Determinantal Patch Processes for Texture Synthesis

Claire LAUNAY[1], Arthur LECLAIRE[2]

[1]Laboratoire MAP5, CNRS, UMR 8145, Université Paris Descartes,
Sorbonne Paris Cité, Paris, 75006, France

[2]Univ. Bordeaux, IMB, Bordeaux INP, CNRS, UMR 5251
F-33400 TALENCE, France

Claire.Launay@parisdescartes.fr, Arthur.Leclaire@math.u-bordeaux.fr

Résumé – Grâce à leur propriété de répulsion, les processus ponctuels déterminantaux (DPP) constituent un outil efficace pour sous-échantillonner des distributions discrètes. Nous cherchons ici à utiliser les DPP pour sous-échantillonner la distribution des patchs d'une texture. Nous montrons que dans un modèle de texture basé sur du transport optimal, les DPP permettent d'atteindre un bon compromis entre qualité visuelle de la synthèse et temps d'exécution.

Abstract – Because of their repulsive property, determinantal point processes (DPPs) provide an efficient tool to subsample discrete distributions. In this paper, we investigate the use of DPPs to subsample the distribution of patches of a texture image. We show that in a texture model based on optimal transportation, DPP subsampling helps to reach a good compromise between visual quality of the synthesis and execution time.

1 Introduction

Exemplar-based texture synthesis consists in producing texture samples that look similar to a given original texture called the exemplar. Many methods have been proposed to solve this problem, broadly categorized between parametric methods [13, 6] and patch-based methods [3, 4, 11]. Here we build on the texture model proposed in [5], which exploits optimal transport (OT) in the patch space in order to reimpose statistics of local features at several resolutions.

More precisely, this model is based on semi-discrete OT, meaning that it uses transformations of the patch space that are designed to optimally transport an absolutely continuous source measure onto a discrete target measure. The chosen discrete target measure in [5] is the subsampled empirical patch distribution of the exemplar texture, so that these OT maps help to reimpose the patch statistics of the exemplar. These OT maps are given by weighted nearest neighbor (NN) assignment on the points of the target measure support. Therefore, the computational time for synthesis highly depends on the discrete sampling of the target distribution. For 3×3 patch distributions, a naive 1000-uniform subsampling gives good results in general. But more accurate subsampling strategies could be used by taking profit of the structure in the patch point cloud.

In this paper, we propose to use a different subsampling strategy based on determinantal point processes (DPPs) defined on patches. A DPP tends to produce diverse sets of points and have appealing theoretical properties [10]. They are defined through a kernel matrix related to the similarity between the points of the original set. The more similar two points are, the less likely they are to be selected together. For that reason, DPPs constitute a common tool to efficiently subsample large data sets. For instance, the authors of [10] use DPPs to summarize text documents. The authors of [14] also study the properties of weighted sub-samples, called coresets, in order to approximate an initial data set with respect to a given learning task, and prove that DPPs produce better coresets than iid methods. The authors of [12] show that DPPs are a convenient object to estimate various statistics on a large population.

Here we propose to integrate the DPP subsampling strategy in the OT-based texture model of [5]. We show that because of the repulsion property of the DPP, it is able to cover efficiently the original patch cloud with a low number of samples. As a result, the obtained transport maps can be applied faster, thus allowing to synthesize very large structure textures with competitive computational time. We also discuss the parameters of the model, in particular the expected cardinal of the DPP, which should depend on the complexity of the input texture.

2 Texture Synthesis with Semi-Discrete Optimal Transport

In this section, we will recall the definition given in [5] of the texture model based on semi-discrete optimal transport. Let $u : \Omega \to \mathbb{R}^d$ be the exemplar texture defined on a domain $\Omega \subset \mathbb{Z}^2$. The patch domain will be denoted by $\omega = \{0, \dots, w-1\}^2$ and the patch space by $\mathbb{R}^D$ where $D = dw^2$.

