What was Monet seeing while painting? Translating artworks to photo-realistic images

Matteo Tomei, Lorenzo Baraldi, Marcella Cornia, and Rita Cucchiara

University of Modena and Reggio Emilia, Modena, Italy
{name.surname}@unimore.it

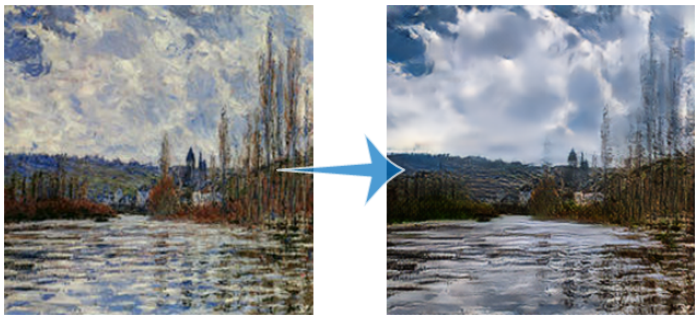


Fig. 1. A sample result from our approach. We propose a method which is capable of generating images with photo-realistic details, preserving the content of an artwork.

Abstract. State of the art Computer Vision techniques exploit the availability of large-scale datasets, most of which consist of images captured from the world as it is. This brings to an incompatibility between such methods and digital data from the artistic domain, on which current techniques under-perform. A possible solution is to reduce the domain shift at the pixel level, thus translating artistic images to realistic copies. In this paper, we present a model capable of translating paintings to photo-realistic images, trained without paired examples. The idea is to enforce a patch level similarity between real and generated images, aiming to reproduce photo-realistic details from a memory bank of real images. This is subsequently adopted in the context of an unpaired image-to-image translation framework, mapping each image from one distribution to a new one belonging to the other distribution. Qualitative and quantitative results are presented on Monet, Cezanne and Van Gogh paintings translation tasks, showing that our approach increases the realism of generated images with respect to the CycleGAN approach.

1 Introduction

In recent years, the Computer Vision community has converged towards unified approaches for image classification and understanding problems. As a matter

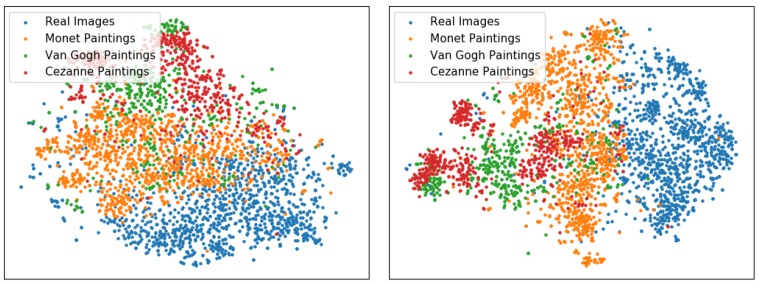
of fact, architectures such as VGG [22] and ResNet [5] are now the standard de-facto for tackling most of the tasks in which an high-level understanding of the image is needed. Nevertheless, the application of state-of-the-art techniques to the domain of Digital Humanities and art is not trivial, as much of the development of the recent years is also due to the availability of large annotated datasets which consist of natural images or videos. This creates strong biases in the trained models, which limit the applicability of current solutions to the artistic domain [1].

A clear visualization of the domain shift between real and artistic data can be obtained by extracting high-level convolutional features from the two domains and visualizing them in a lower-dimensional projection which maintains the structure of the input space, *e.g.* by using a t-SNE transform [15]. In Figure 2, we show the projections of visual features extracted from VGG-19 [22] and ResNet-152 [5] on real and artistic images which roughly describe the same visual domain (in this case, that of landscapes). As it can be observed, even though the content of all distributions is almost identical, features extracted from artistic images are shifted with respect to those extracted from real images, with a distance that increases when selecting less realistic styles, such as those of Cezanne and Van Gogh.

