



COMPRESSION DE COURBES ET SURFACES

F. Derrien, A. Zarouf, O. Soligon et A. Le Méhauté

Dépt. Image et Traitement de l'Information
ENST de Bretagne - BP 832
29285 Brest Cédex

Nous proposons des algorithmes constructifs qui permettent de réduire un ensemble fini de données discrètes définissant une courbe ou une surface affine par morceaux dans un espace euclidien. L'idée essentielle consiste à supprimer les points tels que la courbe ou la surface reconstruite soit assez proche en un certain sens de celle d'origine. Nous utilisons pour cela la distance de Hausdorff entre les courbes ou les surfaces compactes dans \mathcal{R}^n car cette distance convient bien au sens psychovisuel humain. Les applications potentielles se situent en particulier dans le codage de scènes, dans le domaine de l'imagerie médicale ou dans la compression des images numériques.

We present some mathematical constructive algorithms which allow to reduce a discret set of vector valued data defining a linear piecewise curve or surface in some euclidean space. The main idea consists to remove the points such that the reconstructed curve or surface is close in some way to the original one. We shall use the Hausdorff distance between compact curves or surfaces in \mathcal{R}^n since it agrees pretty well with our psychovisual sensitivity. Applications take place particularly in scene coding, in medical imaging or in numerical image compression.

I. Introduction

Depuis une dizaine d'années, plusieurs méthodes de compression de données numériques sont apparues répondant à des besoins économiques de minimisation de l'information à stocker ou à véhiculer. Or il n'existe pas de recette universelle et chaque type de problèmes doit être étudié selon ses spécificités (par exemple en vidéophonie, il convient de prendre en compte la dynamique des séquences d'images à transmettre). La compression est dite conservative dans le cas où la synthèse et le décodage des données réduites s'effectuent sans perte d'information et destructive dans le cas contraire. Les méthodes destructives ont l'avantage d'avoir des taux de compression supérieurs mais nécessitent d'établir des critères subjectifs: on demande que l'information décompressée soit proche de l'originale mais le critère de proximité est relatif en général au sens auditif ou psychovisuel qui varie d'un individu à un autre. Actuellement, nous pouvons dégager deux classes de méthodes de compression: les méthodes spatiales, qui utilisent directement la géométrie des données comme la compression fractale [1] ou l'enlèvement de noeuds par fonctions radiales [2] et les méthodes transformées qui opèrent sur l'information à compresser avant une quantification, comme par exemple les algorithmes pyramidaux de Burt et Adelson [3].

Nous proposons dans cet article une méthode spatiale où la perte d'information, éventuellement nulle, s'effectue selon un critère géométrique de proximité défini arbitrairement par l'utilisateur.

II. Distance de Hausdorff et ε -saucisse de Minkowski

Désignons par $\mathcal{X}(\mathcal{R}^n)$ l'ensemble des compacts de l'espace (\mathcal{R}^n, d) muni de la métrique euclidienne. La distance $d_H(\dots)$ définie sur $\mathcal{X}(\mathcal{R}^n)$ par: $d_H(K_1, K_2) = \max\{f_{K_1}(K_2), f_{K_2}(K_1)\}$ avec $f_{K_1}(K_2) = \max_{k_2 \in K_2} d(k_2, K_1)$, est appelée distance de Hausdorff sur $\mathcal{X}(\mathcal{R}^n)$.

Cette distance peut être exprimée au moyen de l' ε -saucisse de Minkowski $S_K(\varepsilon)$ d'un ensemble $K \in \mathcal{X}(\mathcal{R}^n)$ (vocabulaire dû à B. Mandelbrot parce qu'elle ressemble effectivement à une saucisse dans \mathcal{R}^2 ou \mathcal{R}^3) définie par la réunion des boules fermées $B_\varepsilon(k)$ de rayon ε et de centre k variant dans K :

$$S_K(\varepsilon) = \bigcup_{k \in K} B_\varepsilon(k).$$

En effet, $f_{K_1}(K_2) = \min\{\varepsilon \text{ tel que } K_2 \subset S_{K_1}(\varepsilon)\}$ et on peut alors définir la distance de Hausdorff comme la plus petite des valeurs ε telles que K_2 soit inclus dans l' ε -saucisse de K_1 et K_1 dans l' ε -saucisse de K_2 [4]:



