentemente, tales métodos son muy sensibles al ruido y poco prácticos.

Se puede decir que las técnicas de codificación basadas en vecindad son las más populares y

efectivas. En estas, un punto del patrón se codifica teniendo en cuenta la posición y características (símbolos) de un conjunto de puntos generalmente llamados sus ‘vecinos’.

Así, hay patrones que codifican líneas verticales, horizontales o ambas. Maruyama et al. [9] diseñan un patrón formado por cortes verticales que se determinan en la imagen mediante su propia longitud y las de sus seis segmentos adyacentes. Dado que la longitud de los segmentos proyectados puede variar dependiendo de la normal de la superficie, el eje óptico y la profundidad, la fiabilidad del sistema es limitada. Boyer y Kak [2], considerados los pioneros en la proyección de luz estructurada para escenas dinámicas, utilizan un solo patrón con franjas verticales codificadas con tres colores y separadas con franjas negras mientras que Chen et al. [3] mejoran su método utilizando dos cámaras. Esto les permite establecer una correspondencia más sencilla entre cada par de franjas y calcular la posición 3D tras el proceso de triangulación.

Los métodos basados en las secuencias de De Bruijn constituyen un grupo de técnicas que definen patrones bien construidos en el sentido matemático [8]. La mayoría de ellos utiliza una secuencia unidimensional de n símbolos en los cuales todas las ventanas posibles de dimensión- l ($1 \leq l \leq n$) son únicas en el patrón. Esta idea ha sido directamente aplicada para codificar columnas o filas en patrones. Monks [10] utiliza un patrón compuesto de franjas horizontales con seis colores, tomando ventanas de tres componentes, e identifica cada franja con sus dos vecinas. Salvi et al. [12] proponen un patrón compuesto de una malla de líneas horizontales y verticales codificadas con seis colores. Cada punto de cruce se identifica teniendo en cuenta sus correspondientes colores y los de sus dos líneas contiguas horizontales y verticales. Zhang et al. [16] desarrollan una técnica que consiste en proyectar patrones de franjas de cinco colores que se alternan utilizando una ventana de tres componentes. Así, las transiciones de color proyectado se hacen corresponder con los bordes observados en la imagen. Recientemente Hall-Holt et al. [6] proyectan en la escena cuatro patrones de barras verticales dando cada patrón una parte del código. Este método es capaz de manejar escenas de movimiento lento a pesar de proyectar más de un patrón. Vuylsteke y Oosterlinck [14] utilizan un solo patrón codificado binario que se implementa por medio de cuadrados blancos y negros en ventanas bidimensionales (2×3) separadas por dos cuadrados blancos. De hecho, codifican la columna de cada punto al recuperar cada ventana 2×3 y obtener su código. Esta estrategia puede mejorar las restricciones de las técnicas basadas en ventanas unidimensionales. Ciertamente

es más compacta y menos sensible al ruido o a las discontinuidades de profundidad.

Finalmente, las técnicas que identifican puntos en lugar de líneas son también muy eficientes. En ellas es usual diseñar un mapa de símbolos en el espacio bidimensional con conectividad $4/8$ y donde cada ventana bidimensional aparece solo una vez. Si todas las ventanas posibles se incluyen en el patrón su estructura se denomina *mapa perfecto*, en caso contrario se denomina *submapa perfecto*. Estas técnicas también difieren en la forma en que la codificación se inserta en el patrón. La investigación llevada a cabo por Griffin et al. [5] sobre el máximo tamaño de una matriz bidimensional codificada ha hecho posible el desarrollo de varias técnicas. En el patrón de Griffin se identifica un punto por él mismo y sus cuatro vecinos. El código de color se le asigna utilizando los tres colores primarios. Posteriormente, Davies et al. [4] iluminan la escena con un patrón hexagonal de puntos de color. Sin embargo esta topología no se utiliza para establecer la correspondencia. Por medio de una adaptación tridimensional de la transformada de Hough detectan en la imagen los puntos cercanos a cada línea epipolar y finalmente resuelven el problema de correspondencia. Morano et al. [11] diseñan un *submapa perfecto* utilizando un algoritmo incremental donde se impone la distancia de Hamming entre ventanas (3×3).

1.2 CONTRIBUCIONES DE NUESTRO MÉTODO

Tras presentar brevemente las técnicas de luz estructurada en escenas dinámicas más importantes, presentamos un nuevo método que se puede incluir en la categoría de *submapas perfectos*. Una versión inicial se puede encontrar en [1]. En esta sección, analizamos varios problemas que surgen en estas técnicas y discutimos cómo nuestro nuevo método, basado en patrones desordenados, puede mejorarlas.

