

# Split and Match: Example-based Adaptive Patch Sampling for Unsupervised Style Transfer

Oriol Frigo<sup>1,2</sup>

Neus Sabater<sup>1</sup>

Julie Delon<sup>2</sup>

Pierre Hellier<sup>1</sup>

<sup>1</sup>Technicolor, Research&Innovation, France

<sup>2</sup>Université Paris Descartes, MAP5, France

## Abstract

*This paper presents a novel unsupervised method to transfer the style of an example image to a source image. The complex notion of image style is here considered as a local texture transfer, eventually coupled with a global color transfer. For the local texture transfer, we propose a new method based on an adaptive patch partition that captures the style of the example image and preserves the structure of the source image. More precisely, this example-based partition predicts how well a source patch matches an example patch. Results on various images show that our method outperforms the most recent techniques.*

## 1. Introduction

Style transfer is the task of transforming an image in such a way that it mimics the style of a given example. This class of computational methods are of special interest in film post-production and graphics, where one could generate different renditions of the same scene under different “style parameters” [17] [6]. The difficulty of this task is bound to the complexity of defining the style as a composition of different visual attributes such as color, shading, texture, lines, strokes and regions.

Example-based methods have been widely employed to solve problems such as texture synthesis [7], inpainting [5], and super-resolution [10] with state-of-the-art performance. These non-local and non-parametric approaches draw on the principle of self-similarity in natural images: similar patches (sub-images) are expected to be found at different locations of a single image.

Despite the practical success of patch-based methods for inverse problems, the patch dimensionality remains a sensitive parameter to tune in these algorithms. For instance, to obtain a coherent patch-based texture synthesis, patches should have approximately the same dimensionality of the dominant pattern in the example texture. The problem of patch dimensionality is equally crucial for example-based

style transfer. In this case, we are given as example an image containing a mixture of style and content. Hence, patch dimensions should be large enough to represent the patterns that characterize the example style, while small enough to forbid the synthesis of content structures present in the example image. We propose a style transfer method that is able to meet these requirements by means of an adaptive patch partition.

Fig. 1 illustrates our method. This paper makes the following contributions:

- We suggest that a correct style transfer can be thought as a local transfer of texture and a global transfer of color. A robust method for local texture transfer must capture the style while preserving the image structure, and this can be achieved with a spatially adaptive image partition.
- We show that a relevant partition must incorporate a prediction of how well an image portion of the source image will be matched to the example style image. That naturally leads to an example-based partition, where the partition is bound to the coupling between the source image and the example image.

## 2. Related Work

Style transfer can be related to texture [7] and color transfer [20, 12]. Texture transfer can be seen as a special case of texture synthesis, where example-based texture generation is constrained by the geometry of an original image. Style transfer, for this part, can be seen as a composition of texture and color transfer, where style is transferred from an example to an original image, being modeled as a combination of texture and color. Recent methods modeling style transfer in a color context include [23], where the style of head shots is mimicked through local image statistics and [24] where the daytime of an image is transformed relying on examples. In this work, we approach style mainly from the textural rather than the color aspect.

Texture synthesis by non-parametric sampling is inspired by the Markov model of natural language [22], where text

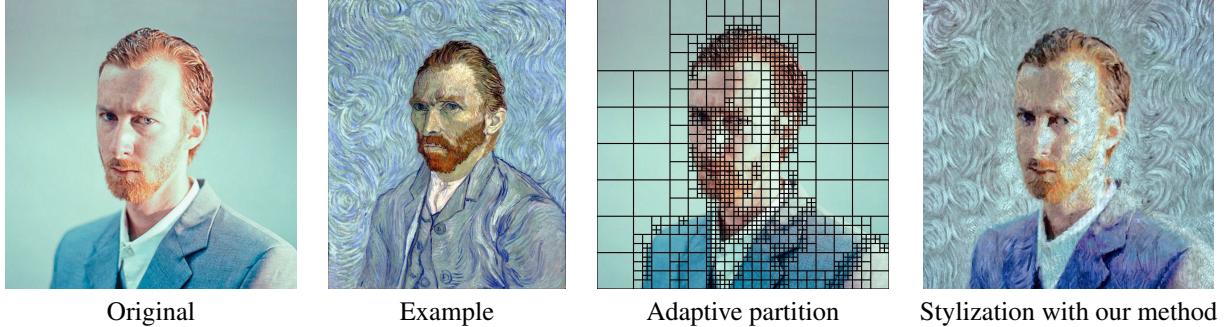


Figure 1. Illustration of the proposed *unsupervised* style transfer. From left to right: the source image, the example image, the adaptive example-based partition, and the result of the style transfer.

generation is posed as sampling from a statistical model of letter sequences ( $n$ -grams) taken from an example text. In an analogous manner, non-parametric texture synthesis relies on sampling pixels directly from an example texture. It became a popular approach for texture synthesis [8] and for texture transfer [7, 15, 27] due to convincing representation of either non-structural and structural textures.

