Optimal transport between GMM for multiscale texture synthesis

Julie Delon, Agnès Desolneux, Laurent Facq, Arthur Leclaire

Optimal transport between GMM for multiscale texture synthesis

Julie Delon[1], Agnès Desolneux[2], Laurent Facq[3], and Arthur Leclaire[3,*]

[1] Université Paris Cité, CNRS, MAP5 UMR 8145, F-75006 Paris, France
julie.delon@u-paris.fr
[2] Centre Borelli, CNRS and ENS Paris-Saclay, F-91190 Gif-sur-Yvette, France
agnes.desolneux@ens-paris-saclay.fr
[3] Univ. Bordeaux, Bordeaux INP, CNRS, IMB, UMR 5251,F-33400 Talence, France
{laurent.facq,arthur.leclaire*}@math.u-bordeaux.fr Corresponding author*

Abstract. Using optimal transport in image processing tasks has become very popular. However, it still faces difficult computational issues when dealing with high-dimensional distributions. We propose here to use the recently introduced GMM-OT formulation, which consists in restricting the optimal transport problem to the set of Gaussian mixture models. As a proof of concept, we use it to improve the texture model Texto based on optimal transport between distributions of image patches. Using GMM-OT in this texture model allows to deal with larger patches, hence providing results with better geometric details. This new model allows for synthesis, mixing, and style transfer.

Keywords: Optimal Transport · Texture Synthesis · Gaussian Mixtures

Keywords: Optimal transport, Gaussian mixture models, texture synthesis

1 Introduction

Numerical optimal transport (OT) has undergone spectacular progress in the last ten years, and is now used in a large variety of applications [1, 4, 11, 7]. In particular, important advances have been made in the numerical approximations of optimal transport, with the emergence of efficient tools like regularized optimal transport [5] or the sliced optimal transport [2]. Nevertheless, it remains complex to compute optimal transport distances between empirical distributions when the dimension (the number of samples n or the space dimension d) of the problem increases too much.

In this context, several questions were raised concerning the ability to compute numerical solutions of high-dimensional problems or the sample complexity of the different transport approximations [15, 10, 3]. However, for several applications, targeting exact solutions of optimal transport might not be desirable, whereas proxy formulations sharing similar properties might deliver more relevant solutions in practice. Among these alternative formulations, an OT-like

distance between Gaussian Mixture Models (GMM) has been introduced in [6]. It consists in restricting the set of possible couplings to GMM in the product space. Solutions of this formulation are easy to compute and merely require to calculate Bures distances between Gaussian measures and solve a small-scale discrete OT problem. When applied to discrete data, the dependency on the dimension d and the number of samples n lies only in the GMM fitting step on the data, and in the computation of Bures distances. This makes the approach very versatile and robust to dimension in practice. In this paper we explore the use of GMM-OT for texture modeling, as a proof of concept. To this aim, we improve the texture model Texto [8], which is based on semi-discrete OT on image patches. We end up with a lighter and simpler formulation of the same problem. More precisely, in this context, the computing time of GMM-OT is at least one order of magnitude faster than the ones of semi-discrete OT or regularized OT, for similar (or even better) quality of synthesized images. This permits to use larger patch dimensions, and much more patches than the original Texto model.

2 Reminders on optimal transport between Gaussian mixture models

This section recalls the main results on OT between GMM [6]. The quadratic Wasserstein distance between two probability measures μ_0, μ_1 on $\mathbb{R}^d$ with finite second moments is defined as

$$W_2^2(\mu_0, \mu_1) := \inf_{\gamma \in \Pi(\mu_0, \mu_1)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|y_0 - y_1\|^2 d\gamma(y_0, y_1), \tag{1}$$

where $\Pi(\mu_0, \mu_1)$ is the set of probability measures on $\mathbb{R}^d \times \mathbb{R}^d$ with marginals $\mu_0,\ \mu_1$. A solution γ^* of (1) is called an OT plan between μ_0 and μ_1. This distance has been extensively used for various applications in data science, and especially to define Wasserstein barycenters of probability measures, which are defined similarly to Euclidean barycenters, replacing the Euclidean distance by W_2.