Reducing the domain shift at the pixel level, *i.e.* transforming artistic images to photo-realistic visualizations, is the objective of this paper. The task has been tackled recently in literature as an instance of a more general domain translation task in unpaired settings [29]; we are not aware, however, of other works which have specifically tackled the translation between art and real. Here, our main source of intuition is that high model capacity is mandatory to memorize the details needed to perform photo-realistic generation. Therefore, instead of delegating the task of learning photo-realistic details exclusively to the min-max game of a generative adversarial model, we empower our model with an external memory of real images, and a search strategy to retrieve elements from the memory when needed to condition the generation.

Our model builds upon a Cycle-GAN [29], which consists of two Generative Adversarial Networks to align two unpaired domains. We extend and improve this approach by building external memory banks of real patches, and conditioning the learning to maximize the similarity of generated patches with respect to real patches. To this end, we devise a differentiable association strategy which, given a generated patch, retrieves the most similar real patch in the external memory. An additional loss term is then used to reduce the distance between generated and real patches. The same strategy is applied at multiple scales, to remove possible artifacts in the generation and increase the quality of the final results. To reduce the computational complexity of the approach, we also build an efficient version of our objective which is coupled with an approximated k-NN search.

Beyond presenting quantitative results obtained using state of the art metrics for image generation [6], we also perform careful perceptual experiments conducting a user study to compare the proposed approach to common unpaired



(a) VGG-19 (b) ResNet-152

Fig. 2. Domain shift visualization between real images and artistic paintings of different artists. Visualization is obtained by extracting visual features from both real and artistic images and by running the t-SNE algorithm on top of that. To encode images, we extract visual features from layer *fc7* of the VGG-19 [22] and from the final average pooling layer of the ResNet-152 [5]. To ensure a fair comparison, images are taken from roughly the same distribution of paintings: both represent landscapes. Best seen in color.

translation models, under different settings. The experiments indicate that the images synthesized by our model are more realistic than those generated by a simple image-to-image translation approach.

2 Related work

The literature for image-to-image translation can be roughly categorized into style transfer [2] approaches and methods based on Generative Adversarial Networks (GANs) [4]. In the first case, the rationale is to synthesize a novel image by combining features of one image with features of another image, extracted at different semantic levels [2, 3, 9, 25]. In the seminal work of Gatys *et al.* [2], a realistic input image was modified by minimizing a cost function aiming to preserve the content of the original image, and the style of a target artistic image, encoded via the Gram matrix of activations of a lower CNN layer.

Johnson *et al.* [9], on the same line, proposed the use of perceptual loss functions for training feed-forward encoders for the style-transfer task. Their method showed very similar qualitative results with respect to previous approaches, and significantly reduced the computational cost. The same problem of improving the overall computational efficiency of [2, 3] was addressed in [25], in which a compact feed-forward network was designed to transfer the style of a given image to another one using complex and expressive loss functions. While these approaches have been successful in transferring the global texture properties of artworks to realistic images, mimicking the appearance of the brush strokes, they are not well suited for transferring from the artistic to the real domain, as texture properties are generally encoded in a translation invariant manner, and generating photo-realistic details by inverting CNN activations remains difficult.

On a different note, GANs [4] generate realistic images by aligning the distributions of real and generated images. They have been adopted for conditional image generation problems such as text to image synthesis [19], image inpainting [18] and future frame prediction [16], and have been successfully applied to other domains like videos [27] and 3D data [28].

A large set of existing methods exploit GANs to translate an image to a different representation of the image itself, as for example generating photographs from sketches [21]. In this context, Isola *et al.* [7] proposed a conditional GAN for learning a mapping from input to output images demonstrating the applicability of their network to a wide variety of image-to-image translation tasks. The main drawback of this kind of approaches is the need of paired training data (*i.e.* paired images before and after the translation). To overcome this problem, several methods have addressed the unpaired setting, where the goal is to translate images from a domain to another without leveraging on paired data to learn the corresponding translation. In particular, Liu *et al.* [14] introduced a coupled generative adversarial network that, thanks to a weight sharing strategy, is able to learn a joint distribution of multi-domain images. An extension of this work for unpaired image-to-image translation was presented in [13], exploiting a combination of variational autoencoders and GANs.

