

E-PLE: un algorithme performant d'inpainting

Yi-Qing WANG

CMLA, ENS Cachan
61 avenue du Président Wilson, 94235 Cachan Cedex, France
yqwang9@gmail.com

Résumé – Le mélange de gaussiennes est un modèle puissant permettant de décrire le comportement *a priori* des patches. Dans cette communication, nous présentons une analyse probabiliste de l'algorithme *piecewise linear estimation* (PLE) appliqué à l'inpainting et, en utilisant efficacement un mélange gaussien, proposons plusieurs améliorations théoriques et numériques.

Abstract – Gaussian mixture is a powerful tool for modeling an image's patch prior. In this communication, a probabilistic view of an existing algorithm *piecewise linear estimation* (PLE) for image inpainting is presented which leads to several theoretical and numerical improvements based on an effective use of Gaussian mixture.

1 Introduction

1.1 Inpainting

Inpainting est une technique d'interpolation qui essaie de ramener une image masquée à son état initial en exploitant au mieux l'information présente dans la partie visible de la même image.

Historiquement, l'un des premiers travaux portant sur le sujet est [10] dans lequel les auteurs, en s'inspirant de la formule de co-aire, proposent de minimiser la courbure de l'image à restaurer à travers une fonctionnelle. Ensuite une variante dans l'espace de fonctions à variation bornée [12] est suggérée. Depuis les efforts se multiplient et plusieurs approches telles que les EDPs [1] et les échantillonnages préférentiels [7] sont mises en avant. Cette dernière fait partie d'un paradigme qui repose sur la similarité et se décline en plusieurs algorithmes de renom dont Non-Local Means [2] et BM3D [4].

À l'initiative des travaux comme [8], une autre voie de recherche s'oriente vers la construction d'un dictionnaire redondant permettant d'avoir une représentation parcimonieuse des patches. Il en résulte des avancées récentes [16, 15] donnant encore un nouveau point de vue via la modélisation gaussienne.

1.2 Estimation linéaire par morceaux

L'idée sur laquelle reposent de nombreux algorithmes de restauration d'image [3, 15] vient de l'observation qu'à l'échelle du patch, la caractéristique la plus saillante d'une image naturelle est l'orientation. Cela conduit à modéliser les patches par une famille de modèles gaussiens décrivant approximativement toute inclinaison d'angle possible, à savoir entre 0 et π . Auxquels s'ajoute un modèle supplémentaire pour rendre compte de la classe de patches contenant essentiellement de la texture.

Pour restaurer une image masquée, il ne reste qu'à associer à tout patch partiellement observé son meilleur modèle dans

la famille gaussienne et puis d'appliquer des outils usuels consistant à trouver un candidat optimal qui minimise un terme d'attache aux données tout en tenant compte d'une certaine contrainte de régularité.

Par conséquent, la restauration s'opère patch par patch et le filtre de Wiener est utilisé par défaut, d'où le nom estimation linéaire par morceau, ou *piecewise linear estimation* (PLE) en anglais pour cette catégorie d'algorithmes. Pour simplifier le langage, dans tout ce qui suit, nous désignons par PLE uniquement l'algorithme tel qu'il est décrit dans [15].

Dans cette communication, nous présentons une analyse probabiliste de PLE appliqué à l'inpainting et, en utilisant efficacement un mélange gaussien, proposons une variante significative E-PLE (où E signifie *Enhanced*) de manière à lui apporter plusieurs améliorations théoriques et numériques.

2 PLE

2.1 Construction et sélection de modèle

Afin de souligner la nouveauté de notre algorithme, nous détaillons comment PLE construit l'ensemble de modèles gaussiens et en choisit un pour un patch donné.

Une gaussienne est complètement caractérisée par son espérance et sa covariance. À l'exception du modèle pour les textures, PLE construit les modèles orientés à partir d'exemples venant de quelques images synthétiques en niveaux de gris dont les gradients à l'intérieur de leurs domaines s'inclinent dans la même direction. Une telle image est créée en convoluant un noyau gaussien avec une image dont les pixels n'ont que deux valeurs distinctes et sont rangés de sorte qu'ils peuvent être séparés selon leur valeur par un segment de droite.

