

Optimización de la representación con superficies NURBS de imágenes de rango

Ernesto Cuartas Morales^a, Flavio Prieto^{*a},

(a) Ingeniería Eléctrica, Electrónica y Computación,
Universidad Nacional de Colombia, Sede Manizales, Colombia
**faprieto@unal.edu.co*

(Recibido: Octubre 25 de 2007 - Aceptado: mayo 21 de 2008)

RESUMEN

La representación de modelos tridimensionales realísticos, a partir de una nube de puntos discretos obtenidos mediante digitalizadores 3D, define un compromiso entre el nivel de detalle y la carga computacional que se requiere para la visualización del modelo. Las superficies paramétricas permiten ajustar la cantidad de datos que se requieren para recrear el modelo 3D. En este trabajo se obtiene una aproximación optimizada de la superficie de un objeto, modificando los grados de libertad de la representación en NURBS, mediante el algoritmo de Levenberg-Marquardt.

PALABRAS CLAVE: Optimización, NURBS, Adaptación de Superficies, Levenberg-Marquardt.

Optimization of representation with range image NURBS surfaces

ABSTRACT

The representation of realistic 3D models, reconstructed from a cloud of discrete points obtained by the 3D-scanning of object's surface, holds a strict balance between the detail level and the computational load required for model visualization.

The parametric surfaces allow adjusting the quantity of data required to recreate the 3D model. The purpose of this work is to obtain and optimize the surface approximation of an object, modifying the representation of degrees of freedom in NURBS using the Levenberg-Marquardt algorithm.

KEYWORDS: Optimization, NURBS, Surface Adaptation, Levenberg-Marquardt.

1. INTRODUCCIÓN

Las B-Splines racionales no uniformes NURBS, son frecuentemente usadas para la parametrización de superficies de forma libre o geométricas comunes, como cónicas, debido a que permiten generar representaciones robustas y flexibles de las mismas. Otra razón para la aplicación de las NURBS en la recreación de modelos 3D es que este tipo de curvas paramétricas son soportadas por muchos paquetes de software como el OpenGL y el 3ds que ofrecen comandos para su creación y manipulación [1-3].

Debido a que muchos objetos del mundo real han sido contruidos usando superficies algebraicas simples y de forma libre, la representación mediante NURBS aparece como un procedimiento universal para el ajuste de superficies en la ingeniería inversa.

La reconstrucción de superficies es un proceso de ingeniería inversa [4], ampliamente utilizado por diversas disciplinas. Las aplicaciones se encuentran en el área de la manufactura, control de calidad, medicina, diseño asistido por computador y realidad virtual, entre otros. El modelado 3D tiene varias etapas que van desde la adquisición, hasta la integración del modelo [7]. Una sola imagen de rango, provee información parcial de la superficie del objeto. Por lo tanto, es necesario combinar múltiples imágenes de rango, desde diferentes puntos de vista en un mismo espacio coordinado para obtener un modelo completo. Este procedimiento involucra los procesos de registro e integración [5, 6]. El proceso de integración tiene como objetivo encontrar la superficie que mejor se aproxime a la nube de puntos obtenida en la etapa de registro.

El desarrollo tecnológico en los digitalizadores 3D permite capturar la superficie de un objeto con una mayor resolución en cada nueva generación de instrumentos, esto incide directamente en el número de muestras obtenidas que regularmente es demasiado grande para ser eficientemente procesado en un ordenador promedio. Por lo tanto, es necesario llegar a una representación optimizada del modelo que muestre un

compromiso entre el nivel de detalle de la representación y la carga computacional del sistema [9, 10].

Este trabajo busca, explora y aporta una solución a este problema, mediante la implementación de un algoritmo que permite optimizar la representación en modelos NURBS de la superficie muestreada de un objeto físico. Con el ánimo de cumplir con este objetivo, se lleva a cabo un proceso de optimización mediante el algoritmo de Levenberg-Marquardt, en el cual se manipulan los grados de libertad del modelo NURBS (Puntos de control, pesos y vectores paramétricos de nodos), para minimizar el error entre las muestras de la superficie y el modelo NURBS.

2. ANTECEDENTES

Las superficies complejas simples son adaptaciones por partes de elementos básicos tales como parches triangulares. A partir de estos elementos se genera una malla intrincada cuyo objetivo es dar la mejor estimación de la superficie del objeto real. Las desventajas que tienen las mallas poligonales simples es que generan un modelo sin una representación suave del objeto, por ello, la calidad de la representación depende de la cantidad de polígonos con los cuales se recrea el modelo [10, 12].