In the literature of texture synthesis and transfer, we find two main approaches to compute non-parametric sampling from an image based Markov Random Field (MRF), which we call here respectively as the *greedy* and the *iterative* strategies. The first strategy considers texture synthesis as the minimization of a greedy heuristic cost function, performing sampling by neighborhood matching to obtain a local solution. The non-parametric texture synthesis method of [8] takes a pixel to be synthesized by random sampling from a pool of candidate pixels selected from an example texture. A similar approach was extended to patch-based texture synthesis and also for texture transfer in [7]. We follow in this work an iterative strategy, inspired by [11] and [25], which considers an explicit probability density modeling of the problem and computes an approximate Maximum a Posteriori (MAP) solution through an iterative optimization based on Loopy Belief Propagation or Graph cuts.

Style transfer can be computed in a *supervised* or *unsupervised* fashion. One of the first methods to propose supervised style transfer posed the problem as computing an “image analogy” given by  $A : A' :: B : B'$  [15], implying that an input image  $B$  should be related to a stylized image  $B'$  the same way as image  $A$  is related to  $A'$ , with  $A$  and  $A'$  known. In this method, inspired by the texture transfer of [1], a pixel to be synthesized in image  $B'$  is directly selected from an example stylized image  $A'$ , by minimizing a cost function that takes into account the similarity between  $B$  and  $A$  and the preservation of local neighborhoods in  $A'$ . The image analogies approach was extended to video in the work of [3], where the problem of temporal coherence is investigated, and recently it was accelerated with hash ta-

bles in [2]. We note that the supervised approach needs a registered pair of example images  $A$  and  $A'$  from which it is possible to learn a style transformation, however this pair of images is hardly available in practice. In this work we rather consider an unsupervised approach.

There are few works dealing with unsupervised style transfer in the literature, the closest to our method being [21], [4] and [27]. Still borrowing from the image analogies notation, we can consider that the *unsupervised* scenario assumes that only an example image  $A'$  and an original image  $B$  are given. In [21] the authors describe a Bayesian technique for inferring the most likely output image from the input image and the exemplar image. The prior on the output image  $P(B')$  is a patch-based MRF obtained from the input image. The authors in [27] decompose original and example images into three additive components: draft, paint and edge. In our approach, the input image is not decomposed into additive parts, as we rather consider a spatial decomposition. Moreover, we note that in both [21] and [27], a MRF is defined for image patches disposed over a regular grid, which is not the case in our approach, where we consider an example-based adaptive image partition.

Finally, the recent work of [13] proposed a style transfer technique using Convolutional Neural Networks (CNN) to separate and recombine the content and the style of two images. Their method differs considerably from our approach, assuming a pre-trained neural network architecture and solving gradient descent for style reconstruction.

### 3. Split and Match Style Transfer

According to the primal sketch theory of visual perception [18], an image may be seen as a composition of *structures*: an ensemble of noticeable primitives or tokens; and *textures*: an ensemble with no distinct primitives in pre-attentive vision. Inspired by this principle, [9] presented a generative model for natural images that operates guided by these two different image components, that they called

as *sketchable* and *non-sketchable* parts.

In this work, we adopt a similar view for example-based style synthesis. Our main motivation comes from the observation that the visual elements accounting for distinctive painting styles in fine arts are often anisotropic with respect to scale. In other words, details corresponding to the geometry (or the sketchable part) of a scene are often painted carefully with fine brushwork, while the scene non-sketchable part is sometimes painted with rougher brushes, where brushwork style is usually more distinct. Obviously, this observation holds more importantly for some particular artistic styles such as impressionism and post-impressionism than other painting styles such as realism.

We remind that in texture transfer, pixel-based models have assumed neighborhoods with regular size, and patch-based methods similarly assume an image decomposition into patches in a regular grid. As we illustrate in Fig. 2, a regular grid assumption is problematic for style transfer. In general, if the patches in a regular grid are small (for instance of size  $8 \times 8$ ), we achieve a realistic reconstruction of the original image, but the style of the example image is hardly noticeable. On the other hand, for larger patch size, the style from the example can be noticed in the reconstructed image, however the fine geometry of the original image is not correctly reconstructed.

In order to overcome this limitation, we propose a method that takes into account the scale problem in stylization. In the following subsections, we give a formal definition for unsupervised style transfer and our proposed solution to the problem.