Zhu *et al.* [29] instead proposed CycleGAN, a model based on generative adversarial networks that, given two unpaired image collections, automatically translates an image from one domain to the other and vice versa. This is achieved by forcing the translation to be cycle consistent in the sense that if an image is translated from a domain to another, and translated back to the original domain, the result should be consistent with the original image. This cycle consistency criterion has been demonstrated to be effective for several tasks where paired training data does not exist, including style transfer, object transfiguration, season transfer, and photo enhancement.

3 Proposed approach

Given an input painting, our goal is to generate a photo-realistic image representing the same content, without leveraging paired training data. In contrast to style transfer approaches [2], the objective is not to transfer a specific artistic style to an image, but rather to remove any artistic style from the painting, bringing the content back to a photo-realistic visualization. In other words, our model tries to show what reality the artist was observing or imagining while drawing.

The model is built on a cycle-consistent framework [29], which is endowed with an external memory of photo-realistic images and a patch-level retrieval strategy. At training time, real patches can be retrieved at multiple scales thanks to an assignment loss between real and generated patches. A summary of the approach is presented in Fig. 3.

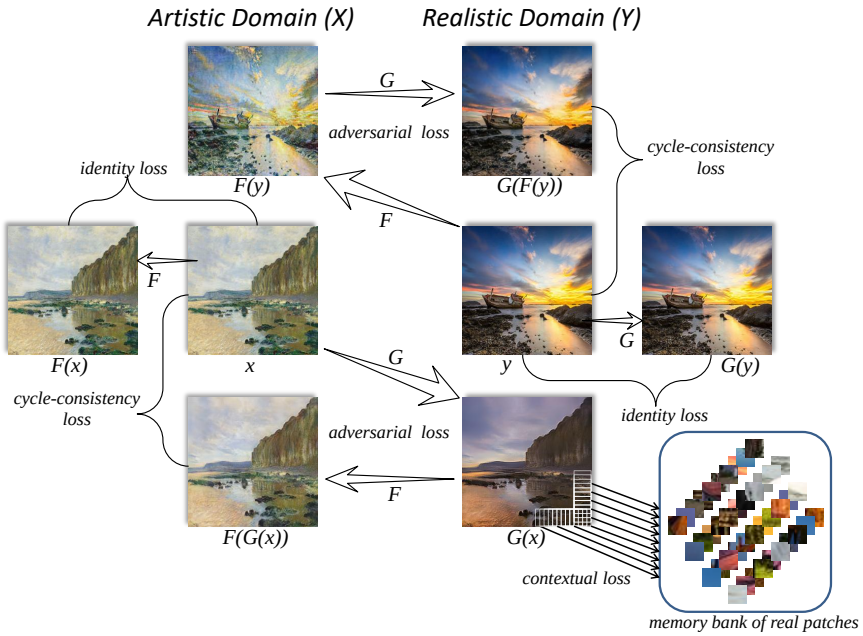


Fig. 3. Overall representation of our model. The model contains two generators (G and F) and two discriminators (not shown in the figure). The adversarial losses [4], combined with cycle-consistency losses, push the generators to produce images belonging to their corresponding target distributions, while imposing a pairing between the two domains. Every generated patch is also associated with respect to a memory bank of real patches, in a multi-scale and differentiable way. An additional cost term minimizes the distance between generated and real patches retrieved from the memory.