La representación mediante superficies implícitas, posee una sola ecuación definida de manera global permitiendo caracterizar la forma a través de un conjunto pequeño de parámetros, la desventaja recae en la necesidad de especificar previamente el tipo de superficie [11, 16].

En [18] se optimiza la disposición del vector de nodos mediante minimización de funcionales de energía, estas técnicas presentan un costo computacional relativamente bajo pero los resultados del mapeo paramétrico no son los mejores.

En la literatura se encuentran diversos trabajos que emplean superficies explícitas para aproximar datos de rango, pero no se encontraron trabajos en los cuales se diera a conocer una medida estimativa del error en la aproximación.

3. NURBS

Las curvas y superficies NURBS evolucionaron a partir del trabajo pionero de Pierre Bézier, quien planteó la formulación matemática de una clase de curvas paramétricas, para crear gráficas en un ordenador, en el año de 1970 [8, 17].

La digitalización mediante NURBS se ha convertido en un estándar en la industria para la representación, diseño e intercambio de datos de información geométrica procesados por ordenador, debido a sus excelentes cualidades [1, 16]. La ecuación (1) ilustra la forma general de las superficies NUBS.

$$S(u, v) = \frac{\sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) w_{i,j} P_{i,j}}{\sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) w_{i,j}} \quad (1)$$

4. OPTIMIZACIÓN

En este trabajo se desarrolló una optimización por aproximación, minimizando el error cuadrático medio entre la nube de puntos y la superficie NURBS, al modificar los grados de libertad de la formulación paramétrica.

4.1 Formulación del Problema

Dada una secuencia de muestras ruidosas (u_i, M_i) con $i = 0, \dots, m$, se pretende encontrar la superficie $X(u, v) = X_{W,D,T}(u, v)$ que mejor se adapte a las muestras M_i , minimizando el error cuadrático medio entre éstas y modificando los grados de libertad de la representación

$$\min_{W,D,T} \sum_{i=0}^m \|X_{W,D,T}(u_i, v_i) - M_i\|^2 \quad (2)$$

Donde W son los pesos, D son los puntos de control y T es el vector de nodos U .

Este problema es muy difícil de resolver debido a la dependencia no lineal de X con los parámetros

W, D, T . Además, se deben tener en cuenta algunas condiciones de variación que garanticen la correcta formulación matemática de las NURBS.

El proceso de optimización aplicado se basa en el trabajo realizado por Randrianarivony y Brunet [14, 19]. En éste dan solución al problema simplificando la función de costo y adicionando algunas penalizaciones que garanticen la formulación de las funciones NURBS.

4.2 Optimización Lineal (Pesos y Puntos de Control)

Si se tiene una secuencia de nodos T dada, se puede disminuir el problema de la ecuación (2), así:

$$\min_{W,D} \sum_{i=0}^m \|X_{W,D,T}(u_i, v_i) - M_i\|^2 \quad (3)$$

En [13] se propone una optimización lineal que da solución a la ecuación (3), minimizando los pesos y los puntos de control.

El sistema lineal está dado por

$$(A + \lambda B)x = \lambda b \quad (4)$$

En donde

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \in R^{4n+4, 4n+4}$$

Siendo

$$A_{11} = \begin{pmatrix} \Sigma \bar{N}_{00}^i (I - c_i P_i P_i^T) & \cdots & \Sigma \bar{N}_{0n}^i (I - c_i P_i P_i^T) \\ \vdots & \ddots & \vdots \\ \Sigma \bar{N}_{n0}^i (I - c_i P_i P_i^T) & \cdots & \Sigma \bar{N}_{nn}^i (I - c_i P_i P_i^T) \end{pmatrix} \in R^{3n+3, 3n+3}$$

$$A_{12} = A_{21}^T = \begin{pmatrix} -\Sigma c_i \bar{N}_{00}^i P_i & \cdots & -\Sigma c_i \bar{N}_{0n}^i P_i \\ \vdots & \ddots & \vdots \\ -\Sigma c_i \bar{N}_{n0}^i P_i & \cdots & -\Sigma c_i \bar{N}_{nn}^i P_i \end{pmatrix} \in R^{3n+3, n+1}$$

$$A_{22} = \begin{pmatrix} \Sigma(1-c_i)\bar{N}_{00}^i & \cdots & \Sigma(1-c_i)\bar{N}_{0n}^i \\ \vdots & \ddots & \vdots \\ \Sigma(1-c_i)\bar{N}_{n0}^i & \cdots & \Sigma(1-c_i)\bar{N}_{nn}^i \end{pmatrix} \in R^{n+1,n+1}$$