Existen varios aspectos de las técnicas expuestas anteriormente que conviene resaltar. En primer lugar, las sombras y oclusiones en la escena pueden producir pérdidas y desorden entre las franjas y pérdidas de puntos en los patrones mapa. Una forma de reducir este problema es incrementar la distancia de Hamming entre códigos. Esto se puede conseguir a través de varias estrategias: incrementando el conjunto de símbolos, permitiendo la repetición de símbolos en el código o incrementado el número de vecinos. Nótese que en todas las técnicas introducidas en la sección 1.1, excepto la de Morano, se trabaja con una distancia de Hamming de uno. Nosotros hemos diseñado un código de siete símbolos que pueden repetirse sin restricciones. Así

se consigue una codificación más robusta cuando se compara con códigos que utilizan una conectividad menor (2, 3, 4) o con códigos restringidos por la unicidad o la posición de los símbolos en el código.

En segundo lugar, la codificación basada en vecindad se suele definir siguiendo un orden específico. Dicho de otra forma, un código es un vector en el que sus componentes tienen una posición específica. Esta limitación implica que, tras la fase de segmentación, se debe llevar a cabo algún proceso adicional para ordenar los vecinos. Una forma de evitar este problema es definir el código, no como vector, sino como conjunto de datos. De esta forma el orden no está presente en el código y se evita el coste de la ordenación. Por lo tanto, nuestro patrón se ha construido mediante un conjunto de ‘códigos desordenados’.

En tercer y último lugar, el proceso de correspondencia (o identificación) patrón-imagen consiste normalmente en la búsqueda de un vector en una base de datos de vectores. Obviamente, si los argumentos son conjuntos en lugar de vectores, el algoritmo de búsqueda se simplifica sensiblemente y las situaciones de correspondencia parcial relativas a pérdida de color se pueden tratar con mayor eficiencia.

Este artículo se dedica a presentar este nuevo método. Para ello se estructura como sigue. En la sección 2 se introduce el patrón desordenado y en la 3 el sistema de recuperación de información 3D. La sección 4 analiza el problema de correspondencia patrón-imagen. La aplicación a la reconstrucción del movimiento 3D se muestra en la sección 5. Finalmente, las conclusiones del trabajo se exponen en la sección 6.

2 EL PATRÓN DESORDENADO

Nuestro patrón consiste en un *submapa perfecto* con conectividad-6 donde una ventana w se define como un conjunto de siete símbolos que verifican las siguientes propiedades:

1. w aparece una sola vez en el patrón.
2. w puede contener símbolos repetidos.
3. El orden en w es irrelevante.

En los párrafos siguientes se enumeran formalmente estas propiedades.

Sea M un conjunto de m puntos en un espacio bidimensional. M es un patrón de conectividad-6 si $\forall c \in M, \exists c_1, c_2, c_3, c_4, c_5, c_6 \in M$ que verifican:

$$\begin{aligned} I) \quad \overline{cc_j} &= \lambda, \quad j=1,2,\dots,6 \\ II) \quad \forall c_j, j=1,2,\dots,6, \exists c_l, c_k; \overline{c_j c_l} &= \overline{c_j c_k} = \lambda, \end{aligned} \quad (1)$$

donde $\overline{ab} = \|\vec{a} - \vec{b}\|$ y λ es un valor real. En este caso, la relación de vecindad V se define en M como:

$$\begin{aligned} V : M &\rightarrow M \times M \times M \times M \times M \times M \\ V(c) &= \{c_1, c_2, c_3, c_4, c_5, c_6\} \end{aligned} \quad (2)$$

Teniendo en cuenta esta definición hay varias propiedades topológicas que M satisface. $\forall c \in M$ y $\forall c_j \in V(c)$:

$$\begin{aligned} I) \quad c &\in V(c_j) \\ II) \quad \exists c_l, c_k \in V(c) \quad l \neq k; V(c) \cap V(c_j) &= \{c_l, c_k\} \end{aligned} \quad (3)$$

Estas propiedades afirman que cada par de puntos adyacentes tienen dos vecinos comunes. Esta estructura geométrica corresponde con una malla hexagonal.

Para diseñar un mapa de símbolos en M , utilizamos siete colores (rojo, verde, azul, magenta, cian, amarillo y blanco) asociando un color a cada punto de M . De ahora en adelante, denotaremos con C el color que ha sido asociado al punto c . Consecuentemente, el código que corresponda a un punto c consiste en el par $[C, V(C)]$ donde la topología y las propiedades para los puntos permanecen para los colores. Ciertamente hemos generado tal patrón estableciendo cada código como un conjunto de elementos. Esto es, los subíndices puestos en la ecuación (2) no presuponen ningún tipo de orden o posición relativa, son solamente una enumeración de los colores vecinos. Por tanto, $[b, bymyng]$ en la Figura 1 es equivalente a $[b, gbyymm]$.

