Conditional Gaussian Models for Texture Synthesis

Lara Raad(✉), Agnès Desolneux, and Jean-Michel Morel

CMLA, Ecole Normale Supérieure de Cachan, Cachan, France
lara.raad@cmal.ens-cachan.fr

Abstract. An ideal exemplar-based texture synthesis algorithm should create a new texture that is perceptually equivalent to its texture example. To this goal it should respect the statistics of the example and avoid proceeding to a "copy-paste" process, which is the main drawback of the non-parametric approaches. In a previous work we modeled textures as a locally Gaussian patch model. This model was estimated for each patch before stitching it to the preceding ones. In the present work, we extend this model to a local conditional Gaussian patch distribution. The condition is taken over the already computed values. Our experiments here show that the conditional model reproduces well periodic and pseudo-periodic textures without requiring the use of any stitching technique. The experiments put also in evidence the importance of the right choice for the patch size. We conclude by pointing out the remaining limitations of the approach and the necessity of a multiscale approach.

Keywords: Texture synthesis · Conditional locally gaussian · Patch size

1 Introduction

Exemplar based texture synthesis is the problem of synthesizing from an input sample a perceptually equivalent larger texture sample. This problem has applications in computer graphics, computer vision and image processing, for example for fast scene generation, inpainting, and texture restoration. From the mathematical viewpoint the problem can be posed as the retrieval of the stochastic process underlying the sample, followed by the generation of new samples by simulating the reconstructed stochastic process. Texture synthesis methods are generally divided into two categories, the parametric [5,7,14] and non-parametric [1,3,4,10–12,17]. Using the texture sample, the parametric methods estimate statistical parameters characterizing the stochastic process. Although these methods can reproduce faithfully some of the global statistics of the sample, they generally do not yield high quality visual results. The non-parametric methods give up any statistical modeling and proceed to a direct simulation by iterating a copy-paste process with texture patches extracted from the original sample. Even though these methods can yield satisfactory results, they lack flexibility and often turn into practising verbatim copies of large parts of the

J.-F. Aujol et al. (Eds.): SSVM 2015, LNCS 9087, pp. 474–485, 2015.
DOI: 10.1007/978-3-319-18461-6_38

input sample. More recently methods such as the work of Tartavel et al. in [16] combine the parametric and non-parametric methods using a variational approach with the aim of overcoming the copy-paste effects. They propose to minimize an energy function that takes into account the use of a sparse, patch-based dictionary combined to constraints on the textures' spectrum.

Statistics-based methods were initiated by Julesz [9], who discovered that many texture pairs having the same second-order statistics would not be discerned by human preattentive vision. This hypothesis is referred to as the first Julesz axiom for texture perception. Its validity can been checked in [5] which proposes to emulate microtextures by maintaining the Fourier modulus of the sample image and randomizing its phase, thus maintaining the second order statistics of the sample. This random phase method correctly emulates certain textures, which turn out to be Gaussian stationary fields. Yet, the method also fails for more structured ones. Both effects can be experimented online in the executable paper [6].

Julesz' approach was extended by Heeger and Bergen in [7] using multiscale statistics. The texture sample is characterized by the histograms of its wavelet coefficients and a new texture is created by enforcing these statistics on a white noise image. Yet this method only measures marginal statistics of the sample and misses important correlations between pixels across scales and orientations. Again, the model gives convincing results on certain textures, but for example fails on oriented textures, as verifiable in the executable paper [8].

Heeger and Bergen's model was extended by Portilla and Simoncelli in [14] who proposed to estimate on the texture sample not less than 700 autocorrelations, cross-correlations and statistical moments of the wavelet coefficients. The synthesis results obtained with this method are strikingly good compared to the previous statistical attempts. Convincing results are observable on a very wide range of textures. Although this method represents the state of the art for psychophysically and statistically founded algorithms, it has too many parameters to learn, and the results often present blur and phantoms effects.