2.1 Definition of MW_2

Let us denote by GMM_d the set of probability distributions which can be written as finite GMM on $\mathbb{R}^d$. OT plans and Wasserstein barycenters between GMM are usually not GMM themselves, which can be troublesome if we rely on this modeling to analyse or generate data. For this reason, the authors of [6] propose to modify the formulation of the classical Wasserstein distance by restricting the set of possible coupling measures to GMM on $\mathbb{R}^d \times \mathbb{R}^d$. More precisely, for $\mu_0, \mu_1 \in GMM_d$, we define

$$MW_2^2(\mu_0, \mu_1) := \inf_{\gamma \in \Pi^{\mathrm{GMM}}(\mu_0, \mu_1)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|y_0 - y_1\|^2 d\gamma(y_0, y_1), \tag{2}$$

where $\Pi^{\mathrm{GMM}}(\mu_0, \mu_1)$ is the set of probability measures in GMM_{2d} with marginals μ_0 and μ_1. It is shown in [6] that MW_2 defines a distance on GMM_d. Moreover,

if $\mu_0 = \sum_{k=1}^{K_0} \pi_0^k \mu_0^k$ and $\mu_1 = \sum_{l=1}^{K_1} \pi_1^l \mu_1^l$, where the π_0^k, π_1^l are non-negative scalars and where the μ_0^k, μ_1^l are Gaussian measures, it can be shown [6] that

$$MW_2^2(\mu_0, \mu_1) = \min_{w \in \Pi(\pi_0, \pi_1)} \sum_{k,l} w_{kl} W_2^2(\mu_0^k, \mu_1^l), \tag{3}$$

where $\Pi(\pi_0, \pi_1)$ is the of $K_0 \times K_1$ matrices with non-negative entries and discrete marginals π_0 and π_1. This discrete expression makes MW_2 easy to compute in practice, even in large dimension. Indeed, the distance W_2 between two Gaussian measures $\mu = \mathcal{N}(m, \Sigma)$ and $\tilde{\mu} = \mathcal{N}(\tilde{m}, \tilde{\Sigma})$ has a closed-form expression:

$$W_2^2(\mu, \tilde{\mu}) = \|m - \tilde{m}\|^2 + \operatorname{tr}\left(\Sigma + \tilde{\Sigma} - 2\left(\Sigma^{\frac{1}{2}} \tilde{\Sigma} \Sigma^{\frac{1}{2}}\right)^{\frac{1}{2}}\right), \tag{4}$$

where we denote by $M^{\frac{1}{2}}$ the unique semi-definite positive square-root of a symmetric semi-definite positive matrix M. If the different parameters of the GMM μ_0 and μ_1 are known, computing (3) boils down to computing $K_0 \times K_1$ Wasserstein distances between Gaussian measures and then to solve a $K_0 \times K_1$ discrete OT problem. Similarly to the Wasserstein distance, it is possible to define barycenters for MW_2, and this also gives rise to a simple discrete formulation.

2.2 Using MW_2 in practice

Optimal plans for MW_2 are not supported on the graph of a function and hence do not directly yield a transport map between the mixtures μ_0 and μ_1. In order to define a transport map from the optimal plan γ^* we can for instance use

$$T(x) = \mathbb{E}_{(X,Y)\sim\gamma^*}[Y|X = x]. \tag{5}$$

As shown in [6], the closed-form formula for T is given by

$$T(x) = \frac{\sum_{k,l} w_{k,l}^* g_{m_0^k, \Sigma_0^k}(x) T_{k,l}(x)}{\sum_k \pi_0^k g_{m_0^k, \Sigma_0^k}(x)}, \tag{6}$$

where w^* is the optimal solution of the discrete problem (3), $T_{k,l}$ are the optimal affine maps between Gaussians μ_0^k and μ_1^l and $g_{m,\Sigma}$ is the density of $\mathcal{N}(m, \Sigma)$. In the following, this map (6) will be called the GMM-OT map.

As shown in [6], this allows to express also the MW_2-barycenters between μ_0 and μ_1 with a closed-form formula:

$$\forall \alpha \in [0,1] \quad \mu_\alpha = \sum_{k,l} w_{k,l}^* \big((1-\alpha)\mathrm{Id} + \alpha T_{k,l}\big)\sharp\mu_0, \tag{7}$$

where $T\sharp\mu$ is the pushforward measure of the measure μ by the map T.

3 TextoGMM, a multiscale texture synthesis approach with optimal transport between patches

Let $u : \Omega \to \mathbb{R}^d$ be a texture defined on a rectangle $\Omega \subset \mathbb{Z}^2$. Let U be an initialization for the synthesized texture. The idea of TextoGMM, inspired by [8], is to use OT to force the patch distribution of U to look like the one of u at several scales. The synthesized texture is then recomposed from this patch distribution by simple local averages. In what follows, we denote by $\omega = \{0, \dots, \mathsf{w} - 1\}^2$ the patch domain, and by $u_{|a+\omega} \in \mathbb{R}^\omega$ the patch at position a in u.