$$d_H(K_1, K_2) = \max \left\{ \min \left\{ \varepsilon \text{ tel que } K_2 \subset S_{K_1}(\varepsilon) \right\}, \min \left\{ \varepsilon \text{ tel que } K_1 \subset S_{K_2}(\varepsilon) \right\} \right\}$$

d'où : $d_H(K_1, K_2) = \min \left\{ \varepsilon \text{ tq } K_2 \subset S_{K_1}(\varepsilon) \text{ et } K_1 \subset S_{K_2}(\varepsilon) \right\}$.

Notons qu'en général $f_{K_1}(K_2) \neq f_{K_2}(K_1)$ pour deux compacts quelconques K_1 et K_2 dans $\mathcal{K}(\mathfrak{R}^n)$. Par définition, si x est un point de \mathfrak{R}^n , nous dirons que l'élément $p_K(x)$ appartenant à $K \in \mathcal{K}(\mathfrak{R}^n)$ est un projeté de x sur K si $d(x, p_K(x)) \leq d(x, k) \quad \forall k \in K$ (un tel élément existe et est unique dans le cas où K est convexe). Alors nous pouvons énoncer la proposition suivante :

Proposition 1:

Si $K_1 = p_{K_1}(K_2) = \bigcup_{k_2 \in K_2} \{p_{K_1}(k_2)\}$ alors $f_{K_2}(K_1) \leq f_{K_1}(K_2)$.

Preuve:

$$\forall k_2 \in K_2, d(k_2, p_{K_1}(k_2)) = \min_{k_1 \in K_1} d(k_2, k_1) \leq \max_{k_2 \in K_2} \min_{k_1 \in K_1} d(k_2, k_1) = f_{K_1}(K_2)$$

Or $K_1 = p_{K_1}(K_2)$ équivaut à $\forall k_1 \in K_1, \exists k_2 \in K_2$ tel que $k_1 = p_{K_1}(k_2)$ et k_2 vérifie donc $d(k_1, k_2) \leq f_{K_1}(K_2)$ ce qui entraîne: $\max_{k_1 \in K_1} \min_{k_2 \in K_2} d(k_2, k_1) = f_{K_2}(K_1) \leq f_{K_1}(K_2)$.

Exemples:

1. Si dans \mathfrak{R}^n , on prend pour K_1 l'origine et pour K_2 la sphère unité $S^{n-1} = \{x/x \in \mathfrak{R}^n, \|x\| = 1\}$ alors $p_{K_1}(K_2) = K_1$ et $p_{K_2}(K_1) = K_2$ donc: $f_{K_1}(K_2) = f_{K_2}(K_1) = d(K_1, K_2) = 1$.
2. Si K_1 est la frontière d'un compact convexe dans \mathfrak{R}^n tel que toute demi-droite issue d'un quelconque de ses points rencontre K_2 , alors $p_{K_1}(K_2) = K_1$.
3. Dans \mathfrak{R}^2 , si $K_1 = [0, 1] \times \{0\}$ et $K_2 = [0, 1] \times \{1\} \cup \{(0, 2)\}$ on a : $K_1 = p_{K_1}(K_2)$ d'où $f_{K_2}(K_1) = 1 \leq f_{K_1}(K_2) = 2$ et $K_2 \not\subset p_{K_2}(K_1) = K_2 \setminus \{(0, 2)\}$.

Proposition 2:

Si $K_1 = \bigcup_{i \in \{1, \dots, n_1\}} K_1^{(i)}$ et $K_2 = \bigcup_{j \in \{1, \dots, n_2\}} K_2^{(j)}$ où $K_1^{(i)}$ et $K_2^{(j)}$ appartiennent à $\mathcal{K}(\mathfrak{R}^n)$ alors:

$$d_H(K_1, K_2) \leq \max \left\{ d_H(K_1^{(1)}, K_2^{(j_1)}), \dots, d_H(K_1^{(n_1)}, K_2^{(j_{n_1})}), d_H(K_2^{(1)}, K_1^{(i_1)}), \dots, d_H(K_2^{(n_2)}, K_1^{(i_{n_2})}) \right\}$$

avec $i_1, \dots, i_{n_2} \in \{1, \dots, n_1\}$ et $j_1, \dots, j_{n_1} \in \{1, \dots, n_2\}$.