Une décomposition spectrale de la covariance estimée donne une base orthonormale. PLE garde tous ses composants pour le

modèle en question même si une matrice de covariance représentant une inclinaison θ n'est pas de plein rang puisqu'aucune combinaison linéaire de ses vecteurs propres ne peut produire l'inclinaison $\theta + \frac{\pi}{2}$.

Le modèle pour les textures est construit avec la base orthonormale DCT et possède les mêmes valeurs propres que les modèles orientés qui, à la différence de E-PLE, sont fixées arbitrairement au début de même que leur espérance.

Dès lors supposons qu'il y a K modèles gaussiens. Soit un patch partiellement observé \tilde{P} . PLE le filtre K fois au total en maximisant la vraisemblance sous chacun de ces modèles

$$\hat{P}_k = \operatorname{argmin}_P \frac{\|\mathfrak{M}P - \tilde{P}\|^2}{\sigma^2} + (P - \mu_k)^T \Sigma_k^{-1} (P - \mu_k) \quad (1)$$

où \mathfrak{M} , σ , μ_k , Σ_k désignent le masque associé au patch \tilde{P} , le niveau de bruit blanc des pixels visibles, l'espérance et la covariance du modèle k pour l'indice k allant de 1 à K .

La sélection du modèle se fait ensuite par l'évaluation des probabilités conditionnelles

$$k_* = \operatorname{argmax}_{1 \leq k \leq K} p(\hat{P}_{k_*}, \tilde{P} \mid \mu_k, \Sigma_k) \quad (2)$$

avec \hat{P}_{k_*} retenu comme le meilleur estimateur de l'état initial.

Enfin, nous mettons à jour tous les modèles avec les estimateurs qui leurs sont attribués et entamons un nouvel cycle de filtrage et d'estimation si besoin.

2.2 Comment améliorer PLE ?

Malgré ses bons résultats, PLE reste améliorable.

Premièrement, si l'algorithme décrit précédemment est appelé EM par ses auteurs, il ne s'agit pas à proprement parler de *expectation maximization* [5], la classe d'algorithmes reconvenue pour leur capacité de faire croître la vraisemblance d'un mélange sur lequel PLE ne repose pas car son modèle manque la probabilité *a priori* w .

$$p(P) = \sum_{k=1}^N w_k \mathcal{N}(P \mid \mu_k, \Sigma_k) \quad (3)$$

qui ne peut être inférée par l'échantillonnage synthétique.

Deuxièmement, il est préférable que les modèles soient alimentés par de vraies données dès le départ, ce qui permet d'accélérer la phase d'apprentissage de l'algorithme face aux images réelles.

Finalement, comme les covariances des patches orientés n'ont pas besoin d'être de plein rang, restreindre le nombre de leurs vecteurs propres réduit le risque de surapprentissage et aussi le coût de calcul.

3 E-PLE

3.1 Construction du mélange de gaussiennes

Dans l'objectif de construire un mélange gaussien réaliste, nous tirons des échantillons d'une base constituée des images du

Berkeley Segmentation Dataset à l'aide de *tensor structure*, un détecteur d'orientation [9]. À une quantification angulaire près, l'orientation d'un patch orienté peut être définie par

$$\begin{aligned} v_* &= \operatorname{argmin}_{\|v\|=1} \sum_{(r,u) \in \operatorname{dom}(P)} \|\nabla P(r,u) - \langle v, \nabla P(r,u) \rangle v\|^2 \\ &= \operatorname{argmax}_{\|v\|=1} v^T \left(\sum_{(r,u) \in \operatorname{dom}(P)} \nabla P(r,u) (\nabla P(r,u))^T \right) v \quad (4) \end{aligned}$$

où $\nabla P(r,u)$ désigne le gradient discret du patch P au site (r,u) à l'intérieur du domaine de P et $\langle \cdot, \cdot \rangle$ le produit scalaire entre deux vecteurs colonnes de \mathbb{R}^2 . Si la matrice entre les parenthèses dans (4) est appelée M_P , nous déduisons

$$v_*^T M_P v_* = \lambda_b$$

et

$$\operatorname{tr}(M_P) = \sum_{(r,u) \in \operatorname{dom}(P)} \|\nabla P(r,u)\|^2 = \lambda_b + \lambda_s$$

avec $\lambda_b \geq \lambda_s \geq 0$ deux valeurs propres de M_P . D'où le critère

