

Mapeo de Texturas Usando Registro 3D-2D

Jorge Hernández * Flavio Prieto *

Fecha de Recibido: 27/04/2008; Fecha de Aprobación: 19/08/2008

Resumen

En este artículo se presenta el diseño y la implementación de un método de registro 3D-2D para el mapeo de textura a objetos del mundo real. La información de rango fue adquirida con el escáner de rango VIVID 9i Konica Minolta y la información de textura fue obtenida con la cámara CCD digital SONY, utilizando su máxima resolución de 2560×1920. La función objetivo diseñada para el registro 3D-2D, sólo utiliza 4 parámetros de cámara. El algoritmo de optimización utilizado fue el Levenberg-Marquardt, aproximando el jacobiano con diferencias finitas. Los resultados presentados muestran el gran nivel de detalle obtenido con la representación texturada.

Palabras clave: *Registro 3D-2D, Mapeo de Texturas, Modelo de Cámara.*

Abstract

In this paper we present the design and the implementation of a register method 3D-2D for texture mapping to objects of the real world. The range information was acquired with the no contact 3D digitizer Konica Minolta VIVID9i and the texture information was obtained with the camera CCD SONY DSC 717, using the maxim resolution of 2560×1920. The function objective designed for the register 3D-2D, only it uses 4 camera parameters. The used algorithm of optimization was the Levenberg-Marquardt, approaching the Jacobian with finite differences. The presented results show the high level of detail obtained with the texture representation.

Keywords: *Register 3D-2D, Texture Mapping, Camera Model.*

1 Introducción

La visión por computador y la computación gráfica son dos campos que se han venido combinando gradualmente, dando como resultado la realidad virtual (VR) [6]. Más y más usos utilizan métodos de la visión por computador para construir modelos computacionales con datos de objetos y/o escenas reales. Los datos entonces, se procesan usando algoritmos de la computación gráfica para realizar la representación y construcción de los modelos [22,23]. Las técnicas de reconstrucción 3D están divididas típicamente en dos grupos. El primero está compuesto

* Universidad Nacional de Colombia sede Manizales, Carrera 27 N° 64-60, Manizales, Caldas, Colombia, {jehernandezl,faprieto}@unal.edu.co

por técnicas activas basadas en sensores de rango que miden la distancia directamente, entre el sensor y puntos en el mundo real [31, 2]. Las técnicas en el segundo grupo son pasivas y reciben la información 3D de varias secuencias de fotografías o video digital (como visión estéreo o fotogrametría) [25, 4, 9, 8, 15]. Una evaluación comparativa de la actuación de sistemas de visión 3D, activos y pasivos, se presenta en [29]. Cada una de estas técnicas tiene ventajas y limitaciones diferentes. Los sensores de rango proporcionan de forma directa los puntos 3D de forma precisa, con poca dependencia de las condiciones de iluminación externa y sin necesidad de una textura determinada. Sin embargo, ellos tienden a ser caros, lentos al adquirir y normalmente tienen una resolución espacial limitada. Por otro lado, las imágenes de intensidad tienen alta definición, lo cual permiten resultados muy exactos con objetos bien definidos. Ellos son fáciles de adquirir y proporcionan los mapas de textura basados en las fotografías reales de color. No obstante, la falta de relación directa entre las imágenes y profundidad del objeto modelado es un problema mayor para las técnicas basadas en intensidad. Por esta razón, las imágenes adquiridas necesitan ser ricas en información de textura y el conjunto de datos resultantes tendrán una densidad variante dependiendo de las imágenes de intensidad. Desafortunadamente, estos sensores son afectados por los cambios en la iluminación. Como se puede observar la información de la imagen de rango y la imagen de intensidad es complementaria [7]. Así, nuestro objetivo en este trabajo es combinarlos para compensar las limitaciones de cada sensor con las ventajas del otro.

Normalmente, las imágenes de rango y de intensidad son obtenidas desde diferentes puntos de vista, de forma completamente independiente [27]. Así, para mapear esas imágenes de color hacia el modelo geométrico, es necesario determinar las relaciones relativas entre estos dos puntos de vista. Este problema, ha sido tratado como una clase especial de registro [1] llamado registro 3D-2D. Básicamente, el registro 3D-2D trata de encontrar la transformación proyectiva (transformación de cuerpo rígido y proyección perspectiva) en la cual una imagen 2D mapea a un modelo computacional de un objeto 3D de forma paramétrica [24].

2 Trabajo Relacionado

La ventaja del mapeo de texturas a modelos computacionales de objetos reales, es que el nivel de realismo de las texturas es muy alto, puesto que se utilizan las texturas reales de los objetos [30, 36]. Sin embargo, el mapeo de las texturas encontradas en las imágenes no es fácil de realizarse, especialmente cuando la adquisición de las imágenes de color y la construcción del modelo geométrico no se unen firmemente [6, 39].