Preuve:

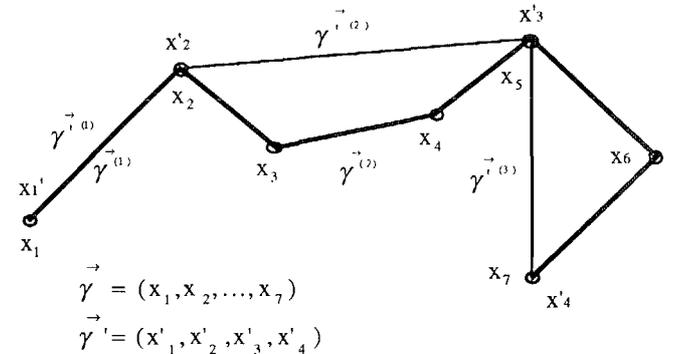
Supposons que le second membre dans l'inégalité soit égal à ε , alors $K_1^{(k)} \subset S_{K_2}^{(k)}(\varepsilon) \subset S_{K_2}(\varepsilon)$ pour $k \in \{1, \dots, n_1\}$ donc $K_1 \subset S_{K_2}(\varepsilon)$. De même $K_2 \subset S_{K_1}(\varepsilon)$ d'où $d_H(K_1, K_2) \leq \varepsilon$.

Nous avons aussi l'égalité évidente si $K_1 \cap K_2$ est différent de K_1 et K_2 : $d_H(K_1, K_2) = d_H(\overline{K_1 \setminus (K_1 \cap K_2)}, \overline{K_2 \setminus (K_1 \cap K_2)})$.

III. Approximation de courbes affines par morceaux

III. 1. Cas des courbes ouvertes

Soit $X = \{x_i\}_{i=1, \dots, N} \subset \mathfrak{R}^n$ un ensemble fini de points ordonnés tels que $x_i \neq x_{i+1}, i = 1, \dots, N-1$ auquel cas nous supprimons le point redondant x_i . Nous associons à X la courbe orientée affine par morceaux de sommets les points de X , notée $\vec{\gamma} = (x_1, x_2, \dots, x_N)$. Désignons par $\vec{\Gamma}$ l'ensemble des lignes brisées (lignes polygonales dans \mathfrak{R}^2) munies de l'orientation définie par la relation d'ordre sur X et de sommets appartenant à un sous-ensemble $X' = \{x'_i\}_{i=1, \dots, N'}$ de X tels que $x'_1 = x_1$ et $x'_{N'} = x_N$.



Si $\vec{\gamma}' \in \vec{\Gamma}$, nous avons $\vec{\gamma} = \bigcup_{i=1}^{N'-1} \vec{\gamma}^{(i)}$ avec $\vec{\gamma}^{(i)} = (x'_i, x'_{i+1})$ et nous pouvons écrire $\vec{\gamma} = \bigcup_{i=1}^{N'-1} \vec{\gamma}^{(i)}$ où $\vec{\gamma}^{(i)}$ est la ligne brisée de sommets x_k compris entre x'_i et x'_{i+1} . Alors d'après la proposition 2:

$$d_H(\vec{\gamma}, \vec{\gamma}') \leq \max_{i=1, \dots, N'-1} \{d_H(\vec{\gamma}^{(i)}, \vec{\gamma}^{(i)}) \text{ tq } \vec{\gamma}^{(i)} \neq \vec{\gamma}'^{(i)}\}$$