### 3.1. Problem definition

Let  $u : \Omega_u \rightarrow \mathbb{R}^3$  be an input image and  $v : \Omega_v \rightarrow \mathbb{R}^3$  an example style image. Style transfer can be posed as finding a correspondence map  $\varphi : \Omega_u \rightarrow \Omega_v$ , which assigns to each point  $x \in \Omega_u$  in the original image domain a corresponding point  $\varphi(x) \in \Omega_v$  in the example image domain. The output image can then be defined as  $\hat{u} = v(\varphi)$ .

In order to capture the style of  $v$  while preserving the structures of  $u$ , the correspondence map  $\varphi$  should ideally be a piecewise constant translation map on a partition  $R = \{R_i\}_{i=1}^n$  of  $\Omega_u$ . In practice, the partition  $R$  should depend on the geometrical content of  $u$ , while ensuring the existence of good correspondences between  $u$  and  $v$  over each region  $R_i$ . To achieve a convincing style transfer, regularity is also required at the boundary between neighboring correspondent regions.

All these requirements could be expressed in a unique non convex energy depending on both  $R$  and  $\varphi$  and requiring an alternating optimization strategy. For the sake of simplicity, our approach rather considers these sub-problems independently, following the four steps below:

1. Split and match: compute an adaptive partition  $R$  of

$\Omega_u$  (Sec. 3.2);

2. Optimization: Search for the optimal map  $\varphi$  (Sec. 3.3);
3. Bilinear blending between neighboring regions and reconstruction of  $\hat{u}$  (Sec. 3.4);
4. Global color and contrast matching (Sec. 3.5).

In the split and match step, we split  $\Omega_u$  into a quadtree  $R$  which takes into account both the geometry of  $u$  and the ability for these regions to have good matches in  $v$ . At the same time, we compute for each region  $R_i$  a reduced set of candidate regions in  $v$ . The search of the optimal map  $\varphi$  is then seen as a graph labeling problem, where the nodes of the graph are the regions  $R_i$ . We denote  $L_i = \{l_{i,k}\}_{k=1}^K$  the set of  $K$  candidate labels for the region  $R_i$ , the label  $l_{i,k} \in \Omega_v$  being a patch coordinate in image  $v$ . This probabilistic labeling problem is solved by belief propagation, followed by bilinear blending for the final reconstruction. We note that  $\hat{u}$  is reconstructed by texture transfer only in luminance (Y channel in YUV color space).

Finally, we suggest applying a global color transfer method [12] in chrominance channel to capture the color style, and a contrast transformation to match the global contrast of the example image.

### 3.2. Split and Match adaptive partition

As we claim throughout this paper, decomposing an image into a suitable partition has a considerable impact in the quality of patch-based style synthesis. We propose a simple yet effective approach based on a modified version of the classic Split and Merge decomposition [16]. In the classic algorithm, the local variance of a quadtree cell decides whether a cell will be split into four cells. Here we propose a “Split and Match” example-guided decomposition, where the stopping criteria for quadtree splitting depends also on the patch similarity between the input and example images.

In our representation, a region  $R_i$  is a square of  $\Omega_u$ , of size  $\tau_i \times \tau_i$ . We denote by  $x_i$  its center and we denote indifferently by  $u(R_i)$  or  $p_{x_i}^u$  the patch of size  $\tau_i^2$  centered at  $x_i$ .

The decomposition starts with one single region  $R_1 := \Omega_u$ . Each region  $R_i$  of the partition is split into four equal squares, each one of size  $(\frac{\tau_i}{2})^2$ , until a patch in the example image  $v$  matches  $u(R_i)$  with some degree of accuracy.

Since quadtree patches can have arbitrary size, we use normalized distances for patch comparison. More precisely, the distance between two patches  $p_{x_i}^u$  and  $p_y^v$  of the same size  $\tau_i^2$  is defined as

$$d[p_{x_i}^u, p_y^v] = \frac{\|p_{x_i}^u - p_y^v\|^2}{\tau_i^2}. \quad (1)$$

Now, if  $y_i$  is the best correspondence of  $x_i$  in  $v$  at this scale  $\tau_i$ :

$$y_i := \arg \min_y d[p_{x_i}^u, p_y^v], \quad (2)$$

the region  $R_i$  is split in four regions if the following condition is satisfied

$$\begin{aligned} \zeta(p_{x_i}^u, p_{y_i}^v) = \\ (\sigma_i + d[p_{x_i}^u, p_{y_i}^v] > \omega \text{ and } \tau_i > \Upsilon_0) \text{ or } \tau_i > \Upsilon_1, \end{aligned} \quad (3)$$

where  $\sigma_i = \sqrt{\text{Var}(p_{x_i}^u)}$  is the standard deviation of  $p_{x_i}^u$ ,  $\omega$  is a similarity threshold (fixed to  $\omega := 15$  in practice),  $\Upsilon_0$  is the minimum patch size and  $\Upsilon_1$  the maximum patch size allowed in the quadtree (respectively fixed to  $8^2$  and  $256^2$ ).