3.1 Unpaired image to image translation

Our model needs to learn a mapping between the domain of paintings from a specific artist, which we call X, and the domain of real images, Y. Denoting the data distributions as $x \sim p_{data}(x)$ and $y \sim p_{data}(y)$, two mapping functions are built to translate data from one domain to another, $G : X \rightarrow Y$ and $F : Y \rightarrow X$. Following the Cycle-GAN approach, we realize the two mapping functions through learnable generators, which are paired with two discriminators D_X and D_Y at training time. The Cycle-Consistent Adversarial Objective features the following losses:

– Two *adversarial losses* [4] to generate images indistinguishable from those in the target domain. For both (G, D_Y) and (F, D_X), the generator is trained to reproduce the target data distribution, creating images that are difficult for the discriminator to distinguish from the real ones, while the discriminator is trained to differentiate between real and synthetic images. In this setting, the generator and the discriminator play a two-player minimax game through

the following objective functions:

$$\mathcal{L}_{GAN}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{data}(y)}[logD_Y(y)] + \mathbb{E}_{x \sim p_{data}(x)}[log(1 - D_Y(G(x)))] \quad (1)$$

$$\mathcal{L}_{GAN}(F, D_X, Y, X) = \mathbb{E}_{x \sim p_{data}(x)}[logD_X(x)] + \mathbb{E}_{y \sim p_{data}(y)}[log(1 - D_X(F(y)))] \quad (2)$$

During training, while the generator (*i.e.* the mapping function) tries to minimize the objective, the discriminator tries to maximize it.

– Observing that the adversarial losses, alone, would lead to an under constrained problem, which would not ensure that the input and the generated images share the same content, a *cycle consistency loss* [29] is applied to reduce the space of possible mapping functions. For this purpose, whenever an image is synthesized by a generator, the result is transformed again by the other generator, taking the image back into the starting distribution, thus obtaining a reconstructed image.
We require the original image and the reconstructed one to be the same, *i.e.* $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$ and $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$. This is imposed by applying an ℓ_1-loss between reconstructed and original images:

$$\mathcal{L}_{cyc}(G, F) = \mathbb{E}_{x \sim p_{data}(x)}[\|F(G(x)) - x\|] + \mathbb{E}_{y \sim p_{data}(y)}[\|G(F(y)) - y\|] \quad (3)$$

– An *identity mapping loss* [24] helps to preserve the color distributions between input and output images. This is done by forcing the generators to behave like an identity function when their input are images from their target domain, through the following loss:

$$\mathcal{L}_{id}(G, F) = \mathbb{E}_{y \sim p_{data}(y)}[\|G(y) - y\|] + \mathbb{E}_{x \sim p_{data}(x)}[\|F(x) - x\|] \quad (4)$$

The full Cycle-Consistent Adversarial Loss [29] can therefore be written as follows:

$$\begin{aligned}\mathcal{L}_{cca}(G, F, D_X, D_Y) = \mathcal{L}_{GAN}(G, D_Y, X, Y) + \mathcal{L}_{GAN}(F, D_X, Y, X) \\ + \mathcal{L}_{cyc}(G, F) + \mathcal{L}_{id}(G, F).\end{aligned} \quad (5)$$

Alone, this objective sets an unpaired image-to-image translation setting between artistic and real images, suitable for performing translations in both directions. In the following, we discuss the incorporation of an external memory and of a retrieval strategy to condition the model on real image elements.

3.2 Retrieving real patches from an external memory

Our goal is to translate images from the artistic domain to the realistic domain. To do so, we rely on the hypothesis that realistic images can be effectively generated by copying visual elements from real images, instead of optimizing the cycle-consistent generative objectives alone.

Given a set of real images, we build a memory bank M by extracting fixed-size patches from each image in a sliding window manner. When the number

of real images is sufficiently large and their content is sufficiently aligned with the distribution represented by the paintings, the memory bank can effectively model a distribution of real patches which can drive the training of the generative model. To do so, each generated image $G(x)$ is split into patches as well, following the same patch size and stride as in the memory bank. Then, a retrieval strategy is designed to pair each generated patch with the most similar in the memory bank M, while an assignment loss is in charge of maximizing the similarity between generated and real patches, under the previously computed assignment. Since we focus on appearance, each patch is encoded with its RGB values, thus obtaining a dimensionality of $l \times l \times 3$ for a patch size of $l \times l$.

