HAL
open science

Synthèse de texture par décomposition parcimonieuse contrainte

Guillaume Tartavel, Yann Gousseau, Gabriel Peyré

To cite this version:

Guillaume Tartavel, Yann Gousseau, Gabriel Peyré. Synthèse de texture par décomposition parcimonieuse contrainte. Gretsi 2013, Sep 2013, Brest, France. hal-00870444

HAL Id: hal-00870444

https://hal.science/hal-00870444v1

Submitted on 7 Oct 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Synthèse de texture par décomposition parcimonieuse contrainte

Guillaume Tartavel[1], Yann Gousseau[1], Gabriel Peyré[2]

[1] LTCI, CNRS / Telecom ParisTech
46 rue Barrault – F-75634 Paris Cedex 13 – France

[2] CNRS / Ceremade, Université Paris-Dauphine
Place du Maréchal De Lattre De Tassigny – 75775 Paris Cedex 16 – France

{Tartavel,Gousseau}@telecom-paristech.fr
Gabriel.Peyre@ceremade.dauphine.fr

Résumé – Cet article aborde la synthèse de texture par une approche variationnelle. À partir d'un échantillon de la texture à synthétiser, nous définissons une énergie non-convexe composée de termes de pénalité associés à des contraintes statistiques. Ces contraintes portent sur l'histogramme et le spectre de l'image, ainsi que sur une décomposition de ses patchs dans un dictionnaire appris au préalable. Le problème de synthèse est alors transformé en un problème de minimisation, pour lequel nous proposons un algorithme. Les minima locaux ainsi obtenus sont de nouvelles synthèses de la texture originale. Des expériences numériques illustrent les résultats de cette approche.

Abstract – This paper tackles the texture synthesis problem from a variational point of view. A non-convex cost function is defined from an exemplar of the texture to be synthesized. Each term of this function is a penalty term derived from a statistical constraint. These constraints take into account the histogram and the spectrum of the image, and also a sparse decomposition of its patches into a dictionary learned from the exemplar image. We propose an algorithm to compute critical points of the cost function, which correspond to synthesis results. Numerical experiments show the usefulness of this approach.

1 Introduction

État de l'art. La synthèse de texture par l'exemple a pour but de générer une image ayant une apparence similaire à une image de départ tout en étant différente. Plusieurs approches ont été proposées pour traiter ce problème.

Les méthodes dites « par l'exemple » effectuent la synthèse d'une nouvelle image de texture en recopiant des morceaux d'une image de référence. C'est le cas de [3, 14] et des nombreuses extensions qui ont suivi. L'état de l'art [13] présente un bon nombre de ces méthodes ainsi que leurs extensions : synthèse de textures dynamiques ou volumiques par exemple.

Une seconde classe de méthodes vise à imposer des contraintes statistiques sur la texture. [5] génère une nouvelle texture en préservant l'amplitude du spectre de Fourier. [6] contraint les distributions des coefficients d'ondelette à chaque échelle, ainsi que la distribution des niveaux de gris de l'image ; [11] contrôle les corrélations entre les coefficients d'ondelette voisins et aux échelles voisines. [9] impose une contrainte de parcimonie sur les patchs de l'image via un dictionnaire appris au préalable, tout en contraignant la distribution des niveaux de gris de l'image.

Dans cette dernière catégorie, les contraintes statistiques sont souvent multiples (e.g. [6], [11], [9]) et sont imposées itérativement les unes après les autres.

Contribution. Nous proposons une formulation variationnelle du problème de synthèse afin de combiner plusieurs contraintes antagonistes, comme la parcimonie des patchs et le spectre de l'image.

Dans notre travail, la fonctionnelle introduite reflète une contrainte de parcimonie inspirée de [9] à laquelle nous ajoutons une contrainte sur la distribution des éléments du dictionnaire, une contrainte de spectre inspirée de [5] et une contrainte sur la distribution des niveaux de gris comme dans [6, 9]. L'objectif est alors de trouver un point critique de cette fonctionnelle.