$$\frac{v_*^T M_P v_*}{\sum_{(r,u) \in \operatorname{dom}(P)} \|\nabla P(r,u)\|^2} = \frac{\lambda_b}{\lambda_b + \lambda_s}$$

pour mesurer à quel point le patch favorise l'inclinaison représentée par v_* . Avec deux seuils empiriques s_{plat} et $s_{\text{orienté}}$ fixés, nous décidons que P est

$$\begin{aligned} \text{plat} & \quad \text{si } \lambda_b < s_{\text{plat}} \text{ et } \lambda_s^{-1} \lambda_b < s_{\text{orienté}}; \\ \text{texturé} & \quad \text{si } \lambda_b \geq s_{\text{plat}} \text{ et } \lambda_s^{-1} \lambda_b < s_{\text{orienté}}; \\ \text{orienté} & \quad \text{si } \lambda_s^{-1} \lambda_b \geq s_{\text{orienté}}. \end{aligned}$$

Notre expérience a utilisé 20 modèles dont un texturé, un plat et 18 orientés vers $[\frac{k-1}{18}\pi, \frac{k}{18}\pi[$ pour k allant de 1 à 18 respectivement. Il s'agit en effet d'une partition de l'espace des patches qui facilite l'estimation de la probabilité *a priori* des modèles : il suffit de tirer au hasard un nombre suffisant d'échantillons et d'évaluer la fréquence d'apparition de chacun.

La figure 1 montre bien que pour capter une orientation, une moitié des vecteurs propres suffisent, d'autant plus qu'au vu de la décroissance rapide des valeurs propres, les composantes à hautes fréquences tendent à apparaître moins souvent en réalité. En conséquence, nous proposons le modèle gaussien

$$P = F_k c + \mu_k + \sigma_k n \quad (5)$$

avec F_k une matrice à déterminer contenant 32 (resp. 1) facteurs pour les patches orientés ou texturés (resp. plats) de taille 8×8 et c leur coefficient gaussien centré et réduit.

L'initialisation des modèles repose sur la distance de Kullback-Leibler [11, 13, 14] afin d'approximer au mieux les covariances empiriques des modèles : soit, par exemple, $(\phi_l, \lambda_l)_{1 \leq l \leq 64}$ les vecteurs et valeurs propres satisfaisant $\forall l, \lambda_l \geq \lambda_{l+1}$ de la covariance d'un modèle orienté k . Alors

$$F_k = [(\lambda_1 - \sigma_k^2)^{1/2} \phi_1, \dots, (\lambda_{32} - \sigma_k^2)^{1/2} \phi_{32}]$$

$$\text{avec } \sigma_k^2 = \frac{1}{32} \sum_{l=33}^{64} \lambda_l.$$

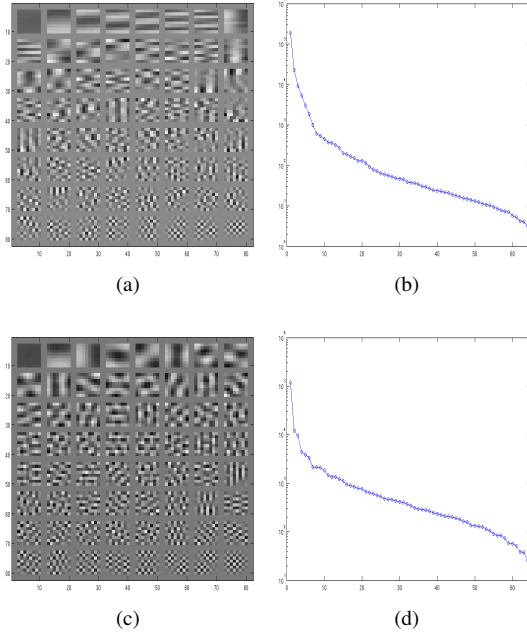


FIG. 1: Les vecteurs et valeurs propres (en échelle logarithmique) obtenus avec des échantillons de taille 8×8 depuis 493 images en niveaux de gris du *Berkeley Segmentation Dataset*. (a)(b) : ceux du modèle orienté verticalement. (c) (d) : ceux du modèle texturé.

3.2 La classification des patches avec EM

Similaire à [13], l’algorithme EM [5] est adapté à notre cadre où les données ne sont que partiellement observées afin d’inférer les paramètres du mélange, y compris ceux pour caractériser les modèles individuels et leur poids probabiliste w_k .