Neighbourhood-based methods were initialized by Efros and Leung [4]. This method extends to images Shannon's Markov random field model for the English language. The simulated texture is constructed pixelwise. For each new pixel, a patch centered at the pixel is compared to all patches with the same size in the input sample. The nearest matches help predict the pixel value in the reconstructed image. This method was significantly accelerated by Wei and Levoy in [17] who fixed the shape and size of the learning patch and also by Ashikmin in [1] who proposed to extend existing patches whenever possible instead of searching in the entire sample texture. These pixelwise algorithms are not quite satisfactory. They are known to produce "garbage" when the compared patches are too small, or may lead to verbatim copies of significant parts of the input sample for large ones. To resolve the "garbage" issue and accelerate the procedure the more recent methods stitch together entire patches instead of synthesizing pixel by pixel. The question then is how to blend a new patch in the existing texture. In [12] this is done by a smooth transition. Efros and Freeman [3] refined this process by stitching each new patch along a minimum cost path across its overlapping zone with the texture under construction. Kwatra et al. in [11] extended the stitching procedure of [3] by a graph cut approach where the edges of the patches are redefined and then quilted in the synthesis image. They also proposed in [10] to synthesize a texture by sequentially improving the quality of the synthesized image by minimizing a patch-based energy function. The patch-based approaches often present satisfactory visual results. Yet, there is still a risk of copying verbatim large parts of the input sample. Furthermore, a respect of the global statistics of the initial sample is not guaranteed.

In a previous work [15] we proposed an algorithm that introduces some degree of statistical modeling in the non-parametric approach, thus trying to blend both approaches. A texture is synthesized in a patch-based approach, but the copy-paste process is preceded by the estimation of a Gaussian texture model for each new patch. In [15] we completed this process by adding a blending step in the spirit of [3].

In this work, we present a refinement of the stochastic modeling introduced in [15]. Our motivation was to dispose of the patch stitching step by using a more robust local model for the texture. To that end, we condition the Gaussian distribution of a patch to the values of its overlapping region and simulate the patch as the most probable sample of that conditional distribution. We present two different models: a conditional locally Gaussian where the overlapping region is maintained as is, and a regularized version that slightly relaxes the conditioning on the overlap region. Using this refined model, we shall prove that it is possible to avoid the blending step for a wide class of periodic or pseudo-periodic textures. Yet our experiments also show that for macro-textures we still need a blending step.

We finally give an interpretation of this fact, that points to the necessity of a multiscale generalization of the conditional approach, to cope with complex multiscale textures.

The rest of this paper is structured as follows. In Section 2 we present two conditional Gaussian models for the patches: a conditional locally Gaussian (CLG) and a regularized conditional locally Gaussian (RCLG). In Section 3 we discuss the synthesis results obtained from the three models: the locally Gaussian ([15]) and the two conditional locally Gaussian. We also analyze the behavior of the RCLG model when varying the patch size and the neighbourhood size. Conclusions are presented in Section 4.

2 Patch Models

We modeled in [15] a texture as a Locally Gaussian (LG) distribution. Given an input texture I_0, an output image I_s is synthesized sequentially patch by patch in a raster-scan order (left to right, top to bottom). Each new patch added to I_s overlaps part of the previously synthesized patch as can be seen in Figure 1. Each patch is simulated following a multivariate Gaussian distribution of mean μ and covariance matrix Σ where

$$\mu = \frac{1}{N}\sum_{i=1}^{N} p_i \text{ and } \Sigma = \frac{1}{N}(P-\mu)(P-\mu)^t, \tag{1}$$

P is a matrix whose columns are the patches p_i in vector form and N is the considered number of nearest neighbours.