2 Décomposition parcimonieuse

2.1 Notations

Les matrices sont notées $A = (a^i_j)_{i,j}$; a^i désigne la ligne i et a_j la colonne j. Pour un vecteur w, la pseudo-norme ℓ^0 est définie par $\|w\|_0 = \#\{i \setminus w_i \neq 0\}$. Étant donné un ensemble $\mathcal{C}$, sa fonction indicatrice est notée $\iota_{\mathcal{C}}$:

elle vaut 0 sur $\mathcal{C}$ et $+\infty$ en-dehors.

Dans une image u, nous considérons l'ensemble des patchs

$$p_k = (u(x_k + t))_{t\in\{0,\dots\tau-1\}^2} \tag{1}$$

de taille $\tau \times \tau$ pixels et aux positions $x_k \in \Delta\mathbb{Z}^2$ (i.e. sur une grille de pas $\Delta > 0$). Considérons la matrice $\Pi(u) = (p_k)_k \in \mathbb{R}^{L\times K}$ avec K le nombre de patchs dans l'image et $L = d\tau^2$ la dimension d'un patch, où $d = 1$ dans le cas d'images en niveaux de gris et $d = 3$ en couleurs.

2.2 Apprentissage de dictionnaire parcimonieux

L'ensemble des patchs $\Pi(u)$ est approximé [4] par D_0W_0, où (D_0, W_0) est solution d'un problème de minimisation non-convexe qui fournit un dictionnaire D_0 et les coefficients W_0 associés :

$$\min_{D,W} \|\Pi(u) - DW\|^2 + \iota_{\mathcal{C}_{\text{dico}}}(D) + \iota_{\mathcal{C}_{\text{col}}}(W) \tag{2}$$

avec la contrainte de parcimonie

$$\mathcal{C}_{\text{col}} = \left\{W \in \mathbb{R}^{N\times K} \setminus \|w_k\|_0 \le S \quad \forall k\right\} \tag{3}$$

qui limite à S le nombre de coefficients non nuls de chaque patch p_k, et la normalisation

$$\mathcal{C}_{\text{dico}} = \left\{D \in \mathbb{R}^{L\times N} \setminus \|d_n\| \le 1 \quad \forall n\right\} \tag{4}$$

des N atomes du dictionnaire. L'algorithme proposé dans [4] fournit un minimum local de (2).

3 Formulation variationnelle

Étant donnés une image u_0 de texture à synthétiser ainsi qu'un dictionnaire D_0 et les coefficients W_0 obtenus par (2) à partir de celle-ci, nous définissons une fonctionnelle E qui mesure la qualité d'une synthèse u :

$$E(u, W) = \frac{1}{Z}\|\Pi(u) - D_0W\|^2 + \iota_{\mathcal{C}_{\text{col}}}(W) + \iota_{\mathcal{C}_{\text{ligne}}}(W) + \alpha\mathcal{W}_2^2(\mu_u, \mu_{u_0}) + \beta\,\||\hat{u}| - |\hat{u}_0|\|^2 \tag{5}$$

où les deux paramètres $\alpha, \beta > 0$ contrôlent l'influence relative des différents termes, $Z = \lceil\frac{\tau}{\Delta}\rceil^2$ est une constante de normalisation et $\hat{u}$ désigne la transformée de Fourier discrète à deux dimensions de u.

Cette fonctionnelle est composée de trois termes : un terme de parcimonie des patchs (avec les deux contraintes $\iota_{\mathcal{C}_{\text{col}}}$ et $\iota_{\mathcal{C}_{\text{ligne}}}$ associées), un terme contrôlant la distribution des niveaux de gris et un terme qui régule l'amplitude du spectre de Fourier. Cette formulation permet d'allier de manière naturelle les trois contraintes que sont la parcimonie avec contrôle des fréquences d'utilisation des atomes, la distribution des niveaux de gris et l'amplitude du spectre.