La classification des patches s’ensuit grâce à la règle de décision suivante qui minimise le risque de Bayes [6]

$$\begin{aligned} k_* &= \operatorname{argmax}_k p(\tilde{P} \text{ est généré par le modèle } k \mid \tilde{P}) \\ &= \operatorname{argmax}_k w_k p(\tilde{P} \mid \mu_k, F_k). \end{aligned} \quad (6)$$

La probabilité conditionnelle dans (6) est aisément calculable en raison de l’hypothèse simplificatrice faite sur le coefficient aléatoire c du modèle (5).

La figure 2 illustre un exemple de classification des patches. L’échantillonnage nous donne une initialisation raisonnable du mélange composé de 20 modèles (voir la figure 2(c)) où la classe plate connaît la plus forte probabilité *a priori*, suivie par celle des patches texturés. À l’issue de trois itérations d’EM, les modèles orientés se voient attribuer plus d’importance au détriment du modèle texturé, un comportement auquel nous nous attendons bien.

4 Résultats numériques et discussions

Dans cette étude numérique, nous avons pris quatre images et deux types de masques, à savoir le masque à texte et le masque

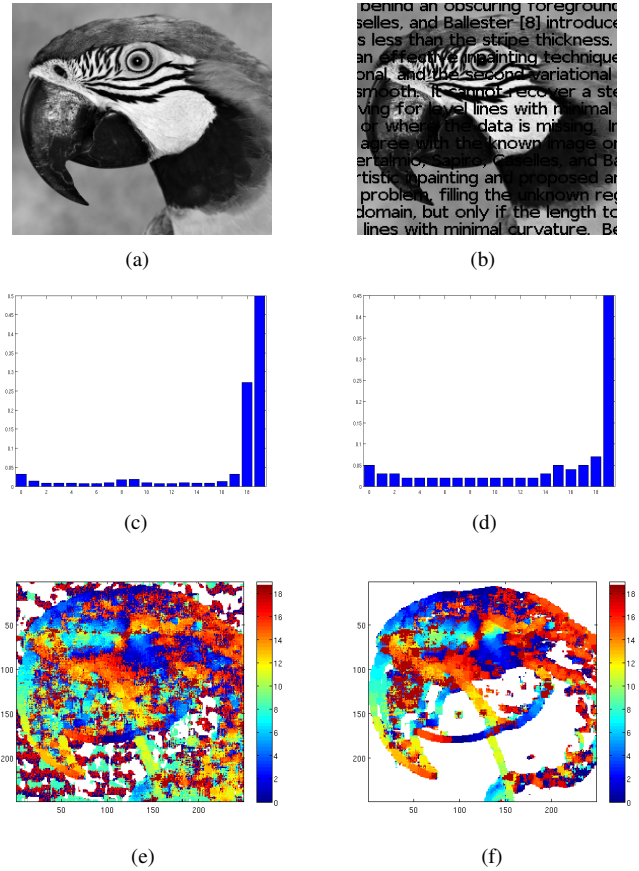


FIG. 2: La classification des patches masqués à l’aide d’EM. (a) image originale (b) image masquée (c) la probabilité *a priori* des modèles initialisés (d) la probabilité à priori inférée par EM au bout de sa troisième itération (e) (resp. (f)) carte d’association patch-modèle au bout de la première (resp. troisième) itération où les couleurs représentent les différents modèles dont la couleur blanche pour le modèle plat.

aléatoire où tout pixel a une certaine probabilité d’être rendu invisible (voir la figure 4(a) à titre d’exemple). Les algorithmes sont itérés six fois dans tous les cas. Le critère d’évaluation est RMSE (*root mean square error*).

Les résultats montrés au tableau 1 confirment que plus les images à traiter sont dégradées, plus l’effet dû à l’incorporation de la probabilité de mélange se fait sentir car c’est à ce moment là que nous avons le plus besoin de faire appel à la connaissance *a priori* pour déterminer les appartenances des patches.

En pratique, avec moins de facteurs à calculer, E-PLE est beaucoup plus rapide que PLE, d’autant plus que grâce à une initialisation plus adaptée, souvent E-PLE n’a besoin que d’une seule itération pour parvenir à un bon résultat en RMSE tandis que PLE typiquement en nécessite beaucoup plus.