Previamente se han desarrollado varias técnicas que tratan la reconstrucción del modelo 3D utilizando la textura. Sato, Wheeler y Ikeuchi en [37] reconstruyen las propiedades de reflectancia de los objetos del mundo real desde fotografías utilizando el modelo de *Torrance-Sparrow*. Ellos usan la misma cámara y la misma posición para adquirir las fotografías e imágenes de rango a partir de las cuales reconstruyen la geometría del modelo; por este motivo, un registro fotográfico no es necesario. Aunque los resultados son muy realistas, el proceso resulta trabajoso. Por ejemplo, para la adquisición de un jarrón cerámico se requirieron 120 fotografías, así como un sistema de adquisición de imagen especial con una sola fuente de luz, la cual debe ser calibrada previamente al proceso de adquisición. De igual forma, Pulli en [25] presenta un sistema de visión estéreo para la reconstrucción de la geometría y el color de una superficie desde un punto de vista determinado, adquiriendo la información de rango y la textura al mismo tiempo y de esta forma las imágenes de los dos sensores se encuentra alineadas completamente.

Muchos algoritmos intentan encontrar la transformación de la cámara minimizando el error entre el contorno de imagen de la textura y el contorno del modelo 3D proyectado [21, 20, 13, 14]. El error se computa como la suma de distancias entre varios puntos de muestreo de un contorno más cercanos al otro [21,20]. Matsushita y Kaneko en [20] describen un algoritmo para la selección del número mínimo de posiciones de cámara que pueden cubrir la superficie entera de un objeto dado; además, presenta el primer acercamiento del registro 3D-2D utilizando la información del contorno proyectado con el contorno de la textura. Ellos ajustan sólo los parámetros extrínsecos de la cámara, empleando un método iterativo de pequeños cambios graduales en los valores iniciales. Neugebauer y Klein en [21] formalizan en una función objetivo la métrica del error para ser minimizada por el algoritmo de Levenberg-Marquardt. Lensch, Heidrich y Seidel en [13,14] utilizan medida de distancia con la extracción de la silueta resultante entre la comparación de la forma de la textura y la forma del modelo proyectado utilizando una XOR píxel a píxel. La función a minimizar emplea el método *Downhill Simplex* el cual es una optimización no lineal.

Otros acercamientos, se enfocan en el mapeo de textura a modelos de ambientes urbanos [31,11,19,32,5,26,35]. El problema del registro de imágenes de textura, consiste en encontrar los segmentos de líneas 3D en el modelo, proyectándolos e igualándolos con los segmentos de línea en las imágenes, las cuales son propias de la geometría en la representación de la ciudad. La función para encontrar los parámetros de la cámara se basan en la distancia entre los segmentos líneas de los dos sensores. Bendels en [12], Kurazume en [27], Sequeira en [28], Beraldin en [16] y Dias en [7] encuentran correspondencias entre la imagen de intensidad y la imagen de rango utilizando el mapa de reflectancia que provee el digitalizador 3D a la hora de la adquisición.

Al establecer las parejas de correspondencias se utiliza el método de calibración de Tsai [33, 34] o alguna técnica de calibración estándar. Umeda en [18] calcula la matriz de la cámara utilizando restricciones de gradiente entre la intensidad de la imagen de rango y la imagen de color; y de esta forma se obtiene una ecuación lineal, con la cual se obtienen las estimaciones de los parámetros extrínsecos e intrínsecos.

3 Registro 3D-2D

3.1 Modelo de Cámara

Durante el registro, los parámetros de cámara deben ser determinados para la imagen que la mapea correctamente el modelo 3D. Una cámara realiza una transformación R^3 hacia R^2 , donde un conjunto de puntos en el espacio (\mathbf{x}) es mapeado a un punto en un plano (\mathbf{X}). El modelo de cámara más utilizado para estimar la transformación es el modelo pinhole (Fig. 1). Donde (x_w, y_w, z_w) es el sistema 3D coordenado del mundo, (x, y, z) es el sistema 3D coordenado de la cámara, C es el centro de proyección llamado centro de cámara o centro óptico y el plano de la imagen es el plano focal donde la componente de profundidad z es igual a la distancia focal f . Esta transformación de espacios $R^3 \leftrightarrow R^2$ está determinada por los parámetros tanto extrínsecos como intrínsecos, de la cámara [33]. Los parámetros extrínsecos determinan la transformación de cuerpo rígido del espacio coordenado del mundo al espacio de la cámara (Ec. 1).