Le terme de parcimonie de notre fonctionnelle reprend (2) et est inspiré de [9], qui montre que le dictionnaire D_0 couplé à la contrainte $\mathcal{C}_{\text{col}}$ permet de capturer l'aspect d'une texture. Dans [9], la parcimonie est imposée indépendamment sur chaque patch. Nous imposons en plus la distribution des atomes du dictionnaire, un peu comme [6] impose la distribution des coefficients d'ondelette à chaque échelle. Pour cela, nous forçons les éléments de D_0 à être utilisés avec la même fréquence dans u_0 et dans u, en introduisant une nouvelle contrainte

$$\mathcal{C}_{\text{ligne}} = \left\{W \in \mathbb{R}^{N\times K} \setminus \|w^n\|_0 \le F_0^n \quad \forall n\right\} \tag{6}$$

qui impose à l'atome d^n d'être utilisé au maximum $F_0^n = \frac{K}{K_0}\|w_0^n\|_0$ fois, où K et K_0 désignent les nombres de patchs dans les images u et u_0 respectivement.

Le terme de distribution des niveaux de gris contrôle l'histogramme de l'image synthétisée, qui est classiquement [6, 9] contraint à être le même que l'histogramme de u_0. Nous utilisons la distance de Wasserstein, notée $\mathcal{W}_2$, afin de mesurer la ressemblance entre les histogrammes. Elle est définie comme

$$\mathcal{W}_2(\mu_u, \mu_{u_0}) = \min_{\sigma} \|u - u_0 \circ \sigma\|_2 \tag{7}$$

où σ parcourt l'ensemble des permutations des pixels de u_0. Pour une image en niveaux de gris ($d = 1$), on montre que la permutation qui minimise cette distance est $\sigma^* = \sigma_{u_0} \circ \sigma_u^{-1}$, où σ_v désigne une permutation telle que les pixels de v soient ordonnés :

$$v_{\sigma_v(1)} \le \dots \le v_{\sigma_v(i)} \le v_{\sigma_v(i+1)} \le \dots \tag{8}$$

Le terme de spectre mesure l'écart entre les spectres de u et de u_0. L'utilisation du spectre en synthèse de texture est détaillée dans [5]. Dans notre cas, il permet de conserver à la fois les très basses fréquences (non capturées par des patchs, trop petits) et les très hautes fréquences (atténuées par la décomposition des patchs).

4 Algorithme de synthèse

La synthèse consiste à trouver des points stationnaires de la fonction de coût (5). Nous procédons par une descente alternée selon les paramètres u (l'image) et W (les poids de la décomposition parcimonieuse) en partant d'un bruit blanc.

4.1 Minimisation par rapport à W

Lorsque l'image u et ses patchs $P = \Pi(u)$ sont fixés, le problème de minimisation par rapport à W s'écrit :

$$\min_{W} \|P - D_0W\|^2 + \iota_{\mathcal{C}_{\text{col}}}(W) + \iota_{\mathcal{C}_{\text{ligne}}}(W). \tag{9}$$

Ce problème d'optimisation combinatoire est NP-difficile. Nous utilisons une heuristique gloutonne inspiré de l'algorithme « Matching Pursuit » (MP) [8] avec la technique de rétro-projection. La contrainte $\mathcal{C}_{\text{ligne}}$, nouvelle par rapport aux précédentes méthodes parcimonieuses, nécessite de décomposer tous les patchs simultanément, car elle

contraint globalement l'utilisation des atomes du dictionnaire.

On part de $W^{(0)} = \mathbf{0} \in \mathbb{R}^{N\times K}$. À une étape ℓ de l'algorithme est choisi un couple patch/atome

$$(n^*, k^*) \in \underset{(n,k)\in\mathcal{I}_{W^{(\ell)}}}{\arg\max} \; |\langle r_k^n, d_n\rangle| \tag{10}$$

où $R = P - D_0 W^{(\ell)}$ et l'ensemble $\mathcal{I}_W$ est :

$$\left\{(n,k) \Big\backslash \sum_{n'\neq n} \|w_k^{n'}\|_0 < S, \; \sum_{k'\neq k} \|w_{k'}^n\|_0 < F_0^n \right\}. \tag{11}$$

Le coefficient associé est alors mis à jour :

$$\left(w^{(\ell+1)}\right)_{k^*}^{n^*} = \left(w^{(\ell)}\right)_{k^*}^{n^*} + \langle r_{k^*}^{n^*}, d_{n^*}\rangle, \tag{12}$$

les autres coefficients restent les mêmes, et l'algorithme passe à l'étape $\ell + 1$. L'algorithme s'arrête lorsque les contraintes $\mathcal{C}_{\text{ligne}}$ ou $\mathcal{C}_{\text{col}}$ sont saturées.