To define the patches p_i, let us consider p as the patch being currently synthesized and taken as a column vector of size $w \times 1$. The patch p will overlap part of the previous synthesis. To synthesize p, we decompose it as

$$p = \begin{pmatrix} Sp \\ Mp \end{pmatrix}, \; S : \mathbb{R}^w \to \mathbb{R}^{w-k}, \; M : \mathbb{R}^w \to \mathbb{R}^k \tag{2}$$

where S and M are projection operators such that Mp is a vector of size $k \times 1$ with the values of p on the overlap area and Sp is a vector of size $(w - k) \times 1$ with the other components of p.

The patches p_i used to learn the parameters of the multivariate Gaussian distribution (1) are the N nearest neighbours in I_0 to the current patch p, for the L^2 distance restricted to the overlap area, given by $\|Mp_i - Mp\|_2$. Once the patch p is synthesized from the Gaussian model (1), the values of Mp change.

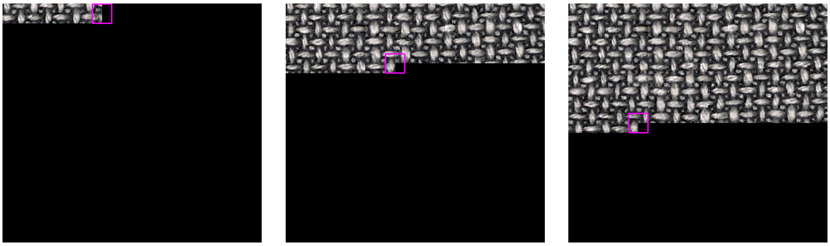

Fig. 1. We show three different iterations of the synthesis process. At each iteration a patch is being synthesized. This patch is represented by the pink square in the three iterations shown.

We observed unwanted transition effects when new patches were added into I_s. To overcome this problem we used Efros and Freeman's stitching method [3]. Yet we found that this solution is not quite satisfactory, as the new patch is simulated without conditioning it to the overlap area. We therefore propose a new patch model that aims to model directly the transition effect between patches. Each new patch will be estimated as a Gaussian vector conditioned to the pixel values of the corresponding overlap region. In this way the simulated patch would naturally "agree" with I_s in the overlap area, thus avoiding a stitching procedure.

2.1 The Conditional Locally Gaussian Model

Let I_s denote the texture image that is being synthesized. At each step of the algorithm a new patch p is added to I_s overlapping the previously synthesized ones. We want p to match with the overlap area pixels, to avoid creating unwanted discontinuities. This can be done by a Conditional Locally Gaussian (CLG) model.

Each patch p is taken as a column vector of size $w \times 1$. Then p can be partitioned as in (2).

We assume throughout that the vector p follows a Gaussian distribution of mean μ and covariance matrix Σ (the LG model). Then these parameters can be partitioned as follows :

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} = \begin{bmatrix} S\mu \\ M\mu \end{bmatrix} \quad \text{and} \quad \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} = \begin{bmatrix} S\Sigma S^t & S\Sigma M^t \\ M\Sigma S^t & M\Sigma M^t \end{bmatrix}.$$

Our problem can be formulated as finding the "best sample" $\tilde{p} = (\tilde{x}_1, \tilde{x}_2)$ conditioned to the overlap values y_0 that are known, i.e. we have $\tilde{x}_2 = y_0$ and the value of $\tilde{x}_1$ is the most probable one conditioned to $\tilde{x}_2$. It is given by

$$\tilde{x}_1 = \arg\max_{x_1} \mathbb{P}_{\mu,\Sigma}(Sp = x_1 \mid Mp = y_0) \tag{3}$$