2.1 Monoscale Model

The model is based on a coarse synthesis obtained with a Gaussian field U defined by

$$\forall\, x \in \mathbb{Z}^2,\ U(x) = \bar{u} + \sum_{y \in \mathbb{Z}^2} t_u(y) W(x-y) \tag{1}$$

where $\bar{u} = \frac{1}{|\Omega|}\sum u(x)$, $t_u = \frac{1}{\sqrt{|\Omega|}}(u-\bar{u})\mathbf{1}_\Omega$ and W is a normalized Gaussian white noise on $\mathbb{Z}^2$. This Gaussian random field is adapted only to the synthesis of unstructured textures. For that reason, the authors of [5] proposed to apply local modifications to reinforce geometric structures in a statistically coherent way. In other words, a transformation $T : \mathbb{R}^D \to \mathbb{R}^D$ is applied to all patches of U, an image is recomposed by simple averaging, thus obtaining the transformed random field

$$\forall x \in \mathbb{Z}^2, \quad V(x) = \frac{1}{|\omega|} \sum_{h \in \omega} T(U_{|x-h+\omega})(h). \tag{2}$$

The map T is chosen to solve a semi-discrete optimal transport problem between the probability distribution μ of U_ω and a discrete target distribution $\nu = \sum_{j=1}^{J} \nu_j \delta_{q_j}$ representing the patches of u (see Section 3.1). This problem can be written as

$$\inf \int_{\mathbb{R}^D} \|p - T(p)\|^2 d\mu(p) \tag{3}$$

where the infimum is taken over all measurable maps T for which the image measure of μ is ν. As proved in [2, 9], the solution can be obtained as a weighted NN assignment

$$T_v(p) = q_{j(p)} \quad \text{where} \quad j(p) = \operatorname*{Argmin}_j \|p - q_j\|^2 - v_j \tag{4}$$

where $v \in \mathbb{R}^J$ solves a concave maximization problem. Solving for v relies on a costly stochastic gradient procedure (see the details in [7, 5]) which is more and more difficult when the number J of points in the target distribution increases. This is a first reason to look for a simplification of the target measure ν with the least possible points. Another reason, which will be highlighted in the experimental section, is that once the map T_v is estimated, applying it to all patches of U amounts to applying a weighted NN projection on a set of J patches; thus the required computational time for synthesis also depends on the number J of points in the target distribution.

2.2 Multiscale Model

One drawback of the stochastic algorithm for semi-discrete OT is that it gets slower when the dimension D increases. In practice, it is thus only applicable for patches of size 3×3. A multiscale extension was proposed in [5] in order to deal with larger structures. It consists in working with subsampled versions $u^\ell, \ell = 0, \dots, L-1$ of the original texture defined on coarser grids $\Omega^\ell = \Omega \cap 2^\ell \mathbb{Z}^2$, and with discrete target patch distributions $\nu^\ell, \ell = 0, \dots L-1$.

Starting from a Gaussian random field U^{L-1} estimated from u^{L-1} as in (1), for $\ell = L-1, \dots, 0$, we apply a transport map T^ℓ to all patches of U^ℓ

$$V^\ell(x) = \frac{1}{|\omega|} \sum_{h \in 2^\ell \omega} T^\ell(U^\ell_{|x-h+2^\ell\omega})(h), \quad x \in 2^\ell \mathbb{Z}^2 \tag{5}$$

and we get $U^{\ell-1}$ by exemplar-based upsampling (taking the same patches than T^ℓ but twice larger). The transport map T^ℓ is designed to solve a semi-discrete OT problem between a source measure μ^ℓ (a GMM estimated from the patches of the current synthesis) and a discrete target distribution ν^ℓ representing the patches of u^ℓ. The output texture is V^0.

One strong feature of this multiscale model is that the maps T^ℓ can be estimated once and for all. Once the model estimated, it can be sampled efficiently since applying the map T^ℓ at each scale consists in a simple weighted NN projection on 3×3 patches.

3 DPP Subsampling of the Target Distribution

In this section, we discuss how to choose the discrete target distribution ν in order to represent efficiently the patches of the original texture u.

3.1 Choosing the Target Distribution

One natural choice to represent all the patches of u is of course to consider the empirical distribution

$$\nu_{\text{emp}} = \frac{1}{I} \sum_{i=1}^{I} \delta_{p_i} \tag{6}$$

where $\{p_i, 1 \leqslant i \leqslant I\}$ is the set of all patches of u. Unfortunately, this choice must often be discarded because the number I of patches is in general very large ($I \gg 10^5$) and thus unsuitable for the stochastic algorithm for semi-discrete OT.