4.2 Descente selon u

Lorsque les coefficients W sont fixés, (5) est non-convexe mais différentiable presque partout. On montre que son gradient est

$$2\mathcal{R}(u, D_0W) + \alpha\nabla_u \mathcal{W}_2^2(\mu_u, \mu_{u_0}) + 2\beta\mathcal{M}(u, u_0) \tag{13}$$

$$\text{avec} \quad \mathcal{R}(u,P) = \frac{1}{Z}\Pi^*\left(\Pi(u) - P\right) \tag{14}$$

$$\text{et} \quad \mathcal{M}(u,u_0) = u - \mathcal{F}^{-1}(|\hat{u}_0|\frac{\hat{u}}{|\hat{u}|}) \tag{15}$$

où $\mathcal{F}^{-1}$ désigne la transformée de Fourier inverse, et

$$\nabla_u \mathcal{W}_2^2(\mu_u, \mu_{u_0}) = 2\left(u - u_0 \circ \sigma_{u_0} \circ \sigma_u^{-1}\right) \tag{16}$$

d'après l'expression (7) et l'expression de σ^* associée. Ce résultat peut être étendu à des images de tailles différentes et à la couleur [12].

Le gradient (13) est $2\,(1+\alpha+\beta)$-Lipschitz : le pas de descente retenu est $\eta = (1+\alpha+\beta)^{-1}/2$.

4.3 Améliorations

Reconstruction non-linéaire : pour éviter le flou introduit par l'application de Π^* dans (14), nous effectuons la fusion de deux patchs adjacents à l'aide d'une coupure similaire à celles décrites dans [7]. Une telle coupure se calcule, dans notre cas, par programmation dynamique.

La reconstruction de l'image entière s'effectue de manière gloutonne, d'abord en fusionnant les patchs horizontalement par coupures verticales pour former des lignes, puis en fusionnant ces lignes par grandes coupures horizontales pour obtenir l'image entière. Ce nouvel opérateur de reconstruction noté Π_{NL}^+ permet de remplacer (14) par $\mathcal{R}_{\text{NL}}(u,P) = u - \Pi_{\text{NL}}^+(P)$.

Multi-échelle : la fonction (5) à minimiser étant non convexe, l'initialisation a une forte influence sur le résultat. Pour éviter de trop mauvais minima locaux, la synthèse est effectuée sur J échelles grâce à une approche multi-échelle classique.

Pour chaque échelle, de la plus grossière $j = J-1$ à la plus fine $j = 0$, l'image u_0 est décimée d'un facteur 2^j pour donner u_j, et un dictionnaire D_j est appris à partir de u_j. La synthèse de u_j est alors effectuée avec l'algorithme décrit, et le résultat est sur-échantillonné et sert d'initialisation à l'échelle $j-1$.

5 Résultats

La figure 1 montre un résultat avec les paramètres suivants : $\tau = 12$ pixels, $\Delta = \tau/2$, $J = 3$ échelles, $S = 4$ coefficients par patch et $\alpha = \beta = 1$. Les images sont issues des bases de Brodatz [2] et VisTex [10].

Notre approche, qui impose l'amplitude des coefficients de Fourier ainsi que les distributions des niveaux de gris et des coefficients du dictionnaire, est comparée à trois autres approches : celle présentée dans [6], basée sur le respect des distributions des coefficients d'ondelette à chaque échelle, l'amélioration proposée dans [11], qui introduit des contraintes de corrélation entre les coefficients voisins, et la méthode de [5] à qui nous empruntons la contrainte de respect des spectres de Fourier.