Or il est facile de vérifier que $p_{\vec{\gamma}^{(i)}}(\vec{\gamma}'^{(i)}) = \vec{\gamma}^{(i)}$ donc

d'après la proposition 1:

$$d_H(\vec{\gamma}^{(i)}, \vec{\gamma}'^{(i)}) = f_{\vec{\gamma}^{(i)}}(\vec{\gamma}'^{(i)}) = \min \{ \varepsilon \text{ tq } \vec{\gamma}'^{(i)} \subset S_{\vec{\gamma}^{(i)}}(\varepsilon) \} = \max \{ f_{\vec{\gamma}^{(i)}}(\{x_k\}) \text{ tq } x'_i < x_k < x'_{i+1} \} = \max \{ d(\{x_k\}, [x'_i, x'_{i+1}]), x'_i < x_k < x'_{i+1} \}$$

Si nous cherchons une courbe $\vec{\gamma}' \in \vec{\Gamma}$ d'ordre minimal et vérifiant $d_H(\vec{\gamma}, \vec{\gamma}') \leq \varepsilon$ où ε est un nombre positif fixé arbitrairement, nous sommes conduits dans certains cas à des effets désastreux, en particulier la longueur de la courbe $\vec{\gamma}'$ peut être très petite devant celle de $\vec{\gamma}$ même si $\varepsilon > 0$ est infiniment petit. Nous allons donc raisonner localement sur chaque sous-courbe (x_{i-1}, x_i, x_{i+1}) de la courbe $\vec{\gamma}$. L'algorithme procède itérativement en extrayant à l'étape n de la courbe $\vec{\gamma}_{n-1} \in \vec{\Gamma}$

définie auparavant (en posant $\vec{\gamma}_0 = \vec{\gamma}$), une ligne brisée $\vec{\gamma}_n$ appartenant à $\vec{\Gamma}$ de la façon suivante:

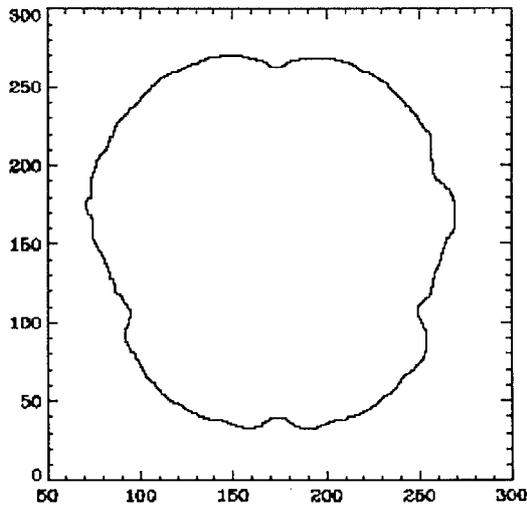
- a) on affecte à chaque sommet intérieur $x_{i,n-1}$ de $\vec{\gamma}_{n-1}$ un poids ω_i calculé par: $\omega_i = d(x_{i,n-1}, [x_{i-1,n-1}, x_{i+1,n-1}])$.
- b) on trie les poids ω_i dans l'ordre croissant.
- c) on enlève un à un les points de poids minimal inférieur à ϵ tel que si $x_{i,n-1}$ est supprimé $x_{i-1,n-1}$ et $x_{i+1,n-1}$ sont conservés, cette dernière contrainte évitant le cumul des erreurs dans l'approximation.

Remarques:

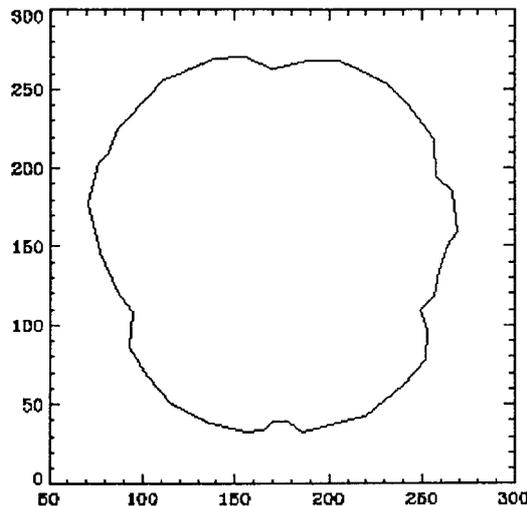
- 1. Si on désigne par $\vec{\gamma}'$ la courbe finale, on a: $d_H(\vec{\gamma}, \vec{\gamma}') \leq \epsilon$.
- 2. Si $2^k \leq N - 2 \leq 2^{k+1} - 1$, l'algorithme converge en $k + 1$ itérations au plus.