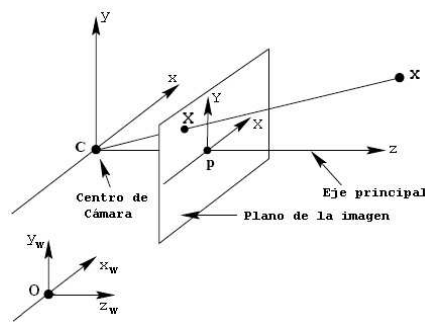


Figura 1. Modelo de Cámara

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = [R] \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + [T] \quad (1)$$

donde, $R(\theta_x, \theta_y, \theta_z)$ es la matriz de rotación y $T(t_x, t_y, t_z)$ es el vector de traslación. La transformación desde el sistema 3D coordenado de la cámara (x, y, z) a la imagen (X, Y) , se rige por los parámetros intrínsecos de la cámara. Los parámetros intrínsecos involucran la transformación

de perspectiva, y las relaciones entre píxeles y el mundo (Ec. 2).

$$\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} fm_x & 0 & m_x p_x & 0 \\ 0 & fm_y & m_y p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (2)$$

donde,

- f : es la distancia focal.
- m_x, m_y : son el número de píxeles por unidad de distancia.
- p_x, p_y : las coordenadas del punto principal en la imagen.

3.2 Descripción del Proceso

Los diferentes acercamientos presentados en la literatura [20, 21, 31, 13, 14, 27, 19, 38, 32, 7, 5, 18] muestran que la característica más relevante para ser utilizada, es el contorno de la imagen comparada con el contorno del modelo proyectado. De esta forma y utilizando el modelo de cámara descrito en la Sección 3.1, se observa que son 11 parámetros (5 intrínsecos, 6 extrínsecos) a ser calculados en cada imagen a rastrear. Este problema puede ser formulado utilizando una función objetivo apropiada, la cual expresa la calidad de un conjunto de parámetros de cámara determinado por la Ec. 3.

$$\varepsilon(f, m_x, m_y, p_x, p_y, \theta_x, \theta_y, \theta_z, t_x, t_y, t_z) \quad (3)$$

La función objetivo puede ser minimizada utilizando algún método de optimización. El proceso diseñado para realizar el registro 3D-2D se ilustra en la Fig. 2.

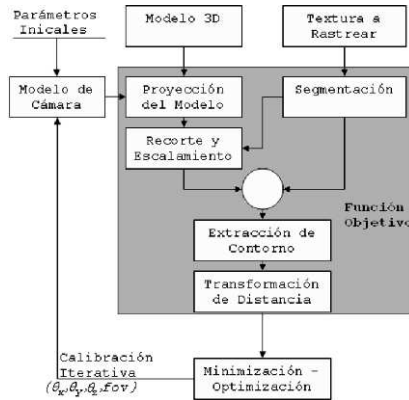


Figura 2. Proceso de Mapeo de una Textura utilizando Registro 3D-2D

El proceso comienza con la segmentación de la textura a rastrear, calculando la caja que envuelve el modelo. Por otro lado, el modelo 3D es proyectado utilizando un modelo de cámara inicial, preferiblemente con los valores del visor utilizado. La imagen proyectada se recorta y se escala a las dimensiones de la caja de segmentación de la textura. Este proceso se diseñó para disminuir el número de parámetros a optimizar en la función objetivo inicial, es decir, los parámetros intrínsecos como m_x, m_y, p_x, p_y y los parámetros extrínsecos t_x, t_y, t_z pueden ser omitidos al ajustar las dimensiones de la imagen de la proyección y a las dimensiones de la textura; y de esta forma la función objetivo es simplificada a solo 4 parámetros en la Ec. 4. Esta modificación reduce el costo computacional a la hora de la optimización.

$$\varepsilon(f, \theta_x, \theta_y, \theta_z) \quad (4)$$

Sin embargo, las unidades de la distancia focal (f) están dadas en metros y las unidades de los ángulos están dadas en grados o radianes. Por este motivo, otro cambio que se le realiza a la función objetivo es expresar la distancia focal f en términos del ángulo del campo de vista fov (*Field of View*) utilizando la Ec. 5. De esta forma, el dominio de la función presenta las mismas unidades.

$$fov = 2 \times \arctan\left(\frac{l_{max}}{f}\right) \quad (5)$$

donde, l_{max} es la máxima arista de la caja que contiene el modelo.

Después de igualar las dimensiones de las imágenes, se procede a la extracción de los contornos; para esta etapa se puede utilizar cualquier filtro pasa alto (Prewitt, Sobel) o alguna técnica de contornos activos, dependiendo del contraste entre el objeto y el fondo. El paso a seguir, es la transformación del contorno utilizando una transformación de distancia. La transformación de distancia juega un papel importante en la comparación de contornos binarios [3]. La transformación utilizada fue presentada por Felzenszwalb en [10], la cual requiere un costo computacional muy bajo a comparación de la transformación euclídea convencional. Posteriormente, se realiza la diferencia absoluta entre las imágenes transformadas, la cual es la salida de la función objetivo. La optimización de la función objetivo fue realizada con el algoritmo de Levenberg-Marquardt.