By classic results on conditional multivariate Gaussian distributions [13], the distribution of Sp conditioned to $Mp = y_0$ is a multivariate Gaussian distribution of parameters $\bar{\mu}$ and $\bar{\Sigma}$ where

$$\bar{\mu} = \mu_1 + (S\Sigma M^t)(M\Sigma M^t)^{-1}(y_0 - \mu_2)$$

and

$$\bar{\Sigma} = (S\Sigma S^t) - S\Sigma M^t(M\Sigma M^t)^{-1}M\Sigma S^t.$$

Since the most probable sample of a multivariate Gaussian distribution is its mean, the solution to $(\mathcal{P}_1)$ is

$$\tilde{p} = \begin{pmatrix} \tilde{x}_1 \\ y_0 \end{pmatrix} = \begin{pmatrix} \mu_1 + (S\Sigma M^t)(M\Sigma M^t)^{-1}(y_0 - \mu_2) \\ y_0 \end{pmatrix}. \tag{4}$$

Feasibility. Yet equation (4) shows that this solution does not make sense if $(M\Sigma M^t)$ is not invertible. This unfortunately is frequent, as the number of neighbours N used to build the Gaussian distribution is often very small compared to the dimension of the vectors we aim to model. Therefore the learnt Gaussian models are strongly degenerated.

This does not necessarily imply that there is no solution to $(\mathcal{P}_1)$. The fact that Σ is not invertible implies that the Gaussian vectors $p \sim \mathcal{N}(\mu, \Sigma)$ live in a subspace of $\mathbb{R}^w$. This leads to the following alternative:

1. The Gaussian vectors subspace intersects the set of Gaussian vectors $(x_1, y_0)^t$.
2. There is no intersection and in that case no solution to our problem.

To overcome the fact that we may have no solution one can modify the Gaussian distribution learnt for p as follows

$$p \sim \mathcal{N}(\mu, \Sigma + \sigma^2 I_w),$$

where σ^2 is a real positive number and I_w is the identity matrix of size $w \times w$. In that way the Gaussian vectors p live in $\mathbb{R}^w$, and this ensures the existence of a solution to problem $(\mathcal{P}_1)$. We denote $\Gamma = \Sigma + \sigma^2 I_w$. When $(\mathcal{P}_1)$ has a solution for $p \sim \mathcal{N}(\mu, \Sigma)$, the new distribution $\mathcal{N}(\mu, \Gamma)$ will slightly modify the solution in (4) for a small value of σ^2. It is thus enough to take a low value for this parameter and the solutions obtained in both cases (with and without the Gaussian noise) will be very close to each other.

2.2 Regularized Conditional Locally Gaussian

In the previous section we conditioned the patches' statistical model to the exact values of the synthesized pixels across the overlap area. This could be too restrictive and then create samples that are very unlikely to exist. Instead of forcing each patch p to take the exact same values on the previously synthesized part, it is therefore natural to allow the patch p to vary slightly on the overlap area. This variation is rendered necessary by the scarcity of patch samples in a small texture sample. Consider the same patch model $\mathcal{N}(\mu, \Sigma)$, but let us now allow the overlap components Mp to take values $x_2 = y_0 + n$ where $n \sim \mathcal{N}(0, \theta^2 I_k)$. Then the most probable sample

$$\tilde{p} = (\tilde{x}_1, \tilde{x}_2) = \arg \max_{(x_1,x_2)} \mathbb{P}_{\mu,\Sigma}(Sp = x_1 | Mp = x_2) \mathbb{P}_{0,\theta^2 I_k}(x_2 - y_0)$$

still is exactly the same as in the CLG model. Thus, to slightly relax the constraint on the overlap we propose to minimize the following energy function

$$E(x_1, x_2) = -\log \mathbb{P}_{\mu,\Sigma}(Sp = x_1, Mp = x_2) + \frac{1}{2\theta^2} \|x_2 - y_0\|_2^2, \quad (5)$$

where the first term is the fidelity term to the Gaussian distribution of p and the second term is a regularization term to impose that x_2 keeps close to y_0. The parameter θ^2 controls the distance between the values of x_2 and the synthesized part over the overlap area. We shall call this model Regularized Conditional Locally Gaussian (RCLG). The optimal sample $\tilde{p}$ is then

$$\tilde{p} = (\tilde{x}_1, \tilde{x}_2) = \arg \min_{(x_1,x_2)} E(x_1, x_2) = (\theta^2 \Sigma^{-1} + M^t M)^{-1} (M^t y_0 + \theta^2 \Sigma^{-1} \mu) \quad (6)$$