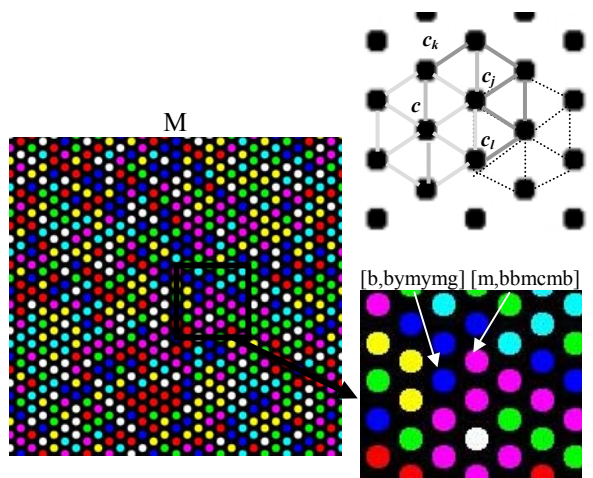


Figura 1. Patrón desordenado de color

Obviamente, el patrón debe ser único en el sentido de que cada código exista solo una vez en el patrón. El número de códigos desordenados de longitud l y h símbolos, en los que se permite repetición, es:

$$N = \binom{h}{1} \cdot C_{h,l-1}^R = \binom{h}{1} \cdot \binom{h+l-2}{l-1} \quad (4)$$

donde C^R denota combinaciones con repetición. En nuestro caso $l=h=7$ y $N=6468$. Sin embargo, el número de códigos que se pueden implementar en M es una cuestión que aun no hemos resuelto.

El color se ha introducido en M aplicando un algoritmo iterativo que comienza con una asignación al azar. En cada iteración, se detectan los códigos repetidos y se cambia al azar el color del nodo central. Se logra la convergencia del algoritmo ya que se permite colores repetidos en códigos desordenados y no hay restricciones sobre distancias de Hamming determinada. Así, la convergencia se alcanza fácilmente cuando m es bastante menor que N . En nuestro caso, $m < N/6$.

Hemos calculado la distancia de Hamming para varios patrones implementados obteniéndose que el valor medio de H es 3.16. Este valor ha sido calculado como sigue:

$$\bar{H} = \frac{1}{6m} \sum_{\forall C \in M} \left[\sum_{\forall C_j \in V(C)} [H(C, C_j) + H(V(C), V(C_j))] \right] \quad (5)$$

donde se considera que cada código tiene seis códigos vecinos y se pueden calcular seis distancias. Obviamente, la distancia mayor entre dos códigos solo puede ser (1+4) porque dos colores son siempre comunes. La figura 2 muestra la función de distribución para H . Los valores obtenidos son: $H(0)=0$; $H(1)=0,05$, $H(2)=0,20$, $H(3)=0,37$, $H(4)=0,30$, $H(5)=0,08$. Nótese que el 95% de estos códigos tienen una distancia de Hamming mayor que uno. Este es un resultado muy importante porque la mayoría de los códigos erróneos o perdidos pueden ser corregidos incrementando así el número de puntos codificados.

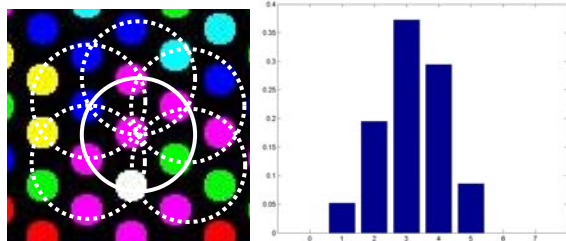


Figura 2. Cada código $[C, V(C)]$ tiene seis códigos vecinos pudiéndose calcular seis distancias. Distribución de la distancia de Hamming

3 SISTEMA DE OBTENCIÓN DE SUPERFICIES

Lo que sigue es una breve descripción de nuestro sistema de luz estructurada. Un color C correspondiente a un punto c de M , con coordenadas en el patrón (x_c, y_c) , se proyecta en el espacio 3D en un punto w , con coordenadas del mundo (x_w, y_w, z_w) . La cámara captura la escena y el punto c se registra en las coordenadas de la imagen (x_i, y_i) . Para recuperar (x_w, y_w, z_w) por medio de (x_c, y_c) y (x_i, y_i) se llevan a cabo dos tipos de calibración: la calibración de la cámara y la del proyector de luz (Figura 3).