III. 2. Cas des courbes fermées

La méthode décrite plus haut se généralise aisément dans le cas où les extrémités de $\vec{\gamma}$ sont confondues. Les figures suivantes permettent d'apprécier la compression d'une courbe polygonale ayant 908 sommets et qui représente le contour d'une coupe horizontale d'un crane humain (cf. figure 1). Après application de l'algorithme avec une tolérance $\epsilon=2$, il ne reste plus que 31 points (cf. figure 2).



Figures 1 et 2



L'exemple suivant est une ligne brisée dans \mathbb{R}^3 ayant 2000 sommets (cf. figure 3). Après application de l'algorithme avec une tolérance $\epsilon=0.3$, il ne reste plus que 233 points (cf. figure 4).

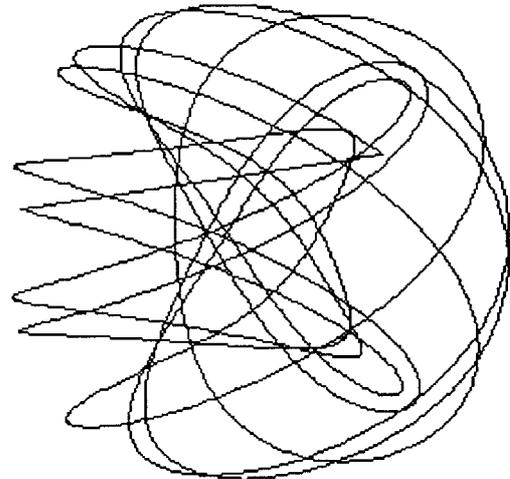


Figure 3

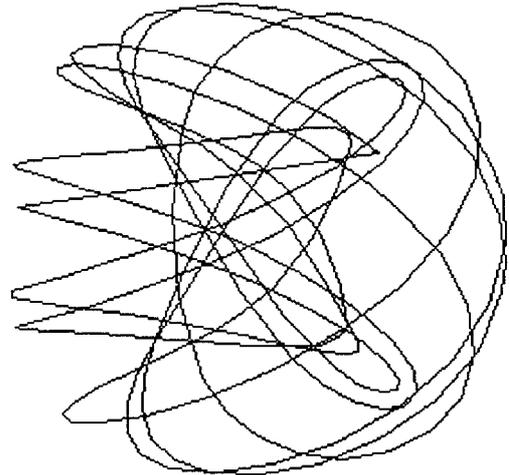


Figure 4

IV. Approximation de surfaces triangulées

Soit $X = \{x_i\}_{i=1,\dots,N}$ un ensemble fini de points distincts définissant les sommets d'une triangulation T dans \mathbb{R}^d . Pour chaque sommet x_i intérieur à T , on désigne par:

S_{x_i} la surface formée par la réunion des triangles dont x_i est un sommet;

S'_{x_i} la surface définie par une triangulation quelconque des noeuds voisins de x_i .

Il est difficile de calculer explicitement $d_H(S_{x_i}, S'_{x_i})$ dans le cas général. Cependant, comme S_{x_i} et S'_{x_i} sont affines par morceaux et ont même bord, cette distance est atteinte en un point appartenant à une arête de S_{x_i} ou de S'_{x_i} . Nous allons chercher une approximation de $f_{S_{x_i}}(S_{x_i})$ en discrétisant les arêtes de S_{x_i} n'appartenant pas à S'_{x_i} . Si $\{y_k\}_{k=1,\dots,M}$ désigne une telle



discrétisation, alors il est aisé de calculer la distance d'un point y_k à la surface triangulée S'_i et on a :