3.1 Monoscale Model

We start by describing the synthesis at a single scale. To initialize the process, we generate a stationary Gaussian random field U with the same mean and covariance as the example u, defined by

$$\forall a \in \mathbb{Z}^2, \quad U(a) = \bar{u} + \frac{1}{\sqrt{|\Omega|}} \sum_{b \in \Omega} (u(b) - \bar{u}) W(a - b) \tag{8}$$

where $\bar{u} = \frac{1}{|\Omega|} \sum_{a \in \Omega} u(a)$ and where W is a random field on $\mathbb{Z}^2$ whose pixel values are i.i.d. with distribution $\mathcal{N}(0, 1)$.

The random field U has some global features of u, but not its details. The distribution $\hat{\mu}$ of patches of U is then sent by optimal transport towards the distribution $\hat{\nu}$ of patches of u in order to reimpose these details on the synthesized image. To make this transport calculation fast, we use here GMM-OT as described in the previous section, approximating the discrete distributions $\hat{\mu}, \hat{\nu}$ by GMMs μ and ν using the EM algorithm[4]. The formula (5) allows us to deduce a transport map $T : \mathbb{R}^\omega \to \mathbb{R}^\omega$ in the patch space from the GMM-OT plan that solves $MW_2(\mu, \nu)$. Finally, the new synthesized image V is computed by averaging all the transported patches:

$$\forall a \in \mathbb{Z}^2, \quad \tilde{U}(a) = \frac{1}{|\omega|} \sum_{h \in \omega} T(U_{|a-h+\omega})(h). \tag{9}$$

Note that because of averaging, the distribution of patches of $\tilde{U}$ is not quite that of the transported patches. Imposing more precisely the distribution of patches of the synthesized image would require more sophisticated techniques.

3.2 Multiscale Model

Let us now extend the previous model to several scales. For $0 \leq s \leq S - 1$, consider a subsampled version u_s of u defined on the subdomain $\Omega_s \subset 2^s \mathbb{Z}^2$. At the coarse scale, U_{S-1} is initialized as the Gaussian field (8) estimated from u_{S-1}.

[4] It would be interesting here to have a GMM estimation method that directly minimizes a transport cost between the GMM and the discrete patch distribution.

Now assume that U_s at scale $s \in \{1, \dots, S-1\}$ is given. Again, we estimate GMMs μ_s and ν_s from the patch distributions of U_s and u_s, and derive a GMM-OT map T_s between μ_s and ν_s, and then recompute a synthesized image $\tilde{U}_s$ at scale s from the transported patches:

$$\forall a \in 2^s\mathbb{Z}^2, \quad \tilde{U}_s(a) = \frac{1}{|\omega|} \sum_{b \in 2^s\omega} T_s(U_{s|a-b+2^s\omega})(b). \tag{10}$$

However, for the need of the upcoming upsampling step, we need to compose with a L^2 nearest-neighbor (NN) projection on the exemplar patches at scale s. Therefore, we set

$$\forall a \in 2^s\mathbb{Z}^2, \quad V_s(a) = \frac{1}{|\omega|} \sum_{b \in 2^s\omega} u_s\big(C_s(a-b)+b\big), \tag{11}$$

where the coordinate map C_s is defined by

$$\forall a \in 2^s\mathbb{Z}^2, \quad C_s(a) = \underset{a' \text{ st. } a'+2^s\omega \subset \Omega_s}{\operatorname{Argmin}} \|T_s(U_{s|a+2^s\omega}) - u_{s|a'+2^s\omega}\|^2. \tag{12}$$

This mechanism makes it easy to initialize synthesis at the next scale, by taking patches twice as large at the same positions. More precisely, U_{s-1} is initialized by setting for all $a \in 2^s\mathbb{Z}^2$ and all $k \in \{0, 2^{s-1}\}^2$,

$$U_{s-1}(a+k) = \frac{1}{|\omega|} \sum_{b \in 2^s\omega} u_{s-1}\big(C_s(a-b)+b+k\big). \tag{13}$$

At the end of the process, we obtain the synthesized image V_0. This *coarse-to-fine* synthesis process is illustrated on Fig. 1. Let us mention that, once the model estimated (i.e. all GMMs and transport plans computed from one synthesis), it can be used off-line to do image synthesis on demand (and of arbitrary size).

3.3 Adaptation to Style Transfer and Texture Mixing

The TextoGMM model can be easily adapted for style transfer and texture mixing. The adaptation to style transfer is a straightforward extension of the technique explained in [12] where the texture information can be treated with GMM-OT maps and then blended with the geometric features.