Once again the solution in (6) makes sense when the inverse of the matrices Σ and $\theta^2 \Sigma^{-1} + M^t M$ exists. To be sure that we always find a solution, we shall slightly modify the Gaussian distribution of p, as we did previously, and this guarantees that these matrices are invertible. Considering $p \sim \mathcal{N}(\mu, \Sigma + \sigma^2 I_w)$ the optimal simulated patch $\tilde{p}$ is as shown in (7).

$$\tilde{p} = (\theta^2 (\Sigma + \sigma^2 I_w)^{-1} + M^t M)^{-1} (M^t y_0 + \theta^2 (\Sigma + \sigma^2 I_w)^{-1} \mu). \quad (7)$$

3 Experiments

The proposed patch models were tested with several types of textures: synthetic periodic, real pseudo-periodic and macro-textures. A summary of the results is presented in this section. The first results are a comparison of the three patch models: LG, CLG and RCLG without using any blending technique. The second ones compare the LG model and RCLG with an addional blending step. The last experiments are focused on comparing the influence of the parameters w and N on the synthesis results where w is the patch size and N the neighbourhood size (number of nearest neighbours used to build the Gaussian model). We do not compare here the proposed synthesis methods to other classical texture synthesis algorithm because this was already done extensively in [15]. Our aim is rather to discuss the different local Gaussian models and their parameters.

3.1 Patch Models Comparison

For the examples shown in Figure 2 the pacth size w varies between 10×10 and 40×40 and the neighbourhood size N between 10 and 20. The comparisons were made without using a blending step when stitching the patches. This permitted to better evaluate the positive impact of considering a local Gaussian model conditioned to the overlap values.

For the first example in Figure 2 one can observe that the three models achieve very good results regardless of the overlap information. This was expected, as periodic synthetic textures present many reliable examples to learn the Gaussian distribution of a patch. No boundary effect is therefore observable. For the LG model minor transitions are visible due to the presence of "dégradé" in this particular example. In the examples 2 and 3 in Figure 2 the stitching effects start being noticeable. For the LG model the patch itself is correctly synthesized but a blending step is necessary to avoid losing the global structure. However, for both models CLG and RCLG we still obtain a reliable patch model and a good global coherence without any blending technique. In the three last examples we meet the case where there are not enough patch examples in the sample texture to build an acceptable Gaussian distribution for the patch. Thus, achieving a correct synthesis requires adding a blending step. Once again, the transitions between the patches are more conspicuous in the LG model compared to the conditioned models.

The RCLG clearly yielded better results then the CLG model. This is due to the fact that we slightly relaxed the condition on the overlap area, thus achieving a more flexible model for the patch itself. For the conditional model (CLG) it might have seemed attractive to impose that a patch keeps exactly the same values in the overlap area. Indeed, this might have generated a patch that is very unlikely to happen. Yet, one can observe in the last two examples how the quality of the synthesis declines by this strong imposition.

In Figure 3 two models were compared: LG and RCLG, with an additional blending step for both cases. We can see in a general way that for the pair of parameters chosen for each texture example the RCLG model conserves a better