Con

$$B = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{B} \end{bmatrix} \in R^{4n+4,4n+4}$$

$$\tilde{B} = \begin{pmatrix} \Sigma\bar{N}_{00}^i & \cdots & \Sigma\bar{N}_{0n}^i \\ \vdots & \ddots & \vdots \\ \Sigma\bar{N}_{n0}^i & \cdots & \Sigma\bar{N}_{nn}^i \end{pmatrix} \in R^{n+1,n+1}$$

Las demás variables de la ecuación (4) están dadas por

$$b = [0, \dots, 0, \Sigma N_0^k(u_i), \dots, \Sigma N_n^k(u_i)]^T$$

$$\bar{d}_i = [w_i d_{ix}, w_i d_{iy}, w_i d_{iz}]^T$$

$$\bar{N}_{pq}^i = N_p^k(u_i) N_q^k(u_i)$$

$$c_i = 1/(1 + P_i^2)$$

La solución del sistema está dada en

$$y = [\bar{d}_0, \dots, \bar{d}_n, w_0, \dots, w_n]^T \quad (5)$$

En el vector y de la ecuación (5) se encuentran los puntos de control racionales $\bar{d}_i$ y w_i los pesos optimizados, donde n el orden de las funciones base Splines $N_n^k(u_i)$.

El orden del sistema lineal de la ecuación (4) no depende del número de puntos de control P_i . El rango de las matrices depende únicamente del grado n de las funciones base.

4.3 Optimización no Lineal (Vector paramétrico de nodos)

Al definir W y D en función de T en la ecuación (3) se tiene:

$$\min_T \sum_{i=0}^m \|X_T(u_i, v_i) - M_i\|^2 \quad (6)$$

La solución de la ecuación (6) está sujeta a la restricción de una solución con una secuencia T de nodos no decrecientes. Adicionalmente, se debe llevar el problema a una forma típica de reducción por mínimos cuadrados. Para ello se tiene:

$$X_T(u, v) = (X_{T,x}(u, v), X_{T,y}(u, v), X_{T,z}(u, v))$$

$$M_i = (M_{ix}, M_{iy}, M_{iz})$$

Con el fin de realizar una expansión lineal de los datos, se define el arreglo vectorial:

$$S = \begin{cases} S_{3i}(T) & = X_{T,x}(u, v) - M_{ix} \\ S_{3i+1}(T) & = X_{T,y}(u, v) - M_{iy} \\ S_{3i+2}(T) & = X_{T,z}(u, v) - M_{iz} \end{cases}$$

Reemplazando S en la ecuación (6) se obtiene

$$\min_T \sum_{i=0}^{3m+2} S_i^2(T) \quad (7)$$

La solución de la ecuación (7) debe garantizar una secuencia de nodos T creciente, para ello se define la función

$$R(x) = \begin{cases} 0 & \text{si } x > 0 \\ (-x)^3 & \text{si } x \leq 0 \end{cases}$$

Adicionalmente se plantea la penalización:

$$R(T) = R(u_{p+2} - u_{p+1}) + \cdots + R(u_{m-1} - u_{m-2}) \quad (8)$$

Al introducir la penalización $R(T)$ en la ecuación (7) se obtiene:

$$\min_T \sum_{i=0}^{3m+2} [S_i(T) + \alpha R(T)]^2 \quad (9)$$

En donde α es un número positivo muy grande. Si T es una secuencia de nodos creciente, entonces $\alpha R(T) \neq 0$. En caso contrario, es decir, si existe algún nodo $u_{i+1} > u_i$, entonces $R(T) \neq 0$, y debido a

que α es muy grande, la ecuación (9) es también muy grande y nunca se podrá realizar este mínimo. Finalmente, se define $r_i(T) = S_i(T) + \alpha R(T)$

$$\min_T \sum_{i=0}^{3m+2} [r_i(T)]^2 \quad (10)$$

La ecuación (10) tiene una forma típica de optimización por mínimos cuadrados y puede resolverse por un método basado en estos, como el Gauss-Newton o el Levenberg-Marquardt.

4.4 Levenberg-Marquardt

El algoritmo de Levenberg-Marquardt ofrece un método iterativo con velocidad de convergencia adaptativa para la optimización de modelos no lineales basados en el error cuadrático medio [14, 15].