L'approche de [6] ne permet pas de représenter d'objets structurés ou de bords, car elle est basée sur les coefficients d'ondelette mais n'impose aucune dépendance entre eux. La méthode de [11], qui introduit de telles dépendances, est l'approche statistique qui donne, à notre connaissance, les meilleurs résultats pour la synthèse statistique de texture. Nos résultats sont comparables à celle-ci. L'approximation de la texture par une décomposition parcimonieuse a tendance à rendre la texture plus floue : les détails sont écrasés devant les grosses structures. La contrainte sur le spectre ainsi que la reconstruction non linéaire présentée au paragraphe 4.3 permettent de contrecarrer en partie ce phénomène, visible sur la figure 1 avec la méthode de [9]. Notre méthode a cependant du mal à synthétiser des textures qui présentent à la fois des grandes structures et des détails plus fins : les briques de l'exemple 1 perdent leur grain.

Remarquons que nous ne nous comparons pas ici aux méthodes par « copier-coller » telle l'approche de [3] (voir la démonstration en ligne [1]). Ces approches produisent en général des résultats meilleurs visuellement, mais qui sont un assemblage de morceaux copiés directement de l'image d'origine et pas d'authentiques synthèses.

La figure 2 montre l'influence des paramètres α et β et illustre le rôle de chacun des termes de notre fonctionnelle (5). La contrainte de spectre est visuellement très importante, car elle contrôle les ondulations aux basses fréquences, qui ne sont pas visibles à l'échelle d'un patch.

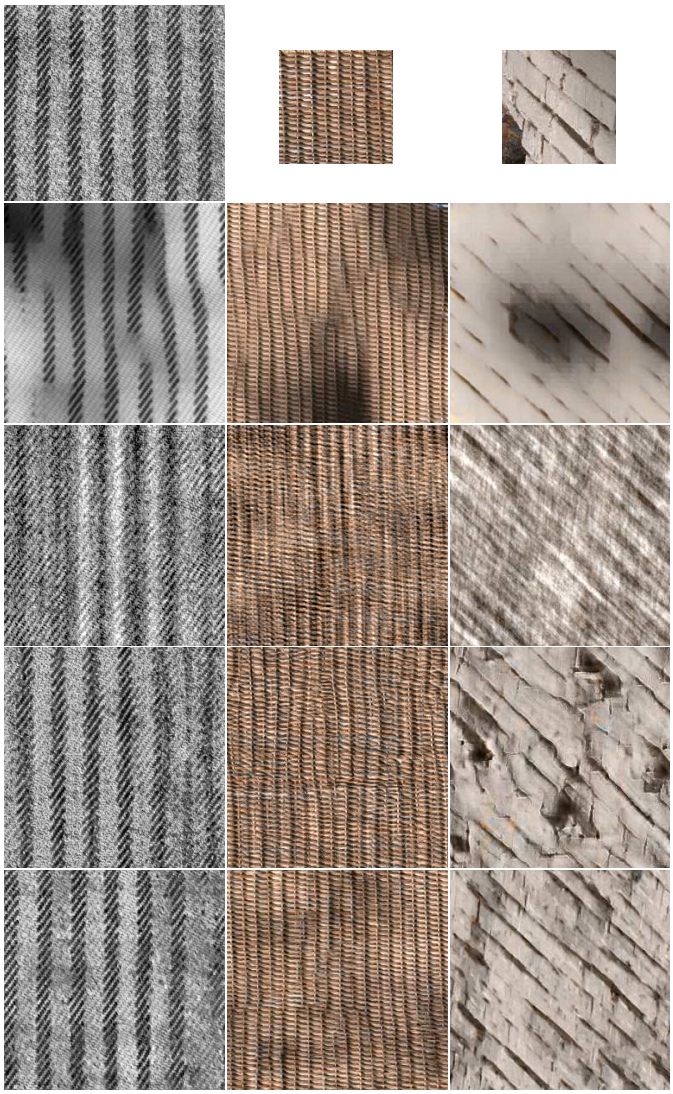

FIGURE 1 – De haut en bas : images d'origine, résultats de [9], de [5], de [11] et résultat de notre algorithme