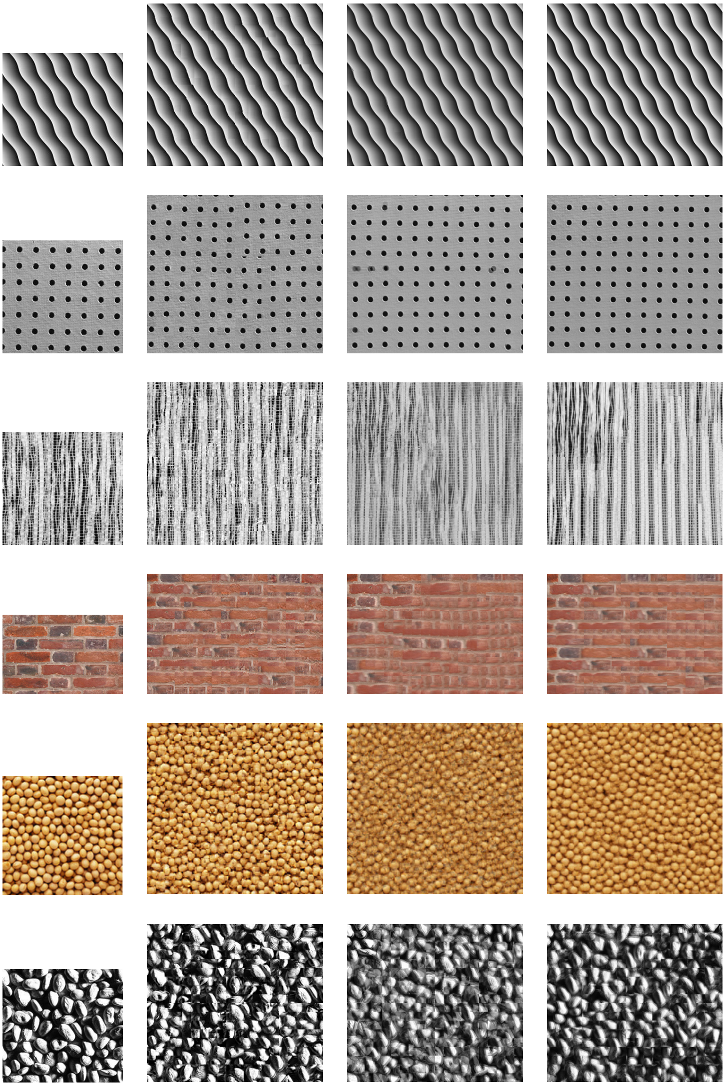

Fig. 2. Patch model comparison on several texture examples. From left to right: local Gaussian model, conditional local Gaussian model and regularized conditional local Gaussian model. The pair of parameters (w, N) used for each example from top to bottom are $(30, 20), (40, 10), (30, 10), (40, 10), (10, 10), (40, 20)$.