3.2 Parametrización de la Superficie

Al obtener los valores de $fov, \theta_x, \theta_y, \theta_z$, se procede a la parametrización de la superficie. Se calcula la máxima arista (l_{max}) de la caja que contiene

el modelo. Con este valor se determina la distancia focal utilizando la Ec. 6 con el campo de vista.

$$f = \frac{l_{\max}}{\tan\left(\frac{fov}{2}\right)} \quad (6)$$

Ahora, con los valores de θ_x , θ_y , θ_z se halla la matriz de rotación R . Se calcula la visibilidad de cada punto (x, y, z) rotado a la distancia focal f . Durante el cálculo de la visibilidad, se determinan los límites máximos (max_x , max_y) y mínimos (min_x , min_y) de la imagen de puntos proyectada. La parametrización de la superficie normalizada se encuentra descrita por la Ec. 7.

$$f(x, y, z) = \begin{cases} u = \frac{\frac{-fx}{z} - \min_x}{\max_x - \min_x} \\ v = \frac{\frac{-fy}{z} - \min_y}{\max_y - \min_y} \end{cases} \quad (7)$$

Un punto en el espacio es visible si el producto punto entre vector normal del punto y el vector de la cámara es igual o menor a cero, como lo muestra la Fig. 3 para los puntos azules y rojos. Sin embargo, esta condición es una condición necesaria pero no suficiente (punto amarillo). Por esta razón se utiliza el algoritmo Zbuffer [17]. El algoritmo Zbuffer de una nube de puntos, guarda la información de profundidad (distancia desde el plano de la imagen) del punto de la superficie más cercano sobre la proyección del punto en la imagen [36].

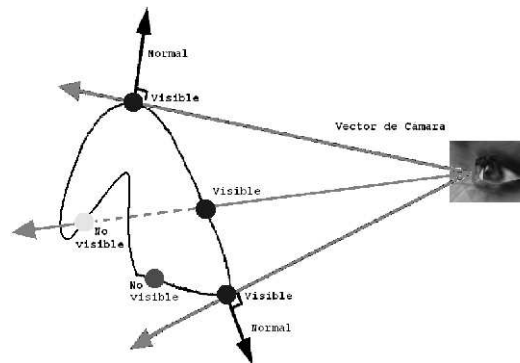


Figura 3. Visibilidad de los puntos

Aunque el algoritmo Zbuffer presenta una solución al problema del traslape cuando la imagen a proyectar es de dimensiones pequeñas y el

número de puntos es alto, no es suficiente en nuestro caso de trabajo, porque la mayoría de imágenes de textura adquiridas son de alta resolución. Por esta razón, se expandió el concepto de profundidad de puntos a profundidad de la superficie (triángulos) y el Zbuffer almacena la distancia de la cámara al baricentro del triángulo sobre toda el área de proyección del mismo.

4 Implementación y Resultados Experimentales

4.1 Adquisición

La escena de adquisición está constituida por el escáner de rango, la cámara CCD y el objeto a digitalizar, como se observa en la Fig. 4. El escáner de rango que posee la Universidad Nacional de Colombia, es el digitalizador 3D no invasivo VIVID 9i de Konica Minolta. El VIVID 9i es basado en la triangulación de un láser. Una cámara CCD recibe la luz reflejada desde la superficie. Las medidas de la superficie del objeto son obtenidas mediante triangulación y convertidas a una malla de polígonos. El VIVID 9i obtiene máximo 640×480 puntos por cada rastreo, adquiriendo simultáneamente la superficie y los datos de color de los puntos. Aunque el VIVID puede producir las imágenes de color así como la geometría 3D, se utilizó la cámara CCD digital Sony DSC - F717 para adquirir imágenes de alta calidad y resolución. La resolución de las imágenes adquiridas es de 2560×1920 .



Figura 4. Escena de Adquisición

Los algoritmos fueron desarrollados bajo el lenguaje C++. Durante el trabajo se desarrolló una plataforma para la visualización de los resultados. Esta plataforma se encuentra implementada utilizando la librería Qt, la visualización de los modelos se realizó usando la librería

OpenGL. El sistema donde se desarrollaron los algoritmos es un computador personal con procesador Pentium M de 2 GHz, 1Gb de memoria RAM, una tarjeta aceleradora de video nVidia GForce 6800 de 256Mb. Como el algoritmo de optimización usado fue Levenberg-Marquardt, se utilizó la librería levmar (Levenberg-Marquardt nonlinear least squares algorithms in C/C++); donde, para nuestro caso en particular el calculo del Jacobiano de la función objetivo, fue calculado utilizando diferencias finitas con cada una de las entradas.