En primer lugar, una calibración habitual de la cámara produce una relación entre el mundo y las coordenadas del ordenador (pixel) a través de la matriz de calibración de la cámara C_i .

$$\begin{bmatrix} \alpha x_i \\ \alpha y_i \\ \alpha \end{bmatrix} = C_i \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}, \quad C_i = \begin{bmatrix} A_{111} & A_{112} & A_{113} & A_{114} \\ A_{121} & A_{122} & A_{123} & A_{124} \\ A_{131} & A_{132} & A_{133} & A_{134} \end{bmatrix} \quad (6)$$

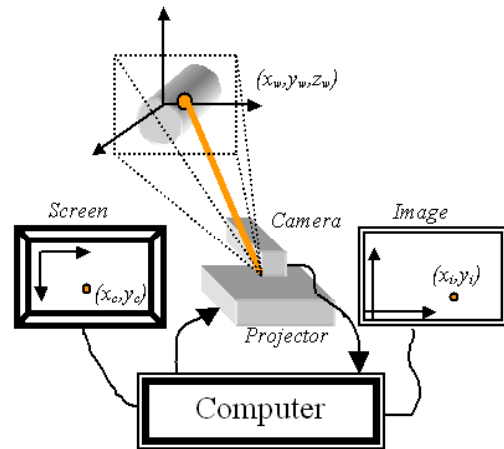


Figura 3. Sistemas de referencia

A continuación se realiza la calibración del proyector. Primero encontramos la relación entre las coordenadas del mundo y las del patrón proyectando haces de luz en puntos 3D conocidos. Para ello hemos desarrollado un software que nos permite proyectar puntos blancos en marcas específicas del cubo de calibración. Un usuario gobierna la proyección de un haz de luz blanca sobre la escena. Dado que ve la escena en un monitor externo, puede decidir cuándo el haz se proyecta en un punto deseado. En nuestro caso, es suficiente con iluminar las marcas de las esquinas del cubo de calibración, ya

que el resto de macas son iluminadas automáticamente. La figuras 6 a) y b) muestran el procedimiento. Se resuelve la siguiente ecuación (7) y se obtiene una segunda matriz de calibración C_c .

$$\begin{bmatrix} \beta x_c \\ \beta y_c \\ \beta \\ 1 \end{bmatrix} = C_c \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}, \quad C_c = \begin{bmatrix} A_{211} & A_{212} & A_{213} & A_{214} \\ A_{221} & A_{222} & A_{223} & A_{224} \\ A_{2131} & A_{232} & A_{2133} & A_{234} \end{bmatrix} \quad (7)$$

Si se conocen C_i y C_c y previamente se hacen corresponder sus respectivas coordenadas (x_c, y_c) y (x_i, y_i) , se puede obtener (x_w, y_w, z_w) para cada punto de la escena utilizando las ecuaciones (6) y (7). Después de manipularlas, se obtiene la siguiente expresión:

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = (A^T A)^{-1} A^T B \quad (8)$$

donde:

$$A = \begin{bmatrix} A_{111} - A_{131}x_i & A_{112} - A_{132}x_i & A_{113} - A_{133}x_i \\ A_{121} - A_{131}y_i & A_{122} - A_{132}y_i & A_{123} - A_{133}y_i \\ A_{211} - A_{231}x_c & A_{212} - A_{232}x_c & A_{213} - A_{233}x_c \\ A_{221} - A_{231}y_c & A_{222} - A_{232}y_c & A_{223} - A_{233}y_c \end{bmatrix} \quad B = \begin{bmatrix} A_{134}x_i - A_{114} \\ A_{134}y_i - A_{124} \\ A_{234}x_c - A_{214} \\ A_{234}y_c - A_{224} \end{bmatrix}$$

Hay que tener en cuenta que, para efectuar una correspondencia robusta entre los colores del patrón y de la imagen, se realiza una calibración del color al principio del proceso. Esto es, se debe establecer la relación entre el color proyectado por el proyector y el color captado. Nótese que las condiciones relativas a la reflectancia, la luz, la cámara o el capturador de imagen pueden modificar el color adquirido.

4 CORRESPONDENCIA PATRÓN-IMAGEN

Una vez que el patrón de color M se proyecta en la escena, una cámara CCD captura una secuencia de tomas y se lee el color de los círculos para cada toma. Este es un sencillo proceso de imagen ya que los círculos de color se encuentran sobre fondo negro y están separados suficientemente. Como resultado, la segmentación es inmediata. A continuación, se calculan las coordenadas de sus centros, (x_i, y_i) y se obtiene cada código $[C, V(C)]$. Dado que se extrae $V(C)$ directamente sin tener en cuenta ningún orden, en esta fase se obtiene una significativa reducción de coste computacional cuando se compara con otros métodos.

Sea B el conjunto de códigos obtenidos en esta fase y b el número de elementos. Si hubiéramos tenido que ordenar el código, se tendrían que haber completado

las siguientes fases. En primer lugar, se debería extraer alguna característica geométrica entre los puntos del código. Por ejemplo, el ángulo de $V(C)$ con respecto a C nos daría información de conceptos tales como ‘fila’, ‘columna’, ‘diagonal’, ‘cruce’, ‘arriba’, ‘abajo’, ‘derecha’, ‘izquierda’, etc. Esto implicaría realizar $b \cdot l$ operaciones elementales en el patrón (en nuestro caso $l=6$). Tras esto, si se aplicara un algoritmo de ordenación habitual en el código de longitud l , se necesitarían $(l^2-l)/2$ comparaciones y $l(\ln l + \gamma)$ movimientos, donde γ es la constante de Euler. Esto significaría, para el patrón completo, $b \cdot (l^2-l)/2$ comparaciones y $b \cdot l(\ln l + \gamma)$ movimientos. En resumen, nuestra estrategia evita este significativo coste computacional.