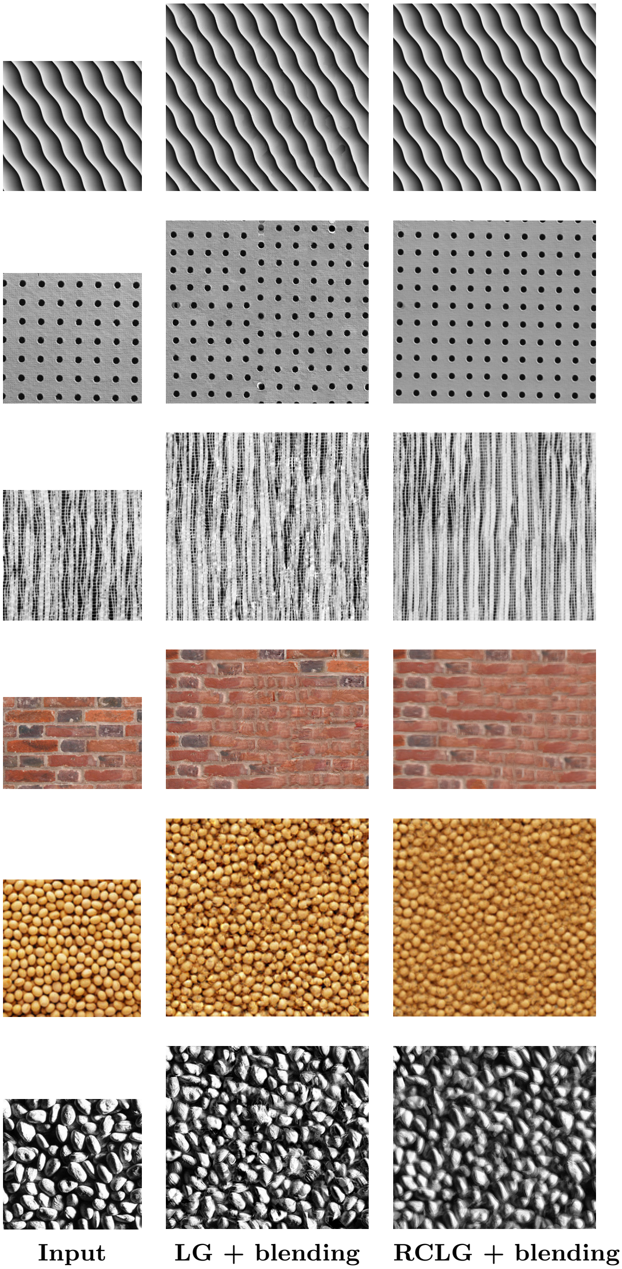

Fig. 3. Comparison of two patch models (LG an RCLG) combined with a blending step as in [3]. From left to right: input texture, local Gaussian model and regularized conditional local Gaussian model. The pair of parameters (w, N) used for each example from top to bottom are $(30, 20), (40, 10), (30, 10), (40, 10), (10, 10), (40, 20)$.

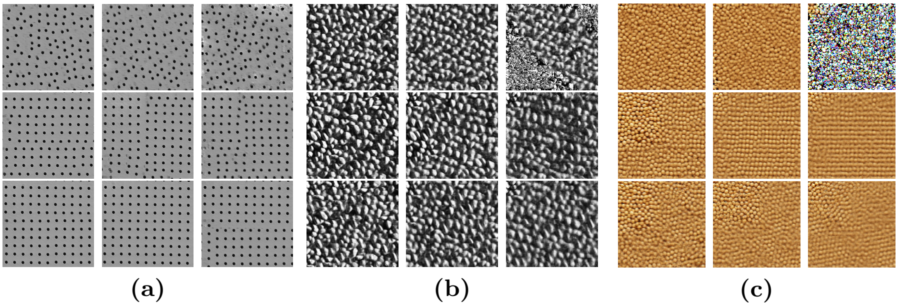

Fig. 4. Comparison of the patch size w and neighbourhood size N influence on three examples synthesis results using the RCLG model. For (a) and (b): from top to bottom, $w = 20, 30, 40$ and from left to right, $N = 10, 20, 50$. For (c): from top to bottom, $w = 10, 20, 30$ and from left to right, $N = 10, 20, 50$.

global coherence at the cost of losing finer details that the LG model manages to keep. These results are not surprising since all patches in the RCLG model were simulated minimizing (5) achieving a smoother result (it is the same as finding the most probable sample from the underlying probability distribution). A possible solution to overcome this effect could be to define the probability law of the RCLG model and sample a patch from it instead of taking the most probable one.

Finally, we noticed that on every synthesis example using any of the three patch models some details of the original texture can be lost (except for the periodic synthetic texture where there is no lack of examples). This brings us to the comparison of the next section, where we shall clearly see how varying the patch size permits on the one hand to capture the thinner details (using a smaller patch size) and on the other to capture the global structure of the texture (using a larger patch size).

3.2 Influence of Parameters

As mentioned above we will compare the influence of the patch size w and of the neighbourhood size N on the synthesis results. We can clearly conclude from the examples shown in Figure 4 that, by increasing the neighbourhood size for a fixed w, the RCLG patch model loses accuracy. Another important conclusion is that for different patch sizes the model is able to capture different details at different scales. In particular in the first example of Figure 4 for a patch of size 20×20 the salient structure is well simulated with the model, although the synthesis of the global arrangement of the black circles fails. Increasing the value of w permits to achieve a good global reconstruction, at the cost of sightly smoothing out the texture. The same observation can be made for the other displayed examples. We can conclude from these experiments that a unique patch size for the synthesis of macro-textures, i.e. containing information at different scales, is not enough to model at the same time the global structure and the finer details.