Se debe encontrar el Jacobiano de la función vectorial $r(T)$, con el fin de aplicar el algoritmo de Levenberg-Marquardt:

$$r(T) = [r_0(T), r_1(T), \dots, r_{3m+2}(T)] \quad (11)$$

Donde $T = (\tau_1, \dots, \tau_l)$ y $l = m - p$ siendo m el número de puntos de control y p el orden de las funciones base.

La matriz Jacobiana $J(T)$ está dada por

$$J_{ij}(T) = \frac{\partial r_i(T)}{\partial \tau_j} \quad (12)$$

En la práctica, la derivación de la función $r(T)$ es muy difícil. En este trabajo se calculó la matriz Jacobiana por medio de aproximaciones finitas de la forma:

$$\frac{\partial r_i(T)}{\partial \tau_s} \cong \frac{r_i(T + \varepsilon e_s) - r_i(T)}{\varepsilon} \quad (13)$$

En donde $e_s = [0, \dots, 0, 1, 0, \dots, 0]$ toma sólo valor 1 en la s -th entrada.

Adicionalmente, se define la función de suma de cuadrados

$$f(T) = 2 \sum [r_i(T)]^2$$

La forma iterativa del algoritmo de Levenberg-Marquardt implementada en el desarrollo de este trabajo es la siguiente:

1. Se inicializan las variables de convergencia $\alpha_0 > 0$, $\gamma > 1$, $0 < \beta < 1$ y se toma una secuencia inicial de nodos uniforme T_0 .
2. Se hace $k=0$ y se calcula $f_0 = f(T_0)$.
3. Se calcula, $J_k = J(T_k)$, $r_k = r(T_k)$ y $Q_k = J_k^T J_k$.
4. Se resuelve el sistema lineal $(Q_k + \alpha_k I)u_k = -g_k$ y se calcula la penalización $R(T_k + u_k)$.
5. Si $R(T_k + u_k) = 0$ se calcula $f_{k+1} = f(T_k + u_k)$. Si no, $\alpha_k = \gamma \alpha_k$ y regresa al paso 4.
6. Se calcula $\sigma_k = (f_{k+1} - f_k) / [g_k^T u_k + 1/2(u_k^T Q_k u_k)]$.
7. Si $\sigma_k < \beta$, $\alpha_k = \gamma \alpha_k$ y regresa al paso 4. Si no, $T_{k+1} = T_k + u_k$, $\alpha_k = \alpha_k / \gamma$.
8. Se incrementa K y se regresa al paso 3.

5. ANÁLISIS Y RESULTADOS

La optimización desarrollada para el trabajo tiene varias restricciones. El proceso es aplicado a datos de rango organizados; por su formulación, los datos para la creación de una superficie NURBS deben estar dispuestos en una matriz rectangular.

Se encontró que con funciones base de grado elevado el tiempo de cálculo del algoritmo aumentaba y se presentaban problemas de estabilidad en el algoritmo.

5.1. Desempeño de la Técnica

Para analizar el desempeño de la técnica se aplicó el proceso de optimización a diferentes tipos de superficies de prueba. Como prueba final se utilizó una imagen de rango real de la base de datos de la Universidad de Ohio.

Tabla 1: Desempeño de la técnica en datos ruidosos. Conjunto de datos Plano, expansión de 100 x 100 mm.

Ruido	Error Promedio	Error Máximo	Varianza	Tiempo
0%	0.0042 mm	0.0404 mm	0.0001	123 s
1%	0.0996 mm	0.3661 mm	0.0157	256 s
10%	0.9410 mm	3.1373 mm	1.4289	262 s
30%	2.3323 mm	8.8598 mm	8.9584	261 s
50%	5.0223 mm	47.255 mm	74.362	258 s

Tabla 2: Desempeño de la técnica, muestreo simple. Conjunto de datos Seno, expansión de 100 x 100 mm.

Muestreo	Datos	Error Promedio	Varianza	Tiempo
100%	900	0.1327 mm	0.0303	1911 s
80%	576	0.2839 mm	0.1538	1088 s
60%	324	0.5362 mm	0.5392	445 s
40%	144	0.9298 mm	1.6326	120 s

Se segmentó la imagen de rango para obtener una matriz rectangular sin puntos redundantes que permitiera aplicar el proceso de optimización.

En la Tabla 1 se puede apreciar que el desempeño de la técnica es bueno incluso para datos con un nivel de ruido elevado (hasta del 50%).