El problema de correspondencia se resuelve empezando por intersectar M con B . Sea $B_1 = M \cap B$, el conjunto de códigos completos que no son no erróneos,

$$c \in B_1 \Leftrightarrow [C, V(C)] \in M \text{ y } [C, V(C)] \in B.$$

Aunque B_1 sea el conjunto completo de códigos sin códigos erróneos, pueden surgir falsas correspondencias patrón-imagen. Sea $B_2, B_2 \subseteq B_1 \subseteq B$, el conjunto de códigos identificado en M . Es decir códigos con correspondencia cierta.

A continuación se lleva a cabo una fase de corrección de códigos erróneos y de recuperación de códigos perdidos sobre los conjuntos B_1-B_2 y $B-B_1$ incrementando B_2 hasta B_3 y B_3 hasta B_4 respectivamente. Formalmente:

$$B_3 = B_2 \cup (B_1 - B_2)^r \quad B_4 = B_3 \cup (B - B_1)^r \quad (9)$$

donde r denota códigos recuperados.

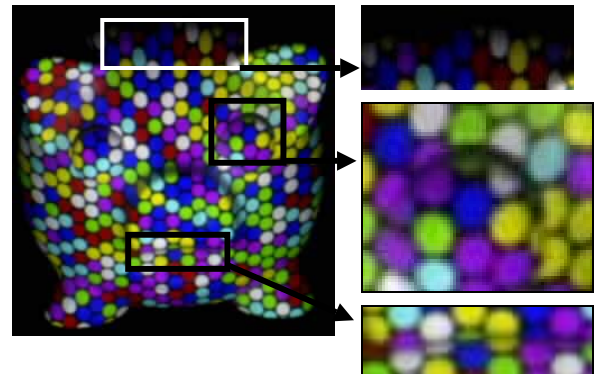


Figura 4. Ejemplo de color erróneo (arriba) y de punto perdido (abajo).

Los elementos de B_1 - B_2 corresponden a códigos completos pero con correspondencia errónea. Esto puede ser debido a problemas de lectura errónea de color o a generación de códigos ficticios por discontinuidades/sombras. Esta circunstancia afecta, en el peor de los casos, a los seis vecinos del punto erróneo. Afortunadamente, habiendo diseñado un *submapa perfecto* donde en la mayoría de los casos la distancia de Hamming es mayor que uno, se pueden recuperar y encontrar estas correspondencias. El conjunto B - B_1 corresponde a códigos incompletos en los bordes de la imagen del objeto. Para identificar tales puntos en el patrón M se consideran los elementos de M - B_1 que contienen parcialmente tales códigos y se utilizan las correspondencias previas para establecer una asignación basada en la situación dentro del patrón.

La figura 5, arriba, muestra un diagrama del proceso, la parte inferior ilustra la recuperación de códigos para un ejemplo.