The authors of [5] coped with this problem by considering the simple subsampling

$$\nu_{\text{unif}} = \frac{1}{J} \sum_{j=1}^{J} \delta_{q_j} \tag{7}$$

where the patches (q_j) are chosen at random (uniformly) among the patches (p_i). Although naive, this solution proved to be sufficient for many textures, with a value of J set as a ground rule to $J = 1000$ for subsampling 3×3 patch distributions. But we propose here to consider alternative choices in order to use even lower values of J while maintaining the visual quality of the output texture.

3.2 DPP Definition and Properties

Determinantal point processes are increasingly used to subsample sets of items containing redundancy, as they capture diversity and assign a low probability to sets of similar items.

A determinantal point process relies on a similarity measure defined on the points of the initial set and stored in a positive semi-definite kernel matrix $K = (K_{i,j})_{1 \leqslant i,j \leqslant N}$.

Definition 3.1 ([10]). A point process Y with values in $\mathcal{Y} = \{1, \ldots, N\}$ is said to be determinantal with kernel K if for all subset $A \subset \mathcal{Y}$,

$$\mathbb{P}(A \subset Y) = \det(K_A), \quad \text{with } K_A = (K_{i,j})_{i,j \in A}. \tag{8}$$

More generally, one can define a DPP with values in a set $\mathcal{Y}$ with cardinal N by using a bijection between $\mathcal{Y}$ and $\{1, \ldots, N\}$.

The existence of such a DPP imposes that the eigenvalues $\{\lambda_1, \ldots \lambda_N\}$ of K must be in $[0, 1]$. One can notice that the diagonal coefficients of K define the marginal probabilities of any singleton, as $\mathbb{P}(i \in Y) = K_{ii}$ for any $i \in \mathcal{Y}$. As the off-diagonal coefficients of K give the similarity between points, the more similar two points are, the less likely they are to be selected together

$$\mathbb{P}(\{i, j\} \subset Y) = \mathbb{P}(i \in Y)\mathbb{P}(j \in Y) - |K_{i,j}|^2. \tag{9}$$

Besides, it is possible to control the cardinal $|Y|$ of the DPP since $\mathbb{E}(|Y|) = \sum \lambda_i = \operatorname{tr}(K)$ and $\operatorname{Var}(|Y|) = \sum \lambda_i(1 - \lambda_i)$. In practice, for a given subsampling problem, the main issue is to define an appropriate matrix K that reflects the structure of the original set $\mathcal{Y}$.

3.3 Determinantal Patch Processes

We want to subsample the set of patches of u with a DPP. A common method to control the repulsion and define an appropriate kernel K is to set $K = L(\mathrm{Id} + L)^{-1}$, where Id is the identity matrix and L a positive semi-definite kernel. Here, we use a Gaussian matrix with parameter s

$$\forall i, j \in \{1, \ldots, N\}, \quad L_{ij} = \exp\left(-\frac{\|p_i - p_j\|_2^2}{s^2}\right) \tag{10}$$

We choose to apply the squared Euclidean distance between patches. It is widely used as a similarity measure on patches, for instance in the patch-based denoising method NL-means. Indeed, it is fast to compute and, despite its natural limitations, it provides satisfying results.

Regarding the choice of the bandwidth parameter s, one can remark that if it is set small, K will become close to the identity matrix and the selection of patches will be close to a random uniform sampling. The larger s is, the more repulsive the DPP is. Yet, because of numerical instability, s should not be too large either. As noticed by [1] or [14], the median of the interdistances between the patches seems to be a satisfying and common choice for s, even if no theoretical guarantee exists to support it. In practice, we also add a multiplicative coefficient so that s evolves in function of the number of samples we want.

The usual algorithm to sample DPPs [8] uses the eigendecomposition of K and costs $\mathcal{O}(N^3)$, which is very costly. Yet, we only need to perform this sampling once and as it enables to significantly reduce the number of patches used to estimate the target distribution, we will see in the next section that this cost can be afforded.