The adaptation to texture mixing requires more explanation, because it relies on a benefit of the GMM-OT cost, which is to have closed-form barycenters. For mixing, we exploit the explicit formula (7) that gives the expression of the MW_2 barycenter between μ_0 and μ_1. Let us fix a parameter $\alpha \in [0,1]$ that controls the mixing between the texture models associated with two source images u^0, u^1. For the initialization at the coarse scale, we can rely on the Gaussian model U^{S-1} obtained as the W_2 barycenter of the Gaussian models associated to u^0 and u^1, which can still be expressed as a convolution of a Gaussian white noise with an explicit image as in (8) [16]. For the patch transport at each scale s we apply a

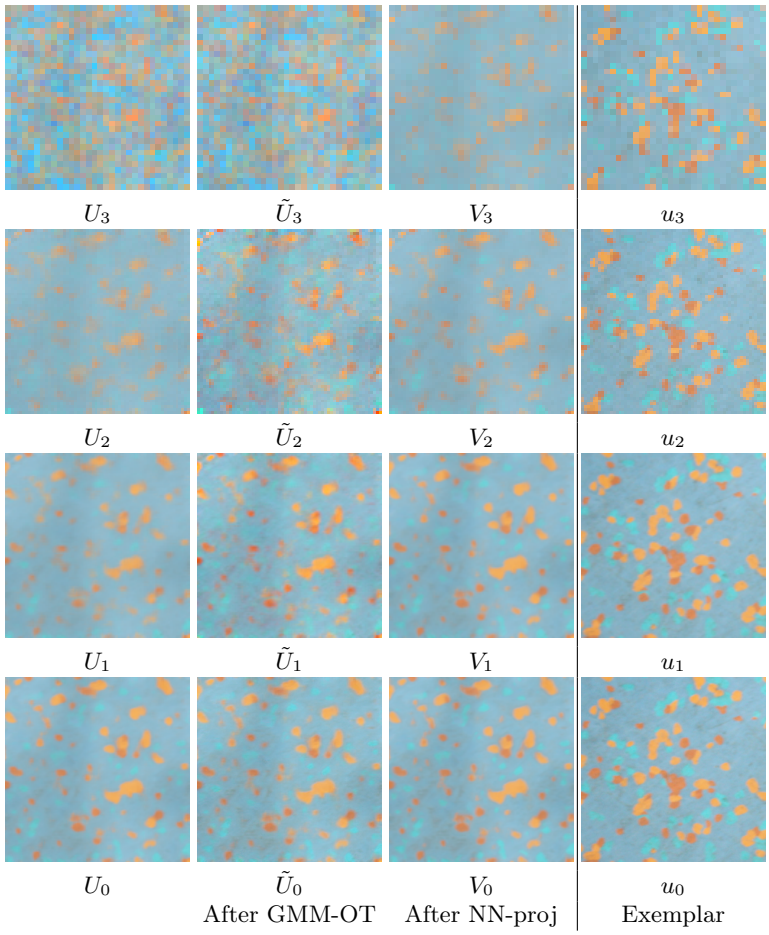

Fig. 1. Coarse-to-fine process ($\mathsf{w} = 5, S = 4$). On this figure, we illustrate the coarse-to-fine synthesis process of one exemplar texture (shown on the 4th column). It must be read from left to right and then top to bottom. At the coarse scale (here $S - 1 = 3$), the synthesis is initialized with the Gaussian field U_3. At each scale, from the current synthesis (U_s, 1st column), the patches are first transported with GMM-OT ($\tilde{U}_s$, 2nd column) and then projected back on exemplar patches with a NN projection (V_s, 3rd column). One can notice here the effect of the patch GMM-OT maps that will help to better cover the exemplar patch distributions, thus counter-acting the effects of NN projections that may sometimes flatten the image dynamic (especially for coarse scales with fewer patches).