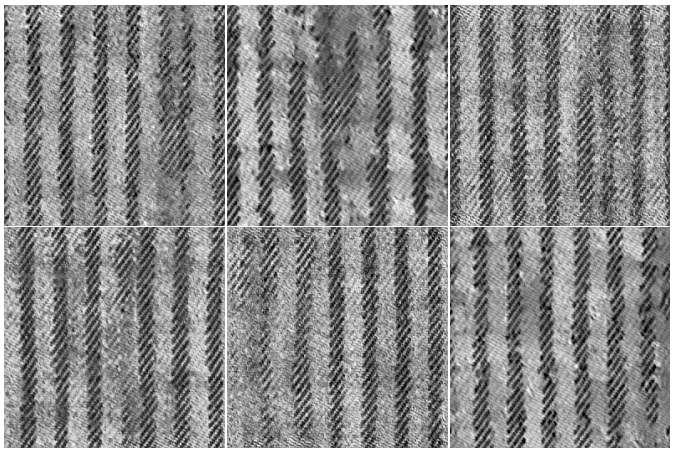

FIGURE 2 – Influence des termes de la fonctionnelle (5). De g. à d. : histogramme, spectre et parcimonie des patchs. En haut avec un poids de 1/9, en bas avec un poids de 9.

Une faible pondération (figure avec $\beta = 1/9$) de ce terme introduit des zones plus claires et d'autres plus foncées, ce qui nuit fortement à la synthèse. La contrainte de parcimonie tend à détruire le grain de la texture : la contrainte de spectre permet de le conserver autant que possible.

Précisons enfin qu'imposer ces trois contraintes tour à tour ne marche pas, d'où la nécessité d'introduire la fonctionnelle (5) afin de les prendre en compte simultanément.

Plus de résultats sont disponibles sur la page web : perso.telecom-paristech.fr/~gousseau/gretsi2013synthesis/

Remerciements. Ce travail a été en parti financé par les projets ANR MATAIM et ERC SIGMA-Vision.

Références

[1] C. Aguerrebere, Y. Gousseau, and G. Tartavel. Exemplar-based texture synthesis : the Efros-Leung algorithm. *Image Processing On Line*, 2013.

[2] P. Brodatz. *Textures : a photographic album for artists and designers*, volume 66. Dover New York, 1966.

[3] A.A. Efros and T.K. Leung. Texture synthesis by non-parametric sampling. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2.

[4] M. Elad and M. Aharon. Image denoising via learned dictionaries and sparse representation. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1.

[5] B. Galerne, Y. Gousseau, and J.M. Morel. Random phase textures : Theory and synthesis. *Image Processing, IEEE Transactions on*, 20.

[6] D.J. Heeger and Bergen J.R. Pyramid-based texture analysis/synthesis. In *SIGGRAPH '95*.

[7] V. Kwatra, A. Schodl, I. Essa, G. Turk, and A. Bobick. Graphcut textures : Image and video synthesis using graph cuts. *ACM Transactions on Graphics*, 22.

[8] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41.

[9] G. Peyré. Sparse modeling of textures. *Journal of Mathematical Imaging and Vision*, 34.

[10] R Pickard, C Graszyk, S Mann, J Wachman, L Pickard, and L Campbell. Vistex database. *Media Lab., MIT, Cambridge, Massachusetts*, 1995.

[11] J. Portilla and E.P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40.

[12] G. Tartavel, Y. Gousseau, and G. Peyré. Constrained sparse texture synthesis. *Proceedings of SSVM'13*, 2013.

[13] L.Y. Wei, S. Lefebvre, V. Kwatra, and G. Turk. State of the art in example-based texture synthesis. In *Eurographics 2009, State of the Art Report, EG-STAR*. Eurographics Association, 2009.

[14] L.Y. Wei and M. Levoy. Fast texture synthesis using tree-structured vector quantization. In *SIGGRAPH '00*.