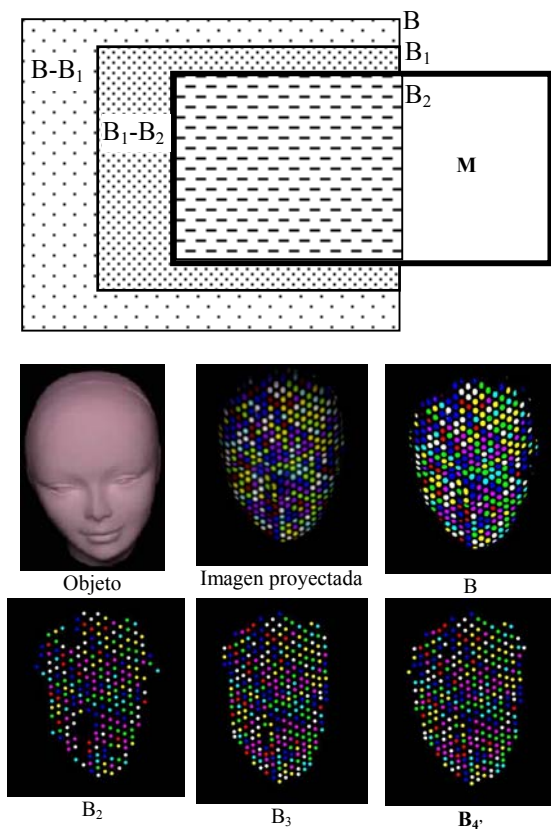


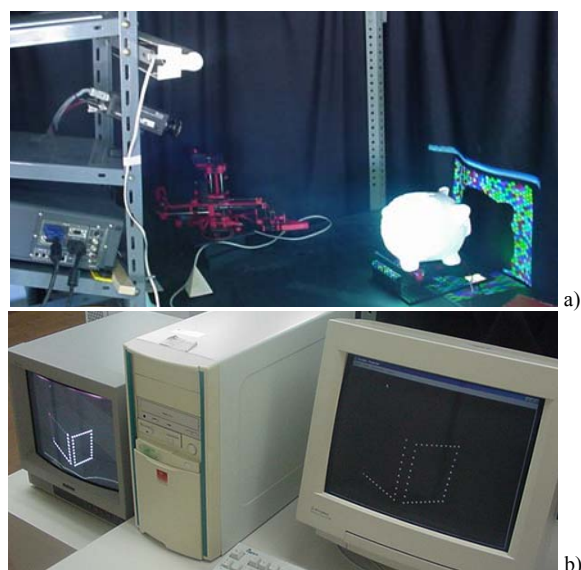
Figura 5. Fases del proceso de correspondencia (arriba). B_4 representa el conjunto de puntos puestos en correspondencia con el patrón M (abajo)

5 APLICACIONES PARA LA RECUPERACIÓN DE MOVIMIENTO 3D

Para validar el método presentado en las secciones previas hemos efectuado un amplio trabajo experimental. La plataforma del experimento consiste en una cámara Sony CCD XC-003P, un proyector 3M MP8755, un PC con una tarjeta capturadora MATROX CORONA y un monitor color externo analógico conectado a la cámara. La cámara se sitúa 25 cm verticalmente sobre el proyector (Fig. 6 a).

Se ha desarrollado un software específico que nos permite realizar las calibraciones de la cámara y el proyector. El sistema de referencia del mundo se establece por medio de un cubo de calibración cuando se calibra la cámara mientras que el sistema de referencia del patrón se define en la pantalla del ordenador. Como se dijo en la sección 3, en la fase de calibración del proyector, el usuario puede ver el cubo en un monitor externo y puede proyectar un haz de luz sobre un punto específico del cubo. Este proceso es semiautomático pues solamente es necesario iluminar manualmente las esquinas del cubo. Fig. 6 b) muestra una situación intermedia del procedimiento.

Para tener una iluminación ambiental controlada, la plataforma se cubre por completo con una cortina negra. A las figuras utilizadas en el experimento se les da un acabado mate con pintura blanca para obtener unas condiciones de reflectancia adecuadas. En estas condiciones, se ha experimentado con una amplia variedad de escenas, normalmente objetos aislados con unas dimensiones de unos 20 cm, que se mueven por medio de una plataforma con dos grados de libertad que nos permite combinar rotación y traslación. Hasta el momento se han registrado movimientos lentos de $40^\circ/s$ y 5 cm/s .



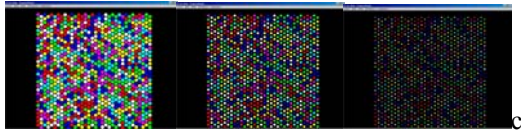


Figura 6. a) Plataforma experimental. b) Proceso de calibración del proyector. c) Patrones de diferentes radios.

El patrón utilizado en nuestra experimentación, consta de 986 puntos dispuestos sobre una malla hexagonal que pueden tomar siete colores: los seis colores básicos y el blanco. Para fijar el tamaño del patrón se ha tenido en cuenta el tamaño de la escena, las condiciones de la plataforma y los requisitos de nuestro experimento. Con esta configuración se extraen alrededor de 2 puntos por centímetro cuadrado en escenas distantes unos 100 cm. Obviamente se necesita registrar un número mínimo de píxeles por círculo para leer correctamente su color. Por lo tanto esta limitación se debe considerar antes de implementar el patrón. La Fig 6 c) muestra nuestro patrón para diferentes radios de círculos. Sin embargo, es importante resaltar que, dado el número teórico de códigos (6468), se pueden construir patrones con mayor número de círculos que el utilizado hasta ahora. De hecho, estamos trabajando con diferentes patrones para futuras aplicaciones.

La figura 7 (arriba) muestra detalles de la recuperación de puntos 3D de un objeto. A la derecha se ilustra una superficie en color ajustada a la nube de puntos 3D recuperados. La profundidad se puede apreciar a través de la escala de color de la representación. Un renderizado para varias tomas desde dos puntos de vista también se muestra en la parte inferior. En este caso el objeto se mueve de derecha a izquierda y de atrás hacia delante.