Reading from the external memory Given the set of real patches from M, $M = \{m_j\}$ and the set of patches from $G(x)$, $K = \{k_i\}$, a pairwise cosine distance function is defined after centering the distributions of real and generated patches with respect to the mean of real patches.

$$d_{ij} = \left(1 - \frac{(k_i - \mu_m) \cdot (m_j - \mu_m)}{\|k_i - \mu_m\|_2 \, \|m_j - \mu_m\|_2}\right), \text{ where } \mu_m = \frac{1}{N} \sum_j m_j \tag{6}$$

Each generated patch is then assigned to its most similar counterpart in the memory bank with a differential assignment strategy. In particular, we first normalize the pairwise distances and then compute pairwise affinities $A_{ij} \in [0, 1]$ as follows:

$$\tilde{d}_{ij} = \frac{d_{ij}}{\min_l d_{il} + \epsilon}, \text{ where } \epsilon = 1e - 5 \tag{7}$$

$$A_{ij} = \frac{\exp(1 - \tilde{d}_{ij}/h)}{\sum_l \exp(1 - \tilde{d}_{il}/h)} = \begin{cases} \approx 1 & \text{if } \tilde{d}_{ij} \ll \tilde{d}_{il} \; \forall \; l \neq j \\ \approx 0 & \text{otherwise} \end{cases} \tag{8}$$

where $h > 0$ is a bandwidth parameter. In practice, each generated patch k_i is softly assigned to the most similar real patch, as determined by the affinity matrix A_{ij}. In other words, k_i will be assigned prominently to the real patch corresponding to $\max_j A_{ij}$, and to others near real patches which happen to have an high degree of affinity with k_i.

Reducing the computational overhead Computing the assignments between real and generated patches requires to compute the entire affinity matrix A_{ij}, which leads to an intractable process when the number of real patches is large. The size of A_{ij}, in fact, grows linearly with the number of patches, which grows linearly with the number of images and quadratically when decreasing the stride.

To reduce the computational overhead, we build a suboptimal Nearest Neighbors index with real patches. Then, for each generated patch k_i, we conduct a k-NN search to get the k nearest samples from the memory bank. We subsequently estimate the affinity matrix A_{ij} by reducing the computation only on the retrieved real patches, thus getting a sparse matrix in which the affinities for non retrieved patches are set to zero. Notice that, when the results of the k-NN

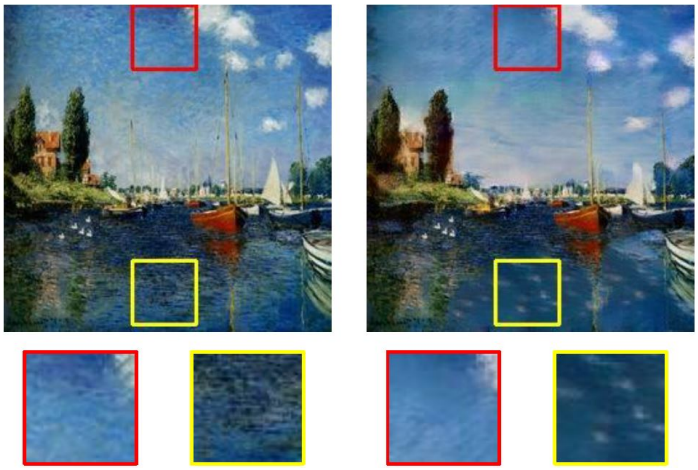


Fig. 4. Comparison between (left) original Monet painting and (right) an image generated by minimizing the contextual loss on real patches, plus a content loss regularization, updating pixel values directly. As shown in the zoomed patches, many brush strokes disappear, recovering realistic textures.

search are reliable, the estimation of A_{ij} is close to the exact results, thanks to the Softmax normalization in Eq. 8.

To speed up the computation of the distances when the number of real images used to generate the memory bank is large, we adopt a suboptimal inverted index with exact post-verification (IndexIVFFlat), which has an high-performance implementation in the Faiss library [8].