Observe that  $R_i$  is not encouraged to be split if there is at least one patch  $p_y^v$  which is similar enough to  $p_{x_i}^u$ , unless the standard deviation of the patch  $\sigma_i$  is large.

Eventually, for every “leaf node” of the quadtree (nodes for which the splitting condition in Eq. (3) is not satisfied), a set of  $K$  candidate labels  $L_i = \{l_{i_k}\}_{k=1}^K$  is selected for  $R_i$  by computing a spatially constrained K-nearest neighbors (k-NN)  $\{p_{l_{i_k}}^v\}_{k=1}^K$  of  $p_{x_i}^u$  in  $v$ . A spatial constraint  $|l_{i_k} - l_{i_{k+1}}| > \chi$  (with  $\chi := \frac{\tau_i}{2}$  in practice), requires that two candidate patch labels  $l_{i_k}$  and  $l_{i_{k+1}}$  are sufficiently distant from each other and encourages label variety. The whole split and match step is summarized in Algorithm 1.

---

#### Algorithm 1 “Split and Match” patch decomposition

---

**Input:** Images:  $u, v$ ; parameters:  $\Upsilon_0, \Upsilon_1, \omega$   
**Output:** Set of regions  $R = \{R_i\}_{i=1}^n$ , set of candidate labels  $L = \{L_i\}_{i=1}^n$

- 1: *Initialization:*  $R_1 \leftarrow \{\Omega_u\}$
- 2: **for** every region  $R_i \in R$  **do**
- 3:    $x_i \leftarrow \text{center of } R_i$
- 4:    $\sigma_i \leftarrow \sqrt{\text{Var}(p_{x_i}^u)}$
- 5:   Compute  $y_i = \arg \min_y d[p_{x_i}^u, p_y^v]$
- 6:   **if**  $\zeta(p_{x_i}^u, p_{y_i}^v)$  is true **then**
- 7:     Split  $R_i$  into four:
- 8:      $m \leftarrow \#R - 1$
- 9:      $R \leftarrow \{R \setminus R_i\} \cup \{R_{m+1}, \dots, R_{m+4}\}$
- 10:   **else**
- 11:     Compute spatially constrained k-NN:
- 12:      $L_i \leftarrow \{l_{i_k}\}_{k=1}^K$  with  $|l_{i_k} - l_{i_{k+1}}| > \chi$
- 13:   **end if**
- 14: **end for**

---

In Fig. 2, we show the interest of adopting an example-based adaptive image partition. Note that for  $8^2$  patch dimensionality, the pointillist texture feature is not captured, while for  $32^2$  patch dimensionality the style is better captured at the cost of having poor reconstruction of structures present in the original image. When a classic adaptive partition is used (based on the variance of the original image),

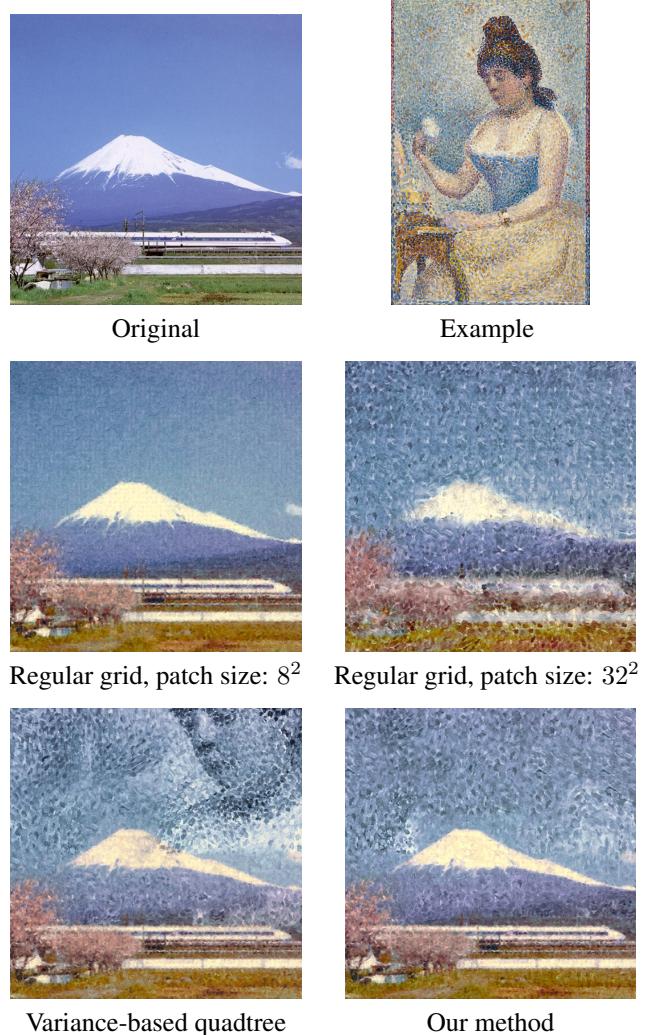


Figure 2. Comparison of partitioning strategies for style transfer. Top row: original and example. Middle row: style transfer using regular patches. Small patches do not manage to capture the style, while large patches do not preserve the image structures. Bottom row: style transfer using adaptive patches. The left adaptive partition based on image variance exhibits an artifact due to a large image portion that cannot be correctly matched. On the contrary, the example-based partition finds good matches for all parts.