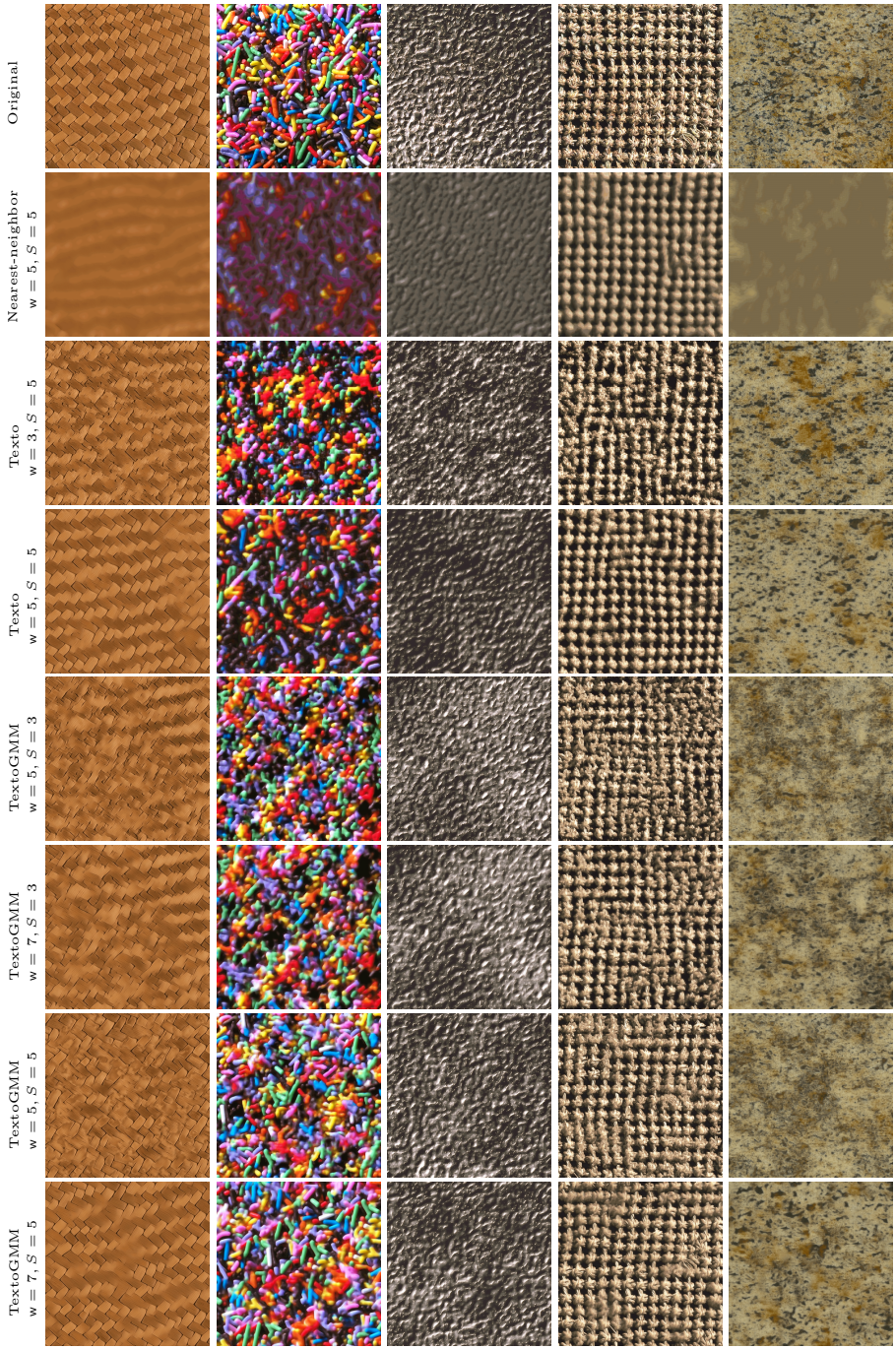

Fig. 2. Synthesis results. In this figure, we display several 512×512 original texture images (row 1), and synthesized images obtained with different models explained in the paper, with different parameters w (patch size) and S (number of scales). In row 2, the synthesis is obtained by using only patch nearest-neighbor projections at each scale. And then, we display the results obtained with the previous Texto model [8] (rows 3,4) and the TextoGMM model proposed here (rows 5–8).

GMM-OT map that targets the mixed patch distribution ν_s^α obtained by mixing the GMM patch distributions ν_s^0, ν_s^1 associated with u^0, u^1.

But a delicate point is that, for mixing, one cannot use a direct NN projection on exemplar patches to perform exemplar-based upsampling, because there is a priori no "mixed exemplar" that is available. We solve this issue by creating a collection of mixed patches by relying, again, on the GMM-OT map T (6) between ν_s^0, ν_s^1: once this transport map obtained, the mixed patches are defined as a linear interpolation $(1-\alpha)p + \alpha q$ between a patch p of ν_s^0 and the nearest neighbor $q = P_{NN}(T(p))$ of the transported patch $T(p)$ in the patches of ν_s^1. Since p and q are both patches taken from the original images at scale s, it is possible to compute the corresponding mixed patch at the next scale, by interpolating twice-larger patches taken at the same positions.