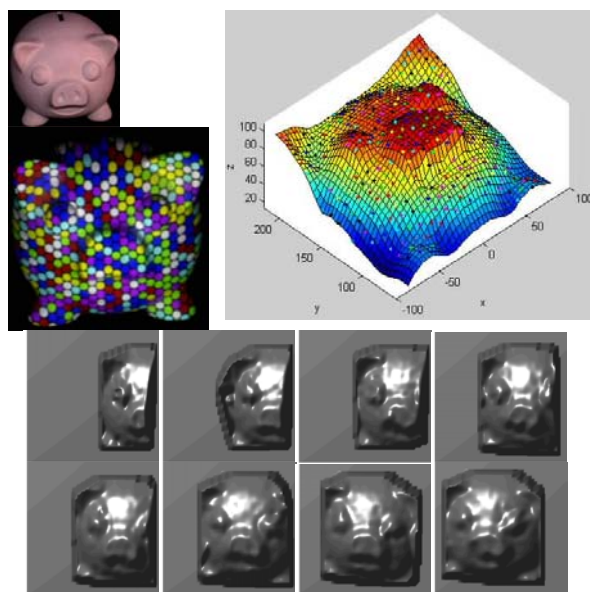


Figura 7. Imagen de intensidad, patrón de color proyectado en un objeto y puntos 3D correspondientes a los puntos del patrón junto a la superficie recuperada (arriba). Detalles de la recuperación 3D en una secuencia de varias tomas (abajo)

Tabla 1

Escena	1	2	3	4	5	6
B	353	355	313	522	346	208
B ₄	316	311	292	447	322	184
%rec.	90	87	93	86	92	88

La tabla 1 muestra los resultados obtenidos en la fase de recuperación de códigos perdidos para diversas escenas. Para cada escena se ha tomado una secuencia de 30 tomas y se calcula la media de puntos de B y B₄. Nótese que se identifica un alto porcentaje de puntos. Los casos de recuperación más difíciles corresponden a puntos de los bordes debido a que se pierden hasta cinco vecinos.

Finalmente, se muestran las reconstrucciones dinámicas 3D para dos objetos (escenas 3 y 4) en la figura 8. En este caso la escena rota enfrente de la cámara. También se muestran representaciones renderizadas e imágenes con el patrón proyectado para varias tomas. Como se puede ver, se consigue una aceptable recuperación de forma 3D.

6 CONCLUSIÓN

Hemos presentado un método de luz estructurada color basado en *submapas perfectos* donde el problema de correspondencia se resuelve fácil y robustamente. Esta técnica es adecuada para la extracción de información tridimensional en escenas dinámicas.

En lugar de los habituales mapas de conectividad 4/8 se adopta una malla de conectividad 6 en nuestro patrón. Por tanto definimos, para cada punto c del patrón, un código $[C, V(C)]$, siendo C el color básico

asociado a c y $V(C)$ los colores asociados a sus seis vecinos.

Al permitir repetición y no considerar orden en el código, se obtienen dos importantes ventajas. Primero, se incrementa significativamente la distancia media de Hamming del patrón, siendo $\bar{H} > 3$. Esto permite que, en general, el sistema sea más robusto frente a ruido, oclusiones y sombras. En segundo lugar, se simplifican las etapas de recuperación de código y el proceso de correspondencia, gracias a la ausencia de orden en los códigos.

Se ha construido en nuestro laboratorio un prototipo de sistema propuesto y se ha llevado a cabo un amplio test con condiciones dinámicas reales. La experimentación se ha desarrollado bajo especificaciones de resolución media y movimiento lento ofreciendo buenos resultados. Nuestro trabajo actual se centra en automatizar totalmente la calibración del proyector, incrementar la densidad del patrón y mejorar la técnica para futuras aplicaciones de seguimiento 3D e interacción con robot en ambientes dinámicos.

Agradecimientos

Esta investigación ha sido financiada por el proyecto CICYT DPI2001-0986.