Maximizing the similarity with real patches Once affinities are computed, we maximize the similarity between each generated patch and its corresponding assignments from the memory bank. To this aim, we employ the contextual loss [17] as follows.

$$\mathcal{L}_{CX}(K, M) = -\log\left(\frac{1}{N}\left(\sum_i \max_j A_{ij}\right)\right) \tag{9}$$

With the contextual loss, we aim to reduce the distance between two distributions: the distribution of the generated image features (*i.e.*, that of generated patches) and that of the memory bank features (*i.e.*, real patches).

Multi-scale To better translate artistic details to real details we adopt a multi-scale variant of the proposed approach, thus building multiple memory banks, each with different patch sizes and strides. During training, we compute the contextual loss for each scale separately, and define the final objective as the sum of the losses obtained at each scale. In practice, we adopt three scales, as

follows:

$$\mathcal{L}_{CXMS}(K, M) = \sum_{s=1}^{3} -\log\left(\frac{1}{N_s}\left(\sum_i \max_j A_{ij}^s\right)\right) \tag{10}$$

where A^s is the affinity matrix computed between patches with scale s and N_s is the number of real patches of the memory bank for scale s.

The role of the Contextual loss For the ease of the reader, we showcase the benefit of this strategy in a simpler setting which does not employ a min-max generative game. Taking inspiration from style-transfer works [2], we build a cost function and minimize it by back-propagating directly on the pixel values of the source image. In particular, a content loss is placed to regularize the training and to maintain the semantic content of the original image, while the contextual loss is applied to maximize the patch-level similarity with respect to a memory bank of real images. A sample result on a Monet painting is shown in Figure 4. As it can be observed, the contextual loss on real patches helps to obtain realistic and plausible results, removing stroke textures in large portions of the image.

3.3 Full Objective

We combine together the unpaired image-to-image translation framework and our retrieval and assignment strategy between real and generated patches, thus obtaining the following overall training loss:

$$\mathcal{L}(G, F, D_X, D_Y, K, M) = \mathcal{L}_{cca}(G, F, D_X, D_Y) + \lambda \mathcal{L}_{CXMS}(K, M) \tag{11}$$

where λ is the contextual loss weight. During our preliminary tests, we found that good values for λ, in the multi-scale version of the approach, lie around 0.1 or less. As a final side note, it is important to underline that we are interested only in generating real images from paintings and not also in the opposite task. For this reason, we do not include a second set of memory banks for the artistic features, and we do not compute the contextual loss also in the opposite direction.

4 Experimental results

In this section, we provide qualitative and quantitative results of the proposed solution as well as implementation details and datasets used in our experiments.

4.1 Datasets and Implementation Details

To evaluate our approach, we use a set of paintings from Monet, Cezanne and Van Gogh and a set of real images. To keep the distribution of paintings and real images roughly aligned, real images are selected from landscape pictures: paintings are downloaded from Wikiart.org, and photos are taken from Flickr using the combination of tags `landscape` and `landscapephotography`. Black-and-white photos were pruned, and the images were scaled to 256×256 pixels.

The number of samples for each training set are Monet: 1072, Cezanne: 583, Van Gogh: 400, Photographs: 2048.

CycleGAN Parameters To build the Cycle-GAN part of our model, we keep the same networks and training parameters as in [29]. The generative networks architecture is adapted from Johnson *et al.* [9] and contains two stride-2 convolutions to downsample the input, followed by several residual blocks and then two convolutional layers with stride 1/2 for upsampling. The discriminative networks are PatchGANs [7, 11, 12] which try to classify if each square patch in an image is real or fake.

Contextual loss Parameters We extract patches at three different scales, to fill our memory banks, from 100 different real images. Keeping the size of the image constant, we extracted real patches of size 4×4, 8×8 and 16×16, using stride values of 4, 5 and 6 respectively. During training, at each iteration we extract patches with the same sizes and strides from the generated image and compute the contextual loss. The contextual loss weight λ in equation 11 was fixed to 0.1.

We train the model for 200 epochs by using the Adam optimizer [10] with a batch size of 1, keeping a learning rate of 0.0002 for the first 100 epochs and then linearly decaying it to zero over the next 100 epochs. Weights are initialized from a Gaussian distribution with 0 mean and standard deviation 0.02.