style transfer is reasonably achieved, but entire structures are also copied from the example image (the woman’s face in the painting). On the other hand, when the “Split and Match” adaptive partition is used, it leads to a convincing synthesis of the example style, while structures in the original image are well preserved.

### 3.3. Markov Random Fields modeling

Markov Random Fields (MRF) are a well known inference model for computer vision problems [14], widely

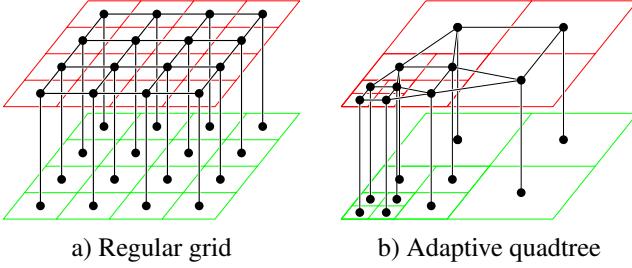


Figure 3. a) MRF for low-level vision problems over a regular grid. Nodes in the bottom layer represent image units from the observed scene, while nodes in the top layer represent hidden image units that we search to estimate through inference. The vertical edges represent data fidelity terms, while the horizontal edges represent pairwise compatibility terms. b) MRF over an adaptive image partition.

used to model texture synthesis [28] and texture transfer [7]. Within this framework, the problem of example-based style transfer can be solved by computing the Maximum a Posteriori from a well chosen joint probability distribution on all image units (quadtree patch labels in our model).

Usually, patch-based MRF models such as in [11] are computed over a graph in a regular grid, as illustrated in Fig. 3a. In this work, we rather propose a MRF model over an adaptive partition, as shown in Fig. 3b. Nevertheless, the neighborhood definition in the proposed quadtree MRF is analogous to a 4-neighborhood in a regular grid.

As discussed in Sec. 3.2, for a quadtree patch  $p_{x_i}^u$ , we first compute a set of  $K$  candidate labels  $L_i = \{l_{i_k}\}_{k=1}^K$  as a strategy to reduce the dimensionality of the labeling problem. We consider now an inference model to compute the most likely set of label assignments for all the patches in  $R$ , where labels represent patch correspondences between  $u$  and  $v$ . More precisely, we search for the set of label assignments  $\hat{L} = \{\hat{l}_i\}_{i=1}^n$  maximizing the probability density

$$P(L) = \frac{1}{Z} \prod_i \phi(l_i) \prod_{(i,j) \in \mathcal{N}} \psi(l_i, l_j), \quad (4)$$

where  $Z$  is a normalization constant,  $\phi$  is the data fidelity term

$$\phi(l_i) = \exp(-d[p_{x_i}^u, p_{l_i}^v] \lambda_d) \quad (5)$$

and  $\psi(l_i, l_j)$  is a pairwise compatibility term between neighboring nodes  $i$  and  $j$  ( $(i, j) \in \mathcal{N}$  means that  $R_i$  and  $R_j$  are neighbors in  $\Omega_u$ )

$$\psi(l_i, l_j) = \exp(-d[\tilde{p}_{l_i}^v, \tilde{p}_{l_j}^v] \lambda_s + |l_i - l_j|^2 \lambda_r), \quad (6)$$

with  $\lambda_d$ ,  $\lambda_s$  and  $\lambda_r$  three positive weights (respectively fixed to 2, 2 and 1 in all experiments). This function  $\psi$  is composed of a smoothness term and a term penalizing label repetitions.

In patch-based MRFs, the compatibility term ensures that neighbor candidate patches are similar in their overlapping region. To define this properly, we first extend each region  $R_i$  of the partition  $R$  by  $\tau_i \theta$  in each direction ( $\theta$  is an overlapping ratio set to 0.5 in practice). This permits to define an overlap between two neighboring extended regions  $\widetilde{R}_i$  and  $\widetilde{R}_j$ . The term  $d[\tilde{p}_{l_i}^v, \tilde{p}_{l_j}^v]$  in  $\psi(l_i, l_j)$  is the distance between the corresponding extended patches in  $v$  over this intersection  $\widetilde{R}_i \cap \widetilde{R}_j$ . While we search for smooth intensity transitions in the overlapping part of neighbor candidate patches, we also aim to penalize two neighbor nodes to have exactly the same label, thus we encourage  $|l_i - l_j|^2$  to be large as a strategy to boost local synthesis variety.