4 Experiments

Implementation details. In order to keep a reasonable computational time, even for very large images and large patches, we do not use all patches in the EM algorithms and the NN projection steps. More precisely, the GMM distributions μ_s, ν_s at each scale are estimated using only $N_p = 10^4$ patches (or less for small images) randomly taken in U_s, u_s respectively. Such subsampling of the patch distribution was observed to be harmless for the texture synthesis application and speed-up these two steps. For the NN projection step (12), we also use the same set of patches taken in u_s. Notice also that, in order to fasten the whole synthesis process, the patch operations (extraction, aggregation and NN projections) must be performed efficiently by relying on existing libraries. These NN projections may even be accelerated with dedicated algorithms [13].

Let us emphasize on the fact that the TextoGMM model associated to a texture can be computed with one pass of analysis-synthesis. Once the model has been estimated, it can be sampled on-the-fly by directly applying the pre-learnt GMM-OT maps, NN projections and upsampling step at each scale.

Computation time. TextoGMM allows for a much faster estimation step than the original Texto model [8], even though TextoGMM handles 10 times more patches. Indeed, the GMM-OT algorithm runs faster than the stochastic algorithm used in [8] for semi-discrete OT or even faster than the Sinkhorn algorithm [5]. For instance, on an OT problem with 10^4 points in source and target distributions on $\mathbb{R}^{147}$ (for 7×7 color patches), with a modern laptop with parallel CPU computations, solving GMM-OT takes $\approx 1'$, to be compared with $2.5h$ for 10^5 iterations of the stochastic OT algorithm, or $40'$ to perform 10^3 iterations of Sinkhorn algorithm. With 5×5 patches, for an image of size 256×256, the whole analysis-synthesis algorithm for TextoGMM (with $N_p = 10^4$) takes ≈ 15" while the analysis-synthesis algorithm for Texto (with $N_p = 10^3$) takes more than $1h$. In the same setting, once the model estimated, one synthesis takes ≈ 5". Therefore, a major benefit of the TextoGMM model is to considerably fasten the model estimation, while allowing for larger patches.

Visual comments. Several synthesis examples are displayed on Fig. 1, Fig. 2, Fig. 3. In the captions, the output of the multiscale process explained in Section 3.2 is referred to as TextoGMM. In all the results of TextoGMM shown here, the estimated GMM models have 4 components, except in Fig. 4.

GMM-OT allows to partially cope with the curse of dimensionality, and thus permits to use much larger patches in TextoGMM than Texto (also because GMM-OT can handle distributions with much more points). This leads to more faithful synthesis of structured textures, with a better preservation of the sharp details, as illustrated in Fig. 2. The second row of Fig. 2 confirms the importance of using OT for patch transformation and not only simple NN matching. Also, the third and fourth rows confirm that the previous Texto model works only with small patches. In contrast, TextoGMM produces remarkable results on these textures with the proper choice of parameters w and S. In Fig. 3, we observe that the TextoGMM attains a visual quality similar to the model of [12] based on a multi-layer approximation of OT, while allowing for much faster estimation. Also, compared to recent texture synthesis methods, TextoGMM is able to reproduce large strutures in a coherent way, while allowing for much faster estimation. In Fig. 4, we vary the number of number of Gaussian components K used in the GMM. Even if the visual details are more precisely retrieved when using more components, one can see that the syntheses obtained with very few Gaussian components (even 1 or 2!) appear already convincing. For $K = 1$, this illustrates the capacity of a very light texture synthesis algorithm based only on affine transformations and NN projections.

Finally, on Fig. 5 and Fig. 6, we display some results of style transfer and texture mixing, respectively, with the TextoGMM model. For both these applications, the visual results appear convincing. In particular, for texture mixing, it is interesting to notice how the intermediate patch models are able to mix some structures seen in the exemplar textures u^0, u^1. To confirm the relevance of this approach, it would be interesting to compare the output distribution of the intermediate images to the true W_2-barycenter obtained with the patch distributions of u^0, u^1.

5 Conclusion

In this paper we proposed to exploit a new formulation of optimal transport specific to Gaussian mixture models, in order to improve the texture model Texto which is based on optimal transport in patch space. This new formulation allows to work with larger patches using a very simple parameterization of the transportation maps. Compared to Texto, it thus brings a clear improvement on the visual quality of generated textures, while considerably reducing the computational time required for estimating the model. Also, this model allows for fast on-the-fly synthesis because it only needs, from coarse to fine scales, a few affine transformations of the patches composed with a nearest neighbor search. Finally, this texture model can also be used for style transfer or texture mixing, exploiting the closed-form barycenters for the GMM-OT cost.