4.2 Registro 3D-2D

El objeto a mapear es una figura precolombina (Fig. 5). Esta figura fue digitalizada y reconstruida (registro e integración) con la información de 14 imágenes de rango (Fig. 5(a)) utilizando el paquete PET (Polygon Editing Tool) y Geomagic Studio. La información de textura fue adquirida sobre un fondo completamente blanco (Fig. 5(b)). Los valores iniciales para este ejemplo son: $\text{fov} = 10^\circ$, $\theta_x = 0^\circ$, $\theta_y = 0^\circ$, $\theta_z = 0^\circ$.



(a) Modelo 3D



(b) Textura a Mapear

Figura 5. Figura precolombina

Siguiendo cada paso del proceso, con la imagen inicial de la textura (Fig. 5(b)) hallamos la caja (Fig. 6(a)) y segmentamos la imagen de la textura (Fig. 6(b)). El tamaño de la caja es de (1157×1510) .

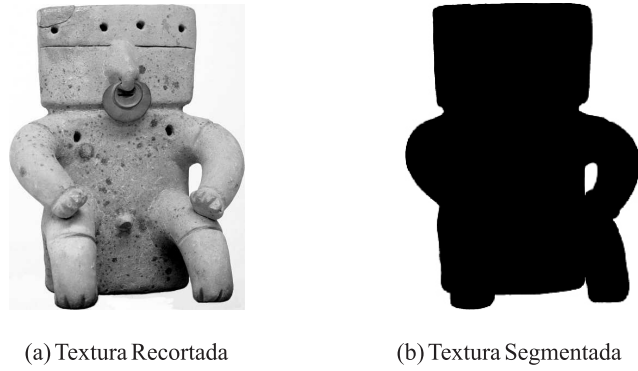


Figura 6. Proceso sobre la Textura

Luego, la proyección es recortada en sus límites dando una caja de dimensiones 322×571 (Fig.7(a)). Después, es escalada a las dimensiones de la textura (1157×1510).

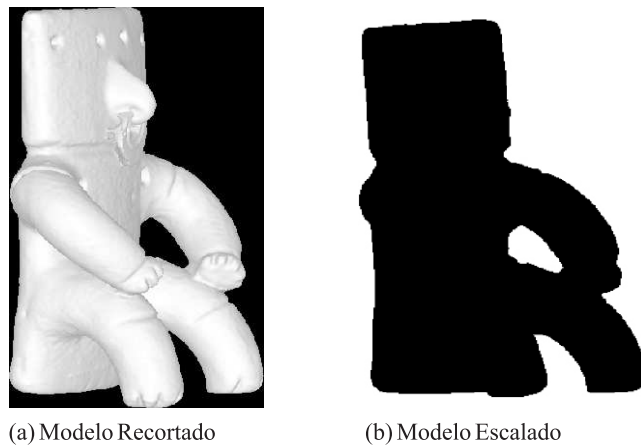


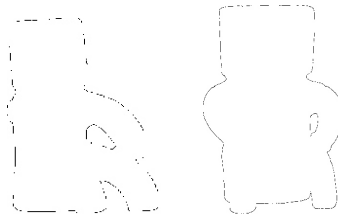
Figura 7. Proceso sobre la proyección del Modelo

Como puede observarse, en la posición inicial del modelo, la proyección no concuerda con el punto de vista de la textura. Esto se puede comprobar al hacer la diferencia absoluta de las figuras segmentadas (formas). La Figura resultante se observa en la Fig. 8.



Figura 8. Diferencia absoluta de formas del Modelo proyectado (Fig. 7(b)) y la Textura (Fig. 6(b))

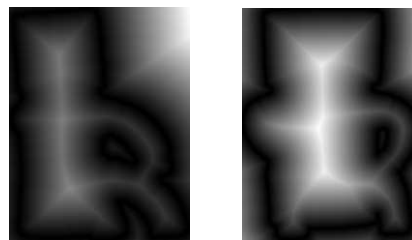
Ahora, se extraen los contornos con un filtro Sobel de cada una de las formas (Fig. 7(b) y Fig. 6(b)) dando como resultado la Fig. 9(a) y la Fig. 9(b) respectivamente.



(a) Contorno del Modelo (b) Contorno de la Textura

Figura 9. Extracción del Contorno

Luego se calcula la transformación de distancia dando como resultado la Fig. 10(a) y la Fig. 10(b).

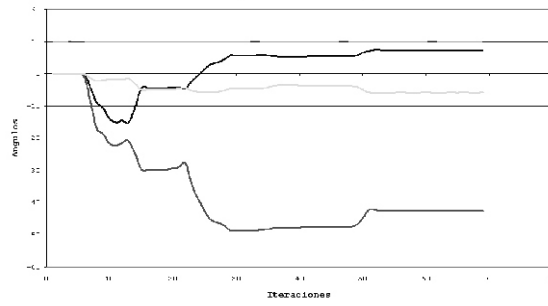


(a) TD del Modelo (b) TD de la Textura

Figura 10. Transformación de Distancia (TD)

La diferencia promedio de las transformaciones de distancia, para este punto de vista inicial del modelo es igual a 55,5838/255. Desde este momento, el proceso es iterativo al utilizar el algoritmo de optimización Levenberg-Marquardt para minimizar la diferencia en cada uno de los píxeles de las transformaciones de distancia.