4.2 Qualitative Results

We compare our results with those from a CycleGAN [29], trained exactly with the same parameters used for our model. Given the subjective nature of the task, before presenting a quantitative discussion, we show some examples of generated images starting from Monet, Cezanne and Van Gogh paintings in Fig. 5.

We observe that our results generally preserve the colors of the original paintings and contain less artifacts than images generated by CycleGAN. This quality improvement is particularly manifest in the details of sky and sea (Fig. 5, first and fourth rows), in the preservation of colors (Fig. 5, third row), and in the smoothness of objects which do not have well defined edges in the original painting, as in the smoke of Fig. 5, second row.

4.3 Quantitative Results

To numerically evaluate the visual quality of the results, we adopt the Fréchet Inception Distance (FID) [6], which has been recently emerging as a reliable metric for evaluating generated images and has been proven to be more consistent with human judgments than the Inception score [20]. FID corresponds to a Wasserstein-2 distance [26] between two multivariate Gaussian distributions fitted on real and generated data, using activations from layers of the Inception-v3 model [23].

In Table 1, we show FID values obtained under different settings. In particular, we measure the FID distance with real images using the original paintings, fake paintings generated using style transfer, and the recovered real images

generated with CycleGAN and our approach. Also, we employ three different Inception-v3 layers to assess the distance using both low-level and high-level visual features. As it can be observed, our model is able to further reduce the distance with real images, when compared to CycleGAN, thus confirming the effectiveness of the approach. The same trend is observable with both low-level and high-level Inception features, and for all the artists.

While being a well-grounded metric for image generation, the FID score cannot be as effective as human judgment. Therefore, we further evaluate our results by conducting a user study. All the tests have involved five volunteering people who were not aware of the details of the proposed approach, and thus not trained to distinguish between our results and those of CycleGAN. In each test, evaluators were presented with different real and generated images, and asked to click on the most realistic one using a web interface. Our tests were structured as follows.

Realism of the generation – users were presented our result and the CycleGAN output for a given input painting, which was not shown in the interface. Generated images were presented in their full size (256×256) and chosen randomly from the dataset. Each user was given 100 image pairs, and asked to select which of the two images seemed more realistic.

Coherency with the source painting – in this test, the interface also showed the original painting to the user, who was asked to click on the generated image that best represented the painting. With this test, we aim to investigate whether our results are more faithful to the original painting colors and composition. Again, 100 image triplets were shown to each user.

Multi-scale comparison with real images – to assess to what extent the generated images look realistic, we also asked the users to rank the realism of the generated images with respect to real images. In this test, the interface showed two images to the user, one generated by our method and the other randomly extracted from the real images dataset. The user was asked to select the more realistic one, and presented with 100 image pairs. The same test was repeated in three different runs, in which images were resized with a ratio of 1, 2/3, 1/2, and ensuring that different real-generated pairs were presented to the same user in different runs.

Tables 2 and 3 show the results of our tests. As it can be observed from Table 2, images generated by our method were evaluated as more realistic than those of CycleGAN 58.4% of the times, thus beating the baseline with a margin of 17%. Also, when showing the input painting to the user, images generated by our method were ranked as more coherent with the input painting, thus underlying that our method is able to preserve color and texture from the painting.

Finally, we also had some chances to win the comparison with real images. As reported in Table 3, even when comparing the results of our generation with real landscapes, sometimes the user was fooled and selected the generated image. As it can be expected, this behavior becomes more frequent when the images are downsized to a small scale. Nevertheless, it is significant to observe that, even at full scale, the user was fooled about 5% of the times.