Note that computing an exact MAP inference to solve directly Eq. (4) is an *intractable combinatorial problem* due to the high dimensionality of image based graphical models, but approximate solutions can be found by iterative algorithms. We adopt in this work the *Loopy Belief Propagation* method [26] [19]. Basically, neighboring variables update their likelihoods by *message passing* and usually after a small number of iterations, the approximate marginal probabilities (beliefs) of all the variables in a MRF are computed [11]. In practice, we do not maximize the density (4) but rather minimize its negative logarithm for computational convenience.

Finally, a patch in the reconstructed image  $\hat{u}$  with estimated label  $\hat{l}_i$  is given by  $p_{x_i}^{\hat{u}} = p_{\hat{l}_i}^v$ .

### 3.4. Bilinear blending

Although we compute label correspondences that are likely to be coherent across overlapping regions, seams can still be noted in the reconstructed image  $\hat{u}$  across the quadtree patch boundaries. In order to remove visible seams we apply an effective method inspired on linear alpha blending. Note that in an overlapping quadtree, a variable number of patches may overlap. Then, a pixel  $x$  in the final reconstructed image  $\hat{u}(x)$  is defined as a linear combination of the  $S$  overlapping patch intensities at  $x$ :

$$\hat{u}(x) = \sum_{s=1}^S \alpha_s(x) \tilde{p}_{x_s}^{\hat{u}}(x), \text{ where } \alpha_s(x) = \frac{\delta(x, \partial \tilde{p}_{x_s}^{\hat{u}})}{\sum_{s=1}^S \delta(x, \partial \tilde{p}_{x_s}^{\hat{u}})} \quad (7)$$

is the weighting factor and  $\delta(x, \partial \tilde{p}_{x_s}^{\hat{u}})$  is the normalized closest distance between pixel  $x$  and the patch border  $\partial \tilde{p}_{x_s}^{\hat{u}}$ :

$$\delta(x, \partial \tilde{p}_{x_s}^{\hat{u}}) = \frac{|x - \partial \tilde{p}_{x_s}^{\hat{u}}|^2}{\tau_s^2}. \quad (8)$$

This blending strategy ensures smooth transitions between neighbor patches at a very low computational cost.

### 3.5. Global color and contrast transfer

We have described in the previous subsections our strategy for texture transfer through an adaptive patch-based ap-

proach. Now, we consider that color and contrast are two features in style that may be consistently modeled as global transformations.

That said, we apply the color transfer method proposed in our previous work [12] to match consistently the color palettes of the original and example images. This color transformation is combined with a global contrast transformation achieved by a parametric histogram specification. In particular, classical histogram specification between images may lead to visual artifacts, thus we approximate the histogram specification curve to a power law model through least squares fitting. That ensures that the contrast is transferred globally without creating artifacts.

## 4. Experiments

We present here a number of experiments performed with our method. In Fig. 4, we present a comparison of our algorithm and two state-of-the-art style transfer methods. The first method, called PatchTable [2], is originally applied for supervised style transfer, but since we do not have a pair of example unfiltered and filtered images (as used in [15]), we apply their method<sup>1</sup> in an unsupervised setting by assuming that the unfiltered and filtered images are the same. In their result, Van Gogh's typical brushwork is transferred at some point, but the complete painting style is poorly recreated. Also the overall structures of the original image are lost. The second method is the Neural Artistic Style [13], based on CNN. While the color palette of the example image is well preserved, the texture is not well recreated in their results. For example, the texture in the sky differs considerably from the snail shapes in Van Gogh's painting. Our result outperforms state-of-the-art methods, capturing the local image texture, color and contrast.

In Fig. 5, we present an illustration of style transfer from two different styles of sketches. Both results show the good performance of our algorithm in transferring the drawing style.

Finally, in Fig. 6, we show style transfer with our method for two different painting styles. With our adaptive quadtree partition each original image has a different partition particularly adapted to the given example. Thanks to this strategy, our algorithm accomplishes to transfer the appearance independently of the scale texture of the example or the geometry of the original image. We believe that a good style transfer implies to reproduce the texture and the color palette of the example image, as we have already mentioned. However, other artistic choices are possible depending on the desired results. For example, some application may request to only transfer texture and keep the colors of the original image. Fig. 7 shows our results when transferring or not the global color.