El desempeño del algoritmo se observa en la Fig. 11. El algoritmo se detiene después de 71 iteraciones por no poder disminuir más la diferencia promedio igual a 3.5782/255; dando como resultado: $\theta_v = 10^\circ$, $\theta_x = 7,20373^\circ$, $\theta = -17.54015^\circ$, $\theta = -5.90880^\circ$



(a) Dominio de la Función Objetivo

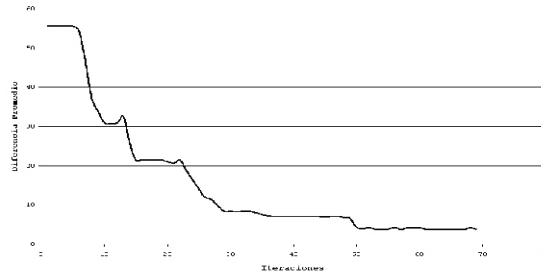


Figura 11. Desempeño del algoritmo de optimización

El mapeo de la textura sobre el modelo se observa en la Fig. 12.

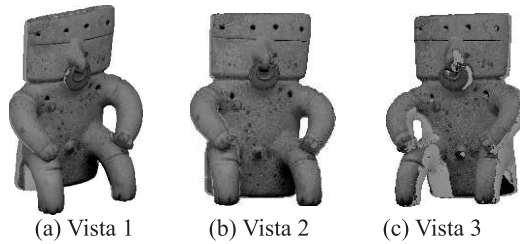


Figura 12. Modelo Texturado con el proceso de Registro 3D - 2D

4.3 Discusión

El método de mapeo de texturas utilizando registro 3D - 2D es un método semi-automático para mapear objetos cuyos contornos son bien definidos. Se define como método semiautomático por la selección manual de los parámetros iniciales antes de comenzar el proceso iterativo. Por esta razón, existen dos condiciones fundamentales a la hora de aplicar el método. La primera condición es que las texturas a rastrear deben contener el objeto en su totalidad y la segunda condición es que el modelo debe estar completamente reconstruido (registrado e integrado en una sola malla). Esto con el fin de hacer la comparación entre los mapas de distancias, producto de la proyección y la textura correspondiente.

Una de sus restricciones, es que no puede ser usado sobre objetos en los que al observar diferentes puntos de vistas, no se presenten grandes cambios, como sólidos de revolución (esferas, cilindros). En la Fig. 13 se observan las texturas de un objeto desde dos puntos de vista diferentes. Aplicar estas texturas sobre la superficie del modelo, la cámara de cada textura es la misma, lo cual no es verdad.

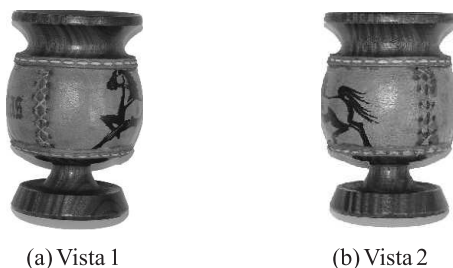


Figura 13. Textura de un objeto desde dos puntos de vista diferentes

La selección de los valores iniciales fov , θ_x , θ_y , θ_z juega un papel crucial; donde, la suma de las diferencias iniciales y finales de los ángulos θ_x , θ_y , θ_z no deben superar los 90° . De no ser así las posibilidades de convergencia del algoritmo son casi nulas. Esto se debe a que ningún algoritmo de optimización incrementaría el error para poder llegar a un mínimo aceptable.

Sobre el parámetro fov de la función objetivo el algoritmo de optimización no tiene mucho efecto por su alta no linealidad. Por este motivo, este valor se selecciona de forma manual en la parte inicial con múltiples pruebas. La no linealidad de la función objetivo presenta

muchos mínimos locales desde diferentes condiciones iniciales (Fig. 14).

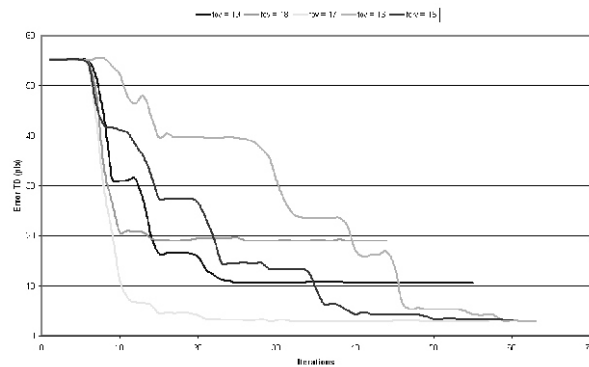


Figura 14. Análisis de mínimos locales de la función objetivo

En la Fig. 14 se muestra el desempeño del método utilizando diferentes parámetros iniciales, exactamente variando de a un grado el valor del fov que es el parámetro más no lineal. De esta forma se comprueba la presencia de mínimos locales en la función objetivo.