REFERENCIAS

- [1] A. Adan, J. Garcia, A. Jimenez and F. Molina, "A 3D object shape recovering system for active scenes using color disordered patterns," ICIP03, *International Conference on Image Processing*, Barcelona, September 2003.
- [2] K.L. Boyer, A.C. Kak, "Color-encoded structured light for rapid active ranging," *Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 9 (1), pp. 14-28, 1987.
- [3] C.S. Chen, Y. P. Hung, C.C. Chiang, J.L. Wu. "Range data acquisition using color structured light and stereo vision," *International Journal on Image and Vision Computing*, 15, pp. 445-456, 1997.
- [4] C.J. Davies, M.S. Nixon. "A hough transform for detecting the location and orientation of three-dimensional surfaces via color spots" *Transactions on Systems, Man, and Cybernetics*, IEEE, 28 (1), pp. 90-95, 1998.
- [5] P.M. Griffin, L.S. Narasimhan, S.R. Yee. "Generation of uniquely encoded light patterns for range data acquisition," *International Journal on Pattern Recognition*, 25 (6), pp. 609-616, 1992.
- [6] O. Hall-Holt, S. Rusinkiewicz. "Stripe boundary codes for real-time structured-light range scanning of moving objects," *International Conference on Computer Vision*, pp. 359-366, 2001.
- [7] D. Hung. "3D Scene modeling by sinusoid encoded illumination". *Image and Vision Computing* 11, pp 251-256, 1993.
- [8] F.J. MacWilliams, N.J.A. Sloane, "Pseudorandom sequences and arrays". *Proc. of IEEE* 64 (12) 1725-1729. 1976.
- [9] M. Maruyama, S. Abe. "Range sensing by projecting multiple slits with random cuts" *Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 15 (6), pp. 647-651, 1987.
- [10] T.P. Monks. "Measuring the shape of time-varying objects," PhD Thesis. University of Southampton, 1994.
- [11] R.A. Morano, C. Ozturk, R. Conn, S. Dubin, S.Zietz, J.Nissanov, "Structured light using pseudorandom codes", *Transactions on Pattern Analysis and Machine Intelligence*, IEEE 20 (3) pp 322-327, 1998.
- [12] J. Salvi, J. Batlle, E. Mouaddib, "A robust-coded pattern projection for dynamic 3D scene measurement," *Pattern Recognition Letters*, ELSEVIER, 19, pp. 1055-1065, 1998.
- [13] J. Salvi, J. Pages, J. Batlle. "Pattern codification strategies in structured light systems". *Pattern Recognition* 37 (4), April 2004, pp 827-849.
- [14] P. Vuylsteke, A. Osterlinck. "Range image acquisition with a single binary-encoded light pattern," *Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 12 (2), pp. 148-164, 1990
- [15] C. Wust, D. W. Capson, "Surface profile measurement using color fringe projection". *Machine Vision and Applications* 4, pp 193-203, 1991.
- [16] L. Zhang, B. Curles, S.M. Seitz. "Rapid shape acquisition using color structured light and multi-pass dynamic programming," *Proceedings of 3D DPVT*, IEEE, pp. 532-535, 2002.

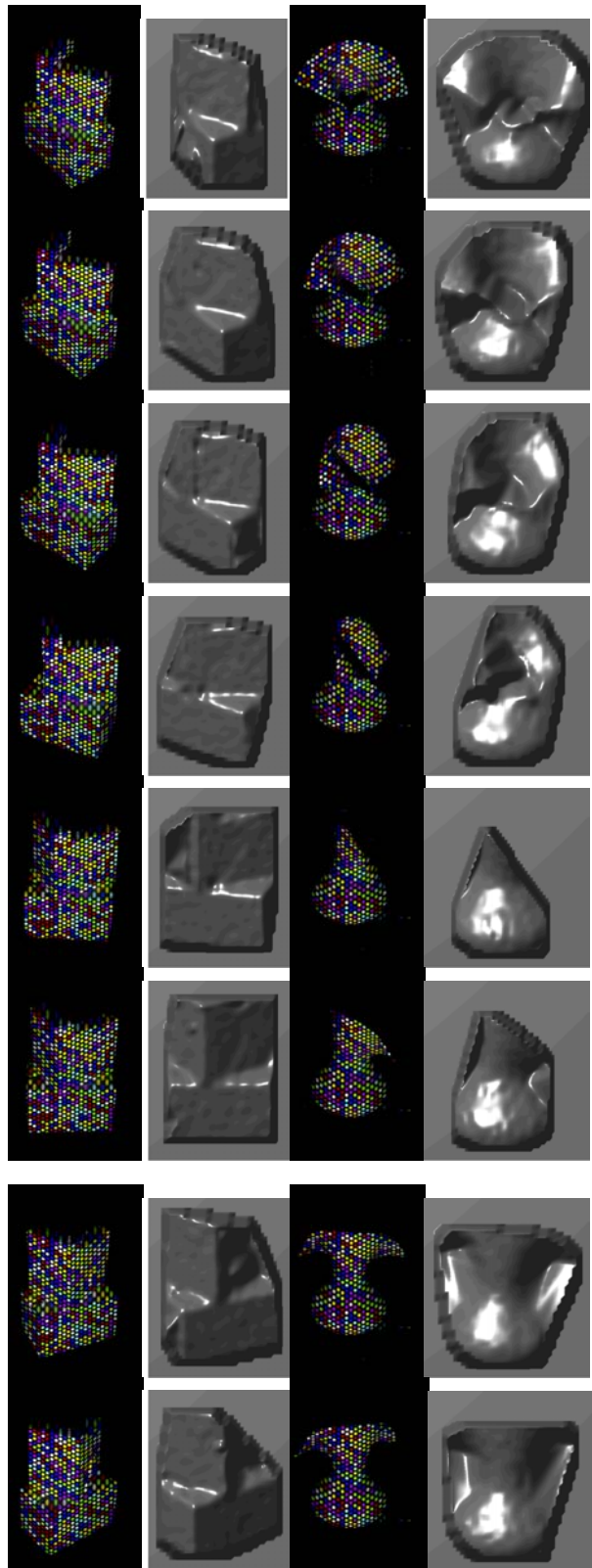


Figura 8. Reconstrucciones 3D de dos escenas dinámicas