<sup>1</sup>We use the code provided in the author's page

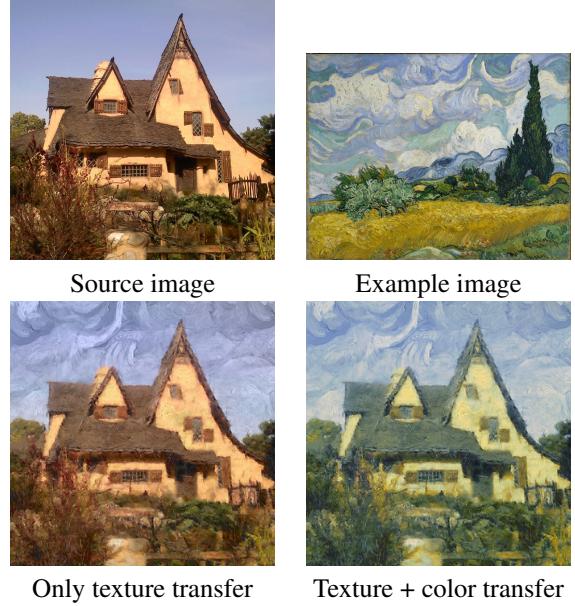


Figure 7. Illustration of only texture versus texture and color transfer. Both results are interesting depending on the artistic choice.

## 5. Conclusion

In this work, we have proposed a new style transfer method that, differently to previous patch-based approaches, is able to synthesize textures independently of their scale. Our results suggest that the decomposition of content and style in artistic images can be achieved with a simple yet efficient adaptive image partition.

Moreover, we have shown that a local texture modeling and a global color transfer strategy leads to a convincing and structure-preserving stylization. On the other hand, state-of-the-art style transfer methods are likely to destroy structures at the cost of synthesizing style.

We note that our method is naturally not guaranteed to transfer textures that belong to the same semantic category in the input and example images. The incorporation of additional semantic constraints or user scribbles could be beneficial to enforce, for instance, that the style in the sky of an example image is mimicked in the sky of the stylized image.

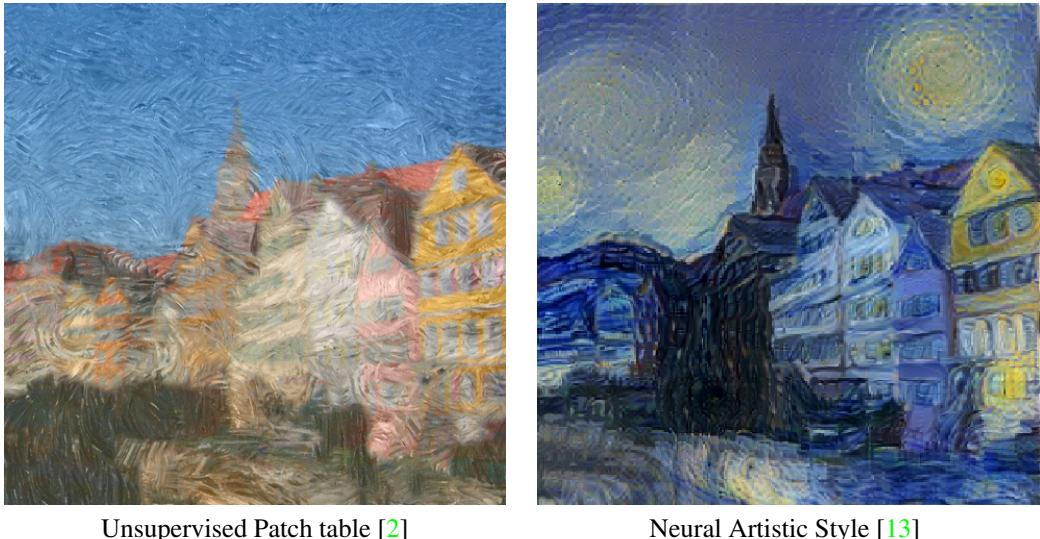
Transferring edge styles is also not guaranteed by our method, since the euclidean norm between image patches is not well adapted to match edges of different thickness and styles. On the other hand, a norm computed in a transformed feature space (for example, the filter responses from the convolutional layers of a CNN) could be more adapted to match edges of different styles.

At the moment, our implementation takes approximately 3 min. for a  $512 \times 512$  image. Future work will concentrate on the acceleration of the patch search, and the adaptation of our method for videos.



Original and Example

Our method



Unsupervised Patch table [2]

Neural Artistic Style [13]

Figure 4. Comparison with state-of-the-art. It can be observed that our method captures the prominent texture and color features from Van Gogh's painting, with an overall accurate reconstruction of buildings. The method of [2] captures partially the brushwork texture from the painting, but the buildings are not well reconstructed. The method of [13] captures accurately the painting colors, however it does not reconstruct all main structures in the original image (buildings in bottom left), and the brushwork textures are not noticeable in the result image, which has a rather blurry effect in the sky.