3.4 Setting the Weights

Once the support $\mathcal{Q} = \{q_j, 1 \leqslant j \leqslant J\}$ has been fixed, one must build a measure ν supported on $\mathcal{Q}$ that accurately represents the patches of u. This amounts to adjusting the masses (ν_j) associated with (q_j) such that $\nu = \sum_{j=1}^J \nu_j \delta_{q_j}$ realizes a good approximation of ν_{emp}. One can formulate this problem using the L^2-Wasserstein distance between discrete distributions $\mu = \sum_{i=1}^I \mu_i \delta_{p_i}$ and $\nu = \sum_{j=1}^J \nu_j \delta_{q_j}$ defined by

$$W_2^2(\mu, \nu) = \inf_{(\pi_{i,j})} \sum_{i,j} \pi_{i,j} \|p_i - q_j\|^2 \tag{11}$$

where the infimum is taken on $(\pi_{i,j}) \in \mathbb{R}_+^{I \times J}$ such that for all i, $\sum_j \pi_{i,j} = \mu_i$ and for all j, $\sum_i \pi_{i,j} = \nu_j$. Finding the masses (ν_j) that minimizes $W_2^2(\nu_{\text{emp}}, \nu)$ is equivalent to solving

$$\pi^*_{i,j} = \operatorname*{Argmin}_{(\pi_{i,j})} \sum_{i,j} \pi_{i,j} \|p_i - q_j\|^2 \tag{12}$$

such that $\forall (i, j), \pi_{i,j} \geq 0$ and $\sum_j \pi_{i,j} = \frac{1}{I}$, which is similar to the original OT problem, but relaxing the second marginal constraint. The solution ν can thus be obtained with

$$\forall j \in \{1, \ldots, J\}, \; \nu^*_j = \sum_i \pi^*_{i,j}. \tag{13}$$

This is simply a linear programming problem with the projection on a simplex that can be solved with the "Interior point" or "Dual simplex" algorithms. Finally we approximate the empirical distribution with the (random) distribution

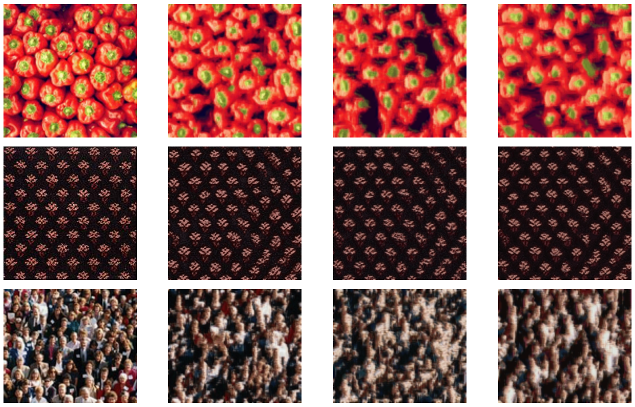
$$\nu_{\text{DPP}} = \sum_{j=1}^J \nu^*_j \delta_{q_j}, \tag{14}$$

where (q_j) is a realization of the DPP with kernel K.

4 Results

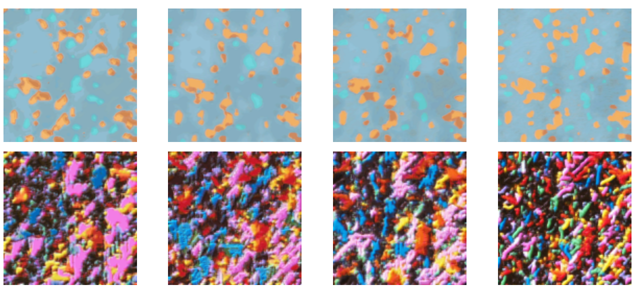
We will now comment the synthesis results obtained by subsampling the target patch measures with DPPs. All parameters of the texture model are set to the default values listed in [5] (4 scales, patches of size 3×3). The only difference lies in the subsampling strategy. At each scale, a first naive subsampling is performed by drawing (uniformly) 1000 patches in the exemplar texture. Then, a second subsampling step is performed with either another uniform subsampling to cardinal J or a DPP subsampling with expected cardinal J. Let us mention that we cannot use a direct DPP subsampling of ν_{emp} because I is often very large ($\approx 10^6$) and it is thus impractical to compute the eigenvalues of the kernel (which is needed for the DPP sampling). In the following experiments, $J \in \{50, 100, 200\}$.

In Fig. 1, one can observe a predictable loss of quality when going from 1000 to 100 patches. However, one can see that for many textures, the visual quality can be maintained to a reasonable level while using 10 times less patches. This will help us to reach a compromise between visual quality and execution time for synthesis (see below). One can also observe on Fig. 1



Original Unif-1000 Unif-100 DPP-100

FIGURE 1 – We compare the synthesis results when using either a target distribution with uniform subsampling (with cardinal 100) and DPP subsampling (with expected cardinal 100).