5 Conclusiones

Cualitativamente los resultados presentados por el método desarrollado de mapeo de textura son aceptables al comparar la superficie del modelo con la textura, especialmente en los lugares donde se presentan la mayor variación de la superficie. Por ejemplo, para el caso de la figura precolombina los huecos, los cuales coinciden en su totalidad. En conclusión, el método de mapeo de texturas utilizando registro 3D-2D, muestra una reducción del error inicial hasta en un 85% para el ejemplo presentado. La optimización de la función objetivo es estable desde las condiciones iniciales presentadas en la discusión. Sin embargo, por las propiedades no lineales de la función objetivo, la presencia de mínimos locales afecta el óptimo funcionamiento del algoritmo.

6 Agradecimientos

Este trabajo se desarrolló durante la tesis de maestría del primer autor, quien fue beneficiario del Programa de Becas para Estudiantes

Sobresalientes de Posgrado, de la Universidad Nacional de Colombia. La investigación se enmarca en el proyecto Modelado de superficies de forma libre empleando técnicas de visión artificial financiado por la Dirección Nacional de Investigación de la Universidad Nacional de Colombia (DINAIN).

Referencias

- [1] P. J. Besl and N. D. McKay. A method for registration of 3d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239-256, 1992.
- [2] Francois Blais. Review of 20 years of range sensor development. *Journal of Electronic Imaging*, 13(1):231-240, January 2004.
- [3] Gunilla Borgefors. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(6):849-865, November 1988.
- [4] C. Montani, C. Rocchini, P. Cignoni and R. Scopigno. Multiple textures stitching and blending on 3d objects. *Proceedings 10th Eurographics Workshop Rendering*, June 1999.
- [5] Russell Sammon, Christian Frueh and Avidah Zakhor. Automated texture mapping of 3d city models with oblique aerial imagery. *IEEE Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'04)*, 2004.
- [6] Nathalie Bertrand, Denis Laurendeau and Régis Houde. The mapping of texture on vr polygonal models. *Second International Conference on 3-D Imaging and Modelling (3DIM '99)*, pages 332-337, October 1999.
- [7] Paulo Dias, Vítor Sequeira, Francisco Vaz, and João G. M. Gonçalves. Registration and fusion of intensity and range data for 3d modelling of real world scenes. *Four International Conference on 3-D Imaging and Modelling (3DIM '99)*, pages 418-426, 2003.
- [8] Carlos Hernández Esteban and Francis Schmitt. Silhouette and stereo fusion for 3d object modeling. Signal and Image Processing Department - Ecole Nationale Supérieure des Télécommunications, France, May 2003.

- [9] Ioana M. Martin Fausto Bernardini and Holly Rushmeier. High-quality texture reconstruction from multiple scans. *IEEE Transactions on Visualization and Computer Graphics*, 7(4):318–332, October-December 2001.
- [10] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Distance transforms of sampled functions. *Technical report*, The University of Chicago, Cornell University, 2004.
- [11] Christian Früh and Avidesh Zakhor. 3d model generation for cities using aerial photographs and ground level laser scans. *Proceedings IEEE Computer Society Conference on Computer Vision And Pattern Recognition (CVPR '01)*, December 2001.
- [12] RolandWahl-Marcel Körtgen Gerhard Heinrich Bendels, Patrick Degener and Reinhard Klein. Image - based reistration of 3d-range data using feature surface elements. *The 5th International Symposium on Virtual Reality, Archaeology and Cultural Heritage*, 2004.
- [13] Wolfgang Heidrich Hendrik P. A. Lensch and Hans-Peter Seidel. Automated texture registration and stitching for real world models. *Proceedings 8th Pacific Conference on Computer Graphics and Applications*, pages 317–326, 2000.
- [14] Wolfgang Heidrich Hendrik P. A. Lensch and Hans-Peter Seidel. A silhouette-based algorithm for texture registration and stitching. *Graphical Models - IDEAL*, (63):245 262, 2001.
- [15] Carlos Hernández and Esteban Francis Schmitt. Using silhouette coherence for 3d image-based object modeling under circular motion. *Technical report*, Département Traitement du Signal et des Images CNRS UMR 5141 Groupe Traitement et Interprétation des Images, 2003.
- [16] S.F. El-Hakim G. Godin C. Latouche V. Valzano J-A Beraldin, M. Picard and A. Bandiera. Exploring a byzantine crypt through a high-resolution texture mapped 3d model: Combining range data and photogrammetry. *Proceedings of the International Workshop on Scanning for Cultural Heritage Recording - Complementing or Replacing Photogrammetry*, Septembre 2002.
- [17] S.K. Feiner J.D. Foley, A van Dom and J.F. Hughes. *Computer Graphics, Pirnciples ans Practice*. Addison-Wesley Publishing Company, 2nd edition, 1992.