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Proceedings of the 27th annual con-*

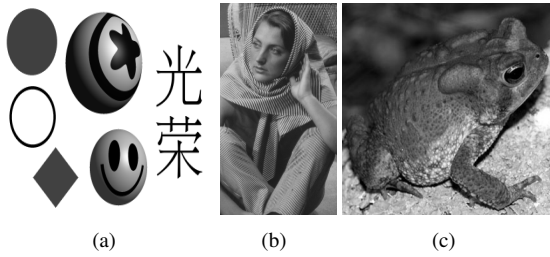


FIG. 3: D'autres images utilisées dans la comparaison des algorithmes. (a) formes (b) barbara (c) grenouille

TAB. 1: Comparison des algorithmes

text	barbara	grenouille	parrot	formes
PLE	4.2	4.3	6.1	5.9
E-PLE	5.0	4.7	6.6	5.7
rand 0.2	barbara	grenouille	parrot	formes
PLE	1.7	2.0	3.7	2.4
E-PLE	1.9	2.2	3.8	2.3
rand 0.4	barbara	grenouille	parrot	formes
PLE	3.7	4.0	6.9	5.4
E-PLE	3.7	4.1	6.7	4.7
rand 0.6	barbara	grenouille	parrot	formes
PLE	10.6	7.1	10.9	11.2
E-PLE	7.9	6.8	10.0	8.8
rand 0.8	barbara	grenouille	parrot	formes
PLE	20.1	11.0	16.0	19.4
E-PLE	16.9	10.8	14.8	16.5

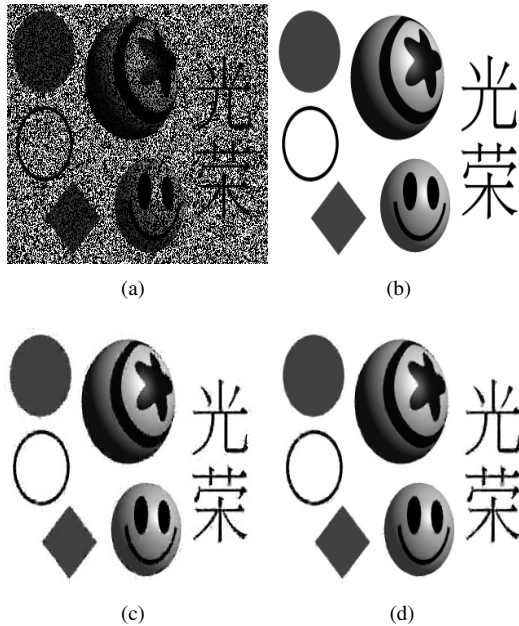


FIG. 4: La comparaison des algorithmes (a) image masquée à 40% visible (b) image originale (c) PLE à six itérations avec RMSE = 11.2 (d) E-PLE à six itérations avec RMSE = 8.8.

ference on Computer graphics and interactive techniques, pages 417–424. ACM Press/Addison-Wesley Publishing Co., 2000.

[2] A. Buades, B. Coll, and J.M. Morel. A review of image

denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.

- [3] P. Chatterjee and P. Milanfar. Clustering-based denoising with locally learned dictionaries. *IEEE Transactions on Image Processing*, 18(7):1438–1451, 2009.
- [4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image restoration by sparse 3D transform-domain collaborative filtering. In *Electronic Imaging*, pages 681207–681207. International Society for Optics and Photonics, 2008.
- [5] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.
- [6] L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic theory of pattern recognition*, volume 31. Springer, 1996.
- [7] A.A. Efros and T.K. Leung. Texture synthesis by non-parametric sampling. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1033–1038. IEEE, 1999.
- [8] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.
- [9] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.
- [10] S. Masnou and J.M. Morel. Level lines based disocclusion. In *International Conference on Image Processing*, pages 259–263. IEEE, 1998.
- [11] S. Roweis. EM algorithms for PCA and SPCA. *Advances in neural information processing systems*, pages 626–632, 1998.
- [12] J. Shen and T.F. Chan. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, 62(3):1019–1043, 2002.
- [13] M.E. Tipping and C.M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482, 1999.
- [14] M.E. Tipping and C.M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.
- [15] G. Yu, G. Sapiro, and S. Mallat. Solving inverse problems with piecewise linear estimators: from Gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21(5):2481–2499, 2012.
- [16] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *IEEE International Conference on Computer Vision*, pages 479–486. IEEE, 2011.