5.2. Muestreo de los datos

Se aplicó el proceso de optimización a diferentes porcentajes del número total de datos, haciendo un muestreo simple del conjunto de datos Seno de la Figura 1. Los resultados del desempeño para esta figura se presentan en la Tabla 2.

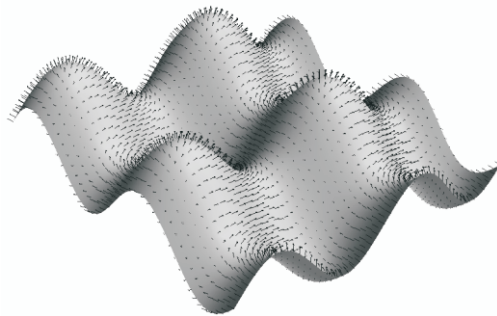


Figura 1: Superficie NURBS. Conjunto de datos Seno con una expansión de 100 x 100 mm

6. CONCLUSIONES

Las funciones NURBS presentan una formulación matemática unificada que permite su fácil implementación y digitalización. Además, el alto grado de adaptación y la capacidad de aproximación local hacen de las superficies NURBS una herramienta poderosa para generar modelos computacionales de objetos tridimensionales a partir de datos de rango.

La elección de los parámetros iniciales adecuados en el algoritmo de Levenberg-Marquardt garantiza una rápida convergencia. Se debe definir un compromiso entre las constantes de velocidad inicial y de ajuste, ya que la escogencia de estos parámetros es crucial en el desempeño del algoritmo.

El método de Levenberg-Marquardt presenta cualidades óptimas para la minimización de los vectores paramétricos de nodos en funciones NURBS. Sin embargo, el costo computacional del algoritmo es demasiado alto para ser aplicado a un número elevado de datos de rango.

La parametrización mediante NURBS para aproximar superficies a conjuntos de datos con demasiados puntos de control, tomando sólo

algunos de estos y ofrece un método de simplificación que garantiza una representación suave y continua. Además, al aplicar el proceso de optimización para aproximar la superficie al conjunto de datos original, da como resultado una simplificación de alta calidad.

7. REFERENCIAS BIBLIOGRÁFICAS

- [1] G. Farin, *Shape*, Springer-Verlag, New York, 3 edition, 2001.
- [2] G. Roth, P. Boulanger, *Cad model building from multiple range images*, in Proceedings of Vision Interface 98, pp. 274281, June 1998.
- [3] I. Soderkvist, *Introductory overview of surface reconstruction methods*, 1999.
- [4] T. Várady y M. Ralph, *Reverse engineering of geometric models*, 1997.
- [5] Ch. Yang and M. Gerard, *Surface description of coplex objects from multiple range images*, Technical report, Institute for Robotics and Intelligent Systems, University of Southern California, 1994.
- [6] J. Valencia y J. Gómez, *Registro de imágenes de rango para la reconstrucción de modelos faciales*, 2004.
- [7] J. B. Gómez y F. Prieto, *Reconstrucción de superficies a partir de imágenes de rango*, 2004.
- [8] T. Wayne, *The NURBS Book*, Springer-Verlag, New York, 2nd edition, 1997.
- [9] P. J. Besl and N. D. McKay, *A method for registration of 3d shapes*, 1992.
- [10] H. Huges, *Surface reconstruction from unorganized points*, Technical report, University of Washington, 1993.
- [11] I. Park and I. Yun, *Constructing nurbs surface model from scattered and unorganized range data*, in International Conference on 3D Imaging and Modeling, October 1999.
- [12] H. Huges, *Mesh optimization*, Technical report, University of Washington, 1993.
- [13] E. Bernhard, *Approximation whit Rational B-splines Curves and Surfaces*, Ph.D. thesis, Vanderbilt University, 1998.
- [14] M. Randrianarivony y G. Brunnett, *Parallel implementation of surface reconstruction from noisy samples*, 2002.
- [15] L. Michael, *Damping-undamping strategies for the levenberg-marquardt nonlinear least-squares*, 1996.
- [16] G. Farin, *Curves and Surfaces for CAGD*, Morgan Kaufmann Publishers, 5 edition, 2002.
- [17] P. Robert y C. Linda, *Generalized bernstein polynomials and symmetric functions*, 2001.
- [18] H. Xie and H. Qin. *Automatic knot determination of nurbs for interactive geometric design*, in Proceedings of International Conference on Shape Modelling and Applications (SMI 2001), Nov. 2001.
- [19] M. Randrianarinovy y G. Brunnett, *Parallel implementation of curve reconstruction from noisy samples*, 2002.