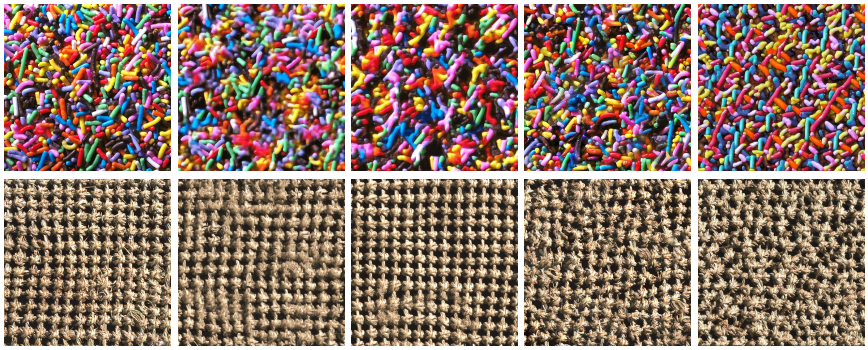

Fig. 3. Comparison with other synthesis algorithms. We display synthesis results obtained with several recent models: the here proposed TextoGMM model, the TextoML model from [12] also based on OT (both with patch size $\mathsf{w} = 5$ and $S = 5$ scales), and the neural-network based techniques from [9] and [14]. One can see that the synthesis quality with TextoGMM gets close to the one attained by TextoML, while keeping a considerably simpler model estimation step. One can also observe that TextoGMM is able to recover some complex and large structures (since it works on large patches) as [9] and [14], but produces blurrier results due to patch aggregation.

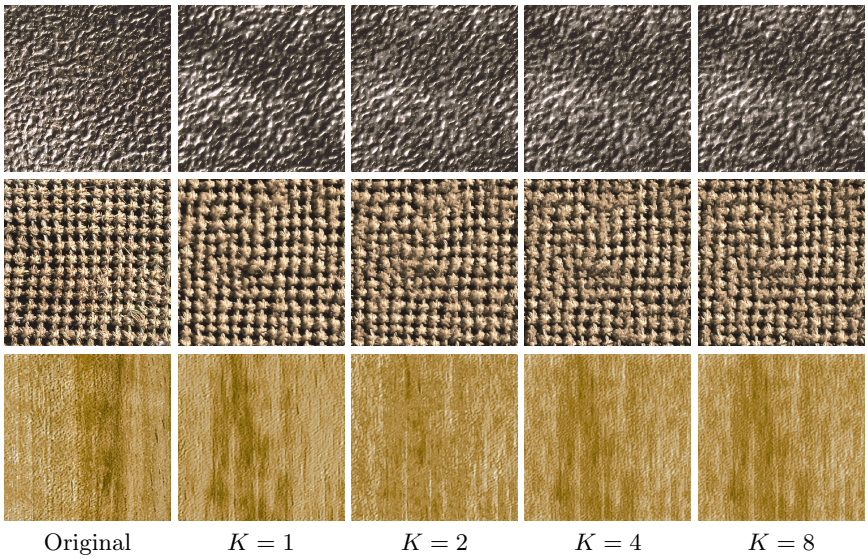

Fig. 4. Number of Gaussian components with $(\mathsf{w} = 5, S = 4)$. We display synthesis results obtained with the TextoGMM model with varying number of components in the GMM. Even if results appear surprisingly good for $K = 1$, increasing K allows to better retrieve fine details of the exemplar.

The main limitation of this new model is now the practical complexity of the EM algorithm used to approximate empirical patch distributions by GMM. Even if the number of used components can be set small (between 1 and 10), the practical behavior of the EM algorithm applied to very rich empirical distributions in high dimension remains problematic. It would be interesting to examine more thoroughly the impact of the obtained GMM approximation on the quality of synthesized textures, and to see if another GMM learning algorithm could be used in order to better scale up to the dimension of the patch space. Also, it would be interesting to compare more thoroughly the GMM-OT approximation of the OT cost with the semi-discrete multilayer approximation of [12]. These methods are respectively based on a soft and hard clustering of a target distribution, and it would be useful to draw a theoretical connection between them.

Fig. 5. Style transfer with TextoGMM. Adapting the style transfer technique from [12] to the TextoGMM model produces convincing style transfer results.