DPP-50 DPP-100 DPP-200 Unif-1000

FIGURE 2 – We display the impact on the visual results of the expected cardinal of the DPP. See the text for comments.

that uniform and DPP subsampling behave quite differently. In particular, DPP subsampling seems to favor patches with sharper edges and less noise. Also, on several textures (like the last example of Fig. 2), the output seems statistically closer to the input texture ; but it would require a more involved analysis to precisely assess this fact.

In Fig. 2, we analyze the influence of the cardinal of the target discrete distribution. For each texture there is a cardinal value under which results get degenerate and over which the visual quality is maintained to a reasonable level. The results shown in Fig. 3 also help to numerically evaluate the gain to use DPP sampling instead of uniform sampling.

Finally, let us highlight the main benefit obtained with the proposed subsampling strategies, which lies in the gain in computation time for synthesis. Once the texture model is estimated, it is indeed very fast to sample large pieces of it, and since it relies on weighted NN assignments at each scale, the execution time depends quasi-linearly on the cardinal J of the target measures. Using a CPU Intel i7-5600U (4 cores at 2.6GHz), for $J \leqslant 200$ we are able to synthesize 512×512 images in ≈ 0.4" and 1024×1024 in $\approx$ 1.6". This execution time would be improved with a GPU implementation. For some textures, the suggested approach thus allows to accelerate the synthesis algorithm of [5] while maintaining the quality of synthesis.

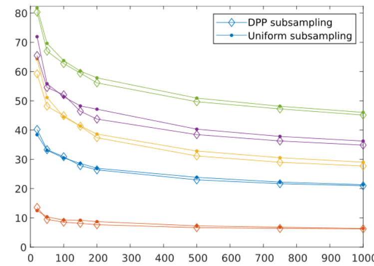


FIGURE 3 – These plots shows, for five textures the mean L^2 distance from a patch to its NN in the DPP, depending on the expected cardinal.

Acknowledgments The authors would like to thank the reviewers for their helpful comments, and acknowledge support of Institut Henri Poincaré (UMS 839 CNRS-Sorbonne Université), and LabEx CARMIN (ANR-10-LABX-59-01).

Références

[1] C. C. Aggarwal. *Outlier Analysis*. Springer Publishing Company, Incorporated, 2nd edition, 2016.

[2] F. Aurenhammer, F. Hoffmann, and B. Aronov. Minkowski-type theorems and least-squares clustering. *Algorithmica*, 20(1) :61–76, 1998.

[3] A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *Proc. of ICCV*, page 1033, 1999.

[4] A.A. Efros and W.T. Freeman. Image quilting for texture synthesis and transfer. *ACM TOG*, pages 341–346, August 2001.

[5] B. Galerne, A. Leclaire, and J. Rabin. A texture synthesis model based on semi-discrete optimal transport in patch space. *SIAM Journal on Imaging Sciences*, 11(4) :2456–2493, 2018.

[6] L. Gatys, A. S Ecker, and M. Bethge. Texture synthesis using convolutional neural networks. In *Proc. of NIPS*, pages 262–270, 2015.

[7] A. Genevay, M. Cuturi, G. Peyré, and F. Bach. Stochastic optimization for large-scale optimal transport. In *Proc. of NIPS*, pages 3432–3440, 2016.

[8] J. B. Hough, M. Krishnapur, Y. Peres, and B. Virág. Determinantal processes and independence. *Probability Surveys*, pages 206–229, 2006.

[9] J. Kitagawa, Q. Mérigot, and B. Thibert. A Newton algorithm for semi-discrete optimal transport. *Journ. of the Europ. Math Soc.*, 2017.

[10] A. Kulesza and B. Taskar. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(2-3) :123–286, 2012.

[11] S. Lefebvre and H. Hoppe. Parallel controllable texture synthesis. *ACM TOG*, 24(3) :777–786, 2005.

[12] V. Loonis and X. Mary. Determinantal sampling designs. *Journal of Statistical Planning and Inference*, 199 :60 – 88, 2019.

[13] J. Portilla and E.P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *Int. J. Comput. Vis.*, 40(1) :49–70, 2000.

[14] N. Tremblay, S. Barthelmé, and P.O. Amblard. Determinantal Point Processes for Coresets. preprint, 2018.