Figure 5. Illustration of example-based style transfer for sketches. From left to right: original image from Magritte and the result of our algorithm using as example the smaller sketches at the right. In this example texture as well as color are very important to reproduce the style.



Figure 6. Illustration of our example-based style transfer for different painting styles. Example images from van Gogh's and Seurat's are on the top and original images are on the left. Our algorithm transfers successfully the style for different texture scales and it preserves the image geometry of the original images.

## References

- [1] M. Ashikhmin. Synthesizing natural textures. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, pages 217–226, New York, NY, USA, 2001. [2](#)
- [2] C. Barnes, F.-L. Zhang, L. Lou, X. Wu, and S.-M. Hu. Patchtable: Efficient patch queries for large datasets and applications. In *SIGGRAPH*, Aug. 2015. [2](#), [6](#), [7](#)
- [3] P. Bénard, F. Cole, M. Kass, I. Mordatch, J. Hegarty, M. S. Senn, K. Fleischer, D. Pesare, and K. Breen- den. Stylizing animation by example. *ACM TOG*, 32(4):119:1–119:12, July 2013. [2](#)
- [4] L. Cheng, S. Vishwanathan, and X. Zhang. Consistent image analogies using semi-supervised learning. In *CVPR*, 2008. [2](#)
- [5] A. Criminisi, P. Perez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE T-IP*, 13(9):1200–1212, Sept 2004. [1](#)
- [6] F. Durand. An invitation to discuss computer depiction. In *NPAR*, pages 111–124, New York, NY, USA, 2002. [1](#)
- [7] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In *SIGGRAPH*, pages 341–346, New York, NY, USA, 2001. [1](#), [2](#), [5](#)
- [8] A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *ICCV*, pages 1033–, Washington, DC, USA, 1999. [2](#)
- [9] C. en Guo, S. C. Zhu, and Y. N. Wu. Towards a mathematical theory of primal sketch and sketchability. In *ICCV*, 2003. [2](#)
- [10] W. Freeman, T. Jones, and E. Pasztor. Example-based super-resolution. *IEEE Comput. Graph. Appl.*, 22(2):56–65, Mar 2002. [1](#)
- [11] W. Freeman, E. Pasztor, and O. Carmichael. Learning low-level vision. *IJCV*, 40(1):25–47, 2000. [2](#), [5](#)
- [12] O. Frigo, N. Sabater, V. Demoulin, and P. Hellier. Optimal transportation for example-guided color transfer. In *ACCV*, pages 655–670, 2014. [1](#), [3](#), [6](#)
- [13] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015. [2](#), [6](#), [7](#)
- [14] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE T-PAMI*, 6(6):721–741, Nov. 1984. [4](#)
- [15] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. In *SIGGRAPH*, pages 327–340, New York, NY, USA, 2001. [2](#), [6](#)
- [16] S. Horowitz and T. Pavlidis. Picture segmentation by a directed split and merge procedure. In *ICPR*, pages 424–433, 1974. [3](#)
- [17] J. Kyprianidis, J. Collomosse, T. Wang, and T. Isenberg. State of the art: A taxonomy of artistic stylization techniques for images and video. *IEEE TVCG*, 19(5):866–885, May 2013. [1](#)
- [18] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., New York, NY, USA, 1982. [2](#)
- [19] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988. [5](#)
- [20] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Comput. Graph. Appl.*, 21(5):34–41, Sept. 2001. [1](#)
- [21] R. Rosales, K. Achan, and B. J. Frey. Unsupervised image translation. In *ICCV*, pages 472–478, 2003. [2](#)
- [22] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948. [1](#)
- [23] Y. Shih, S. Paris, C. Barnes, W. T. Freeman, and F. Durand. Style transfer for headshot portraits. *ACM TOG*, 33(4):148, 2014. [1](#)
- [24] Y. Shih, S. Paris, F. Durand, and W. T. Freeman. Data-driven hallucination of different times of day from a single outdoor photo. *ACM TOG*, 32(6):200, 2013. [1](#)
- [25] X. Wang and X. Tang. Face photo-sketch synthesis and recognition. *IEEE T-PAMI*, 31(11):1955–1967, 2009. [2](#)
- [26] Y. Weiss. Belief propagation and revision in networks with loops. Technical report, Cambridge, MA, USA, 1997. [5](#)
- [27] W. Zhang, C. Cao, S. Chen, J. Liu, and X. Tang. Style transfer via image component analysis. *IEEE Transactions on Multimedia*, 15(7):1594–1601, 2013. [2](#)
- [28] S. C. Zhu, Y. Wu, and D. Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *IJCV*, 27(2):107–126, Apr. 1998. [5](#)