References

1. Bonneel, N., Van De Panne, M., Paris, S., Heidrich, W.: Displacement interpolation using Lagrangian mass transport. In: Proceedings of the 2011 SIGGRAPH Asia conference. pp. 1–12 (2011)
2. Bonnotte, N.: Unidimensional and evolution methods for optimal transportation. Ph.D. thesis, Paris 11 (2013)
3. Chizat, L., Roussillon, P., Léger, F., Vialard, F.X., Peyré, G.: Faster Wasserstein distance estimation with the Sinkhorn divergence. Advances in Neural Information Processing Systems **33**, 2257–2269 (2020)
4. Courty, N., Flamary, R., Habrard, A., Rakotomamonjy, A.: Joint distribution optimal transportation for domain adaptation. Advances in Neural Information Processing Systems **30** (2017)
5. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. In: Advances in neural information processing systems. pp. 2292–2300 (2013)
6. Delon, J., Desolneux, A.: A Wasserstein-type distance in the space of Gaussian mixture models. SIAM Journal on Imaging Sciences **13**(2), 936–970 (2020)
7. Feydy, J., Roussillon, P., Trouvé, A., Gori, P.: Fast and scalable optimal transport for brain tractograms. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 636–644. Springer (2019)

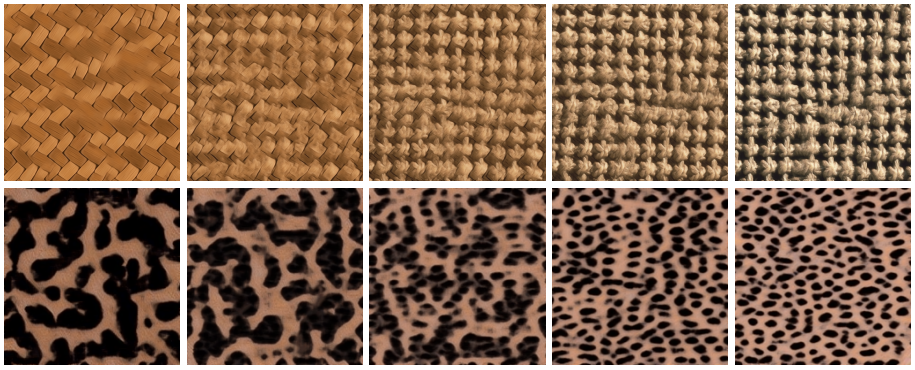

Fig. 6. Texture mixing. We display here examples of texture mixing computed from two source images u_0, u_1 (not shown in this figure). From left to right, we display a sample of the mixed texture model with mixing parameter $\alpha \in [0, 1]$. For $\alpha = 0, 1$, we get a sample of the TextoGMM model associated with u_0, u_1. It is interesting to see how the mixed TextoGMM model is able to combine the geometric structures of u_0, u_1. Parameters are $(\mathsf{w}, S) = (7, 5)$ for the 1st row and $(5, 4)$ for the 2nd row.

8. Galerne, B., Leclaire, A., Rabin, J.: A texture synthesis model based on semi-discrete optimal transport in patch space. SIAM Journal on Imaging Sciences **11**(4), 2456–2493 (2018)
9. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2414–2423 (2016)
10. Genevay, A., Chizat, L., Bach, F., Cuturi, M., Peyré, G.: Sample complexity of Sinkhorn divergences. In: The 22nd international conference on artificial intelligence and statistics. pp. 1574–1583. PMLR (2019)
11. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4401–4410 (2019)
12. Leclaire, A., Rabin, J.: A stochastic multi-layer algorithm for semi-discrete optimal transport with applications to texture synthesis and style transfer. Journal of Mathematical Imaging and Vision **63**(2), 282–308 (2021)
13. Liang, L., Liu, C., Xu, Y.Q., Guo, B., Shum, H.Y.: Real-time texture synthesis by patch-based sampling. ACM Transactions on Graphics (ToG) **20**(3), 127–150 (2001)
14. Ulyanov, D., Lebedev, V., Vedaldi, A., Lempitsky, V.: Texture networks: feed-forward synthesis of textures and stylized images. In: Proc. of the Int. Conf. on Machine Learning. vol. 48, pp. 1349–1357 (2016)
15. Weed, J., Bach, F.: Sharp asymptotic and finite-sample rates of convergence of empirical measures in Wasserstein distance. Bernoulli **25**(4A), 2620–2648 (2019)
16. Xia, G., Ferradans, S., Peyré, G., Aujol, J.: Synthesizing and Mixing Stationary Gaussian Texture Models. SIAM Journal on Imaging Sciences **7**(1), 476–508 (2014)