4 Conclusion

In this work, we have proposed a novel local texture sampling method in the patch space where each patch is conditioned to the values in the overlapping area.

The experiments in Section 3 show that both resulting models CLG and RCLG yield good synthesis results on periodic and quasi-periodic textures, without requiring the use of a stitching technique. For general macro-textures this need is reduced with CLG an RCLG with respect to the locally Gaussian model, that did not take into account the overlap values. Comparing both conditional models, CLG and RCLG, we observed that the results obtained for the regularized version are visually better. Indeed, imposing the simulated Gaussian patch to have exactly the same values over the overlap area was excessive. The experiments of Section 3.2 show the importance of fixing a correct patch size for a given texture example. They prove that a macro-texture cannot be correctly synthesized with a single patch size.

Hence, future work will focus on a multi-scale approach to catch both the global structure of the texture and its details, in the spirit of the recent work [16].

References

1. Ashikhmin, M.: Synthesizing natural textures. In: Proceedings of the 2001 symposium on Interactive 3D graphics, pp. 217–226. ACM (2001)
2. Desolneux, A., Moisan, L., Ronsin, S.: A compact representation of random phase and gaussian textures. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2012, Kyoto, Japan, March 25–30, 2012, pp. 1381–1384 (2012)
3. Efros, A.A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: SIGGRAPH, pp. 341–346 (2001)
4. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: IEEE International Conference on Computer Vision, pp. 1033–1038 (1999)
5. Galerne, B., Gousseau, Y., Morel, J.-M.: Random phase textures: Theory and synthesis. IEEE Transactions in Image Processing (2010)
6. Galerne, B., Gousseau, Y., Morel, J.-M.: Micro-Texture Synthesis by Phase Randomization. Image Processing On Line (2011). http://dx.doi.org/10.5201/ipol.2011.ggm_rpn
7. Heeger, D.J., Bergen, J.R.: Pyramid-based texture analysis/synthesis. In: SIGGRAPH, New York, NY, USA, pp. 229–238 (1995)
8. Briand, T., Vacher, J., Galerne, B., Rabin, J.: The Heeger and Bergen Pyramid Based Texture Synthesis Algorithm. Image Processing On Line **4**, 276–299 (2014). http://dx.doi.org/10.5201/ipol.2014.79
9. Julesz, B.: Visual pattern discrimination. IEEE Trans. Inf. Theory **8**(2), 84–92 (1962)
10. Kwatra, V., Essa, I., Bobick, A., Kwatra, N.: Texture optimization for example-based synthesis. In ACM Transactions on Graphics (TOG), vol. 24, pp. 795–802. ACM (2005)
11. Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A.: Graphcut textures: image and video synthesis using graph cuts. In: ACM Transactions on Graphics (TOG), vol. 22, pp. 277–286. ACM (2003)

12. 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)
13. Morrison, D.F.: Multivariate statistical methods. 3. New York, NY. Mc (1990)
14. Portilla, J., Simoncelli, E.P.: A parametric texture model based on joint statistics of complex wavelet coefficients. International Journal of Computer Vision **40**(1), 49–70 (2000)
15. Raad, L., Desolneux, A., Morel, J.-M.: Locally gaussian exemplar-based texture synthesis. In: IEEE International Conference on Image Processing, ICIP 2014, Paris (2014)
16. Tartavel, G., Gousseau, Y., Peyré, G.: Variational texture synthesis with sparsity and spectrum constraints. Journal of Mathematical Imaging and Vision (2014)
17. Wei, L.-Y., Levoy, M.: Fast texture synthesis using tree-structured vector quantization. In: SIGGRAPH, pp. 479–488 (2000)