- [18] Guy Godin Kazunori Umeda and Marc Rioux. Registration of range and color images using gradient constraints and range intensity images. *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, 3:23–26, August 2004.
- [19] Jung S. K. Lee S.C. and Nevatia R. Automatic pose estimation of complex 3d building models. *Workshop on Application of Computer Vision*, 2002.
- [20] Kenji Matsushita and Toyohisa Kaneko. Efficient and handy texture mapping on 3d surfaces. *Proceedings Euographics '99*, 18, 1999.
- [21] Peter J. Neugebauer and Korand Klein. Texturing 3d models of real world objects from multiple unregistered photographic views. *Proceedings Euographics '99*, 18, 1999.
- [22] Gregory M. Nielson. On marching cubes. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):283–297, July-September 2003.
- [23] I. Park, I. Yun, and S. Lee. Constructing nurbs surface model from scattered and unorganized range data. *Proceedings 2nd International Conference on 3-D Digital Imaging and Modelling*. Ottawa, Canada, October 1999.
- [24] J. Feldmar, N. Ayache and F. Betting. 3D-2D projective registration of free-form curves and surfaces. *Proceedings of the Fifth International Conference on Computer Vision (ICCV '95)*, pages 549–556, 1995.
- [25] Kari Pulli. Surface Reconstruction and Display from Range and Color Data. *PhD thesis*, University of Washington, 1997.
- [26] A. M. Day R.G. Laycock. Automatic techniques for texture mapping in virtual urban environments. *Proceedings of the Computer Graphics International (CGI'04)*, pages 586–589, June 2004.
- [27] Mark D. Wheeler Ryo Kurazume and Katsushi Ikeuchi. Mapping textures on 3d geometric model using reflectance image. *IEICE Trans. on Information and Systems*, J85-DII(6):1038–1046, June 2002.

- [28] Vitor Sequeira and joao G.M. Goncalves. 3d reality modelling: Photo-realistic 3d models of real world scenes. *Proceedings of the First International Symposium on 3D Data Processing Visualization and Transmission (3DPVT'02)*, page 776, 2002.
- [29] F. Blais S.F. El-Hakim, J.A. Beraldin. A comparative evaluation of the performance of passive and active 3-d vision systems. *SPIE Proc. 2646, St Petersburg Conf. on Digital Photogrammetry*, pages 14–25, June 1995.
- [30] Murali Subbarao Soon Yong Park. Automatic 3d model reconstruction based on novel pose estimation and integration techniques. *Image and Vision Computing - ELSEVIER*, (22):623–635, Janury 2004.
- [31] Ioannis Stamos and Peter K. Allen. 3d model construction using range and image data. *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'00)*, I:531–536, June 2000.
- [32] Ioannis Stamos and Peter K. Allen. Geometry and texture recovery of scenes of large scale. *Computer Vision and Image Understanding (CVIU)*, 8(22):94–118, November 2002.
- [33] Roger Y. Tsai. An efficient and accurate camera calibration technique for 3d machine vision. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Miami Beach, FL, pages 364–374, 1986.
- [34] Roger Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, August 1987.
- [35] Nathaniel Williams, Kok-Lim Low, Chad Hantak, Marc Pollefeys, and Anselmo Lastra. Automatic image alignment for 3d environment modeling. *In SIBGRAPI '04: Proceedings of the Computer Graphics and Image Processing, XVII Brazilian Symposium on (SIBGRAPI'04)*, pages 388–395, Washington, DC, USA, 2004. IEEE Computer Society.
- [36] F. Schmitt Y. Yemez. 3d reconstruction of real objects with high resolution shape and texture. *Image and Vision Computing - ELSEVIER*, pages 1–17, June 2004.

- [37] Mark D. Wheeler Yoichi Sato and Katsushi Ikeuchi. Object shape and reflectance modeling from observation. *Proceedings Computer Graphics (SIGGRAPH'97)*, pages 379–388, August 1997.
- [38] Yuuichi Omori Yuuya Iwakiri and Toyohisa Kaneko. Texture mapping on free-form surfaces immune to camera positioning errors. *Systems and Computers in Japan*, 33, 2002.
- [39] Jin Jin Zheng and Jian J Zhang. Texture mapping on irregular topology surface. *IEEE Proceedings of the Sixth International Conference on Information Visualization (IV02)*, 2002.