$$f_{S'_i}(S_{x_i}) = \max_{k=1, \dots, M} d(y_k, S'_i).$$

En opérant de même pour $f_{S'_i}(S'_i)$, on obtient une valeur approchée de $d_H(S_{x_i}, S'_i)$. Par analogie avec le cas des courbes, étant donné un réel positif ϵ , nous extrayons à l'étape n de la surface triangulée T_{n-1} définie précédemment ($T_0 = T$) une surface T_n construite telle que :

a) on affecte à chaque sommet intérieur $x_{i,n-1}$ de T_{n-1} un poids ω_i calculé par : $\omega_i = \min d_H(S_{x_i}, S'_i)$ où le min est pris sur l'ensemble des triangulations des voisins de x_i .

b) on trie les poids ω_i dans l'ordre croissant.

c) on enlève un à un les points de poids minimal inférieur à ϵ tel que si $x_{i,n-1}$ est supprimé, ses voisins sont conservés.

Exemples :

Nous avons appliqué cette méthode à la surface triangulée de la figure 5, qui contient 1089 sommets. Après 3 itérations, il reste 398 sommets et 754 triangles (cf. figure 6) et l'erreur relative entre les surfaces initiale et finale est environ de 2 %.

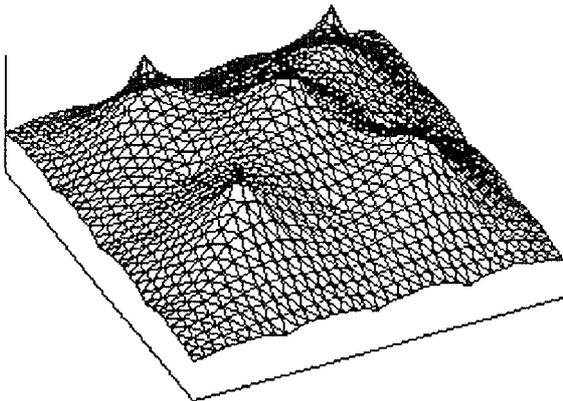


Figure 5

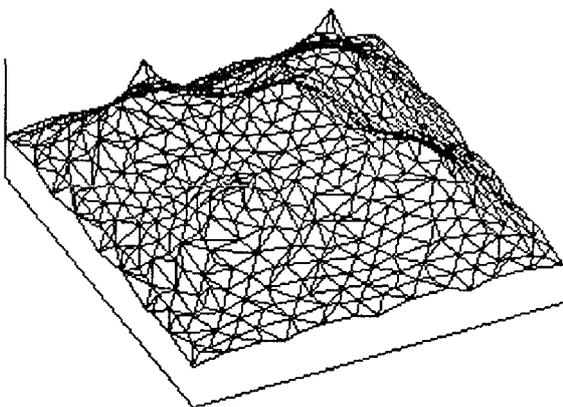


Figure 6

V. CONCLUSION

Les méthodes proposées sont indépendantes du choix du repère de l'espace affine \mathfrak{R}^n . Elles sont fortement parallélisables et les calculs sont très rapides, en particulier pour les courbes. En utilisant la distance de Hausdorff localement sur l'objet géométrique et non pas de manière globale, on *conserve* des informations importantes comme la longueur ou l'aire surfacique. De plus, la distance de Hausdorff entre l'objet initial et l'objet final est majorée par le critère de proximité ϵ choisi par avance, ce qui signifie qu'on ne cumule pas les erreurs entre les différentes étapes de l'algorithme.

REFERENCES

- [1] M.F. BARNESLEY et A.D. SLOANE, " A better way to compress images", Byte, Vol. 13, No 1, 1988, pp 215-223
- [2] M.D. BUHMANN, F. DERRIEN & A. LE MEHAUTE, " Spectral properties and knot removal for interpolation by pure radial sums", à paraître dans Math. Methods for Curves and Surfaces, Vanderbilt University Press, Nashville, 1995
- [3] P.J. BURT and E.H. ADELSON, "The Laplacian pyramid as a compact image code", IEEE Trans. Commun., Col. COM-31, April 1983, pp 532-540
- [4] C. TRICOT, "Courbes et dimensions fractales", Springer-Verlag, 1992