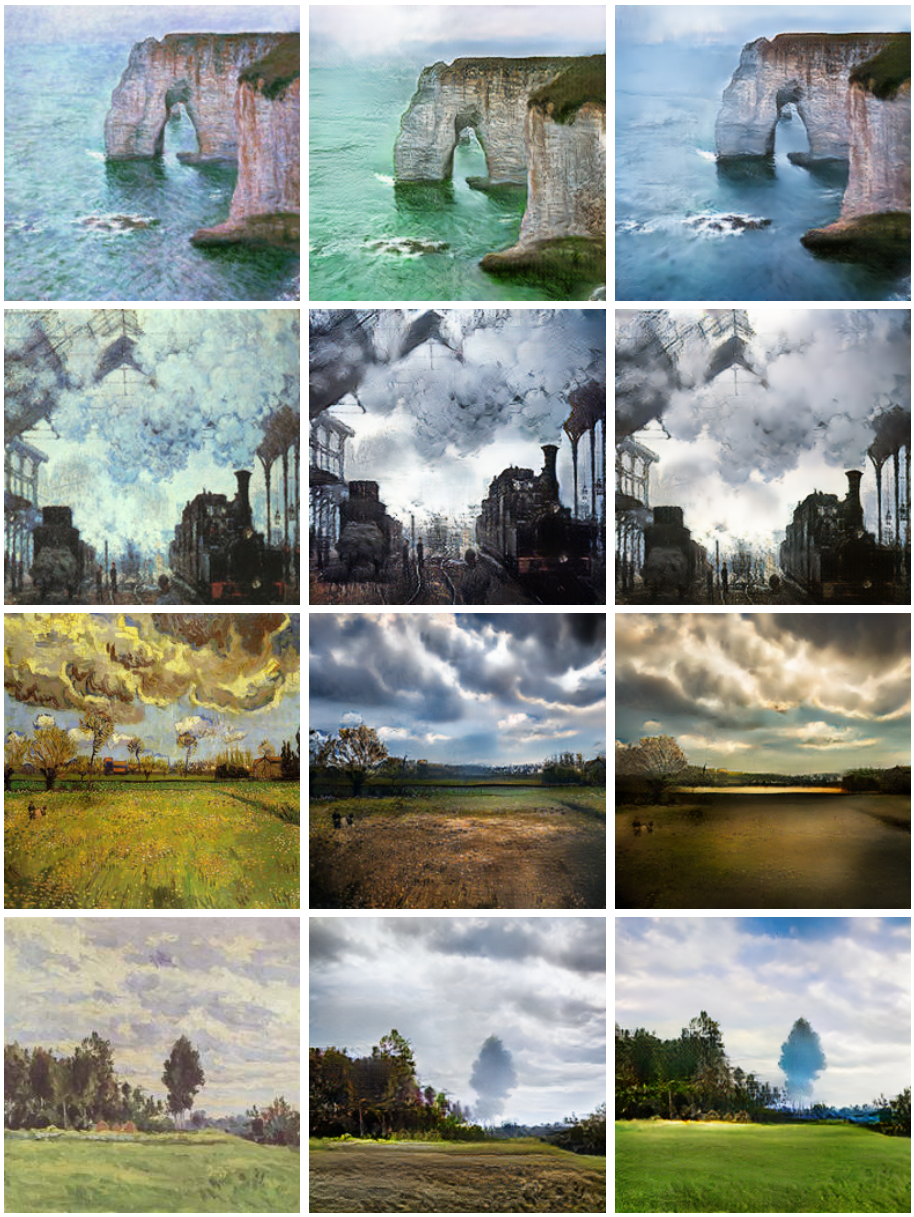


Fig. 5. Results of applying our method to Monet (first and second rows), Van Gogh (third row) and Cezanne (fourth row) paintings. (left) Original painting, (center) CycleGAN [29] output, (right) our method output.

Table 1. Fréchet Inception Distance (FID) [6] computed between real images (landscape pictures) and different sets: artist's original paintings, images obtained transferring the artist's style to the real images through Gatys *et al.* [2], images generated with CycleGAN [29] and with our method. The FID is computed using different feature layers of Inception-v3: the second max pooling (192-d), the pre-auxiliary classifier layer (768-d) and the final average pooling layer (2048-d). FIDs computed at different Inception-v3 layers are not directly comparable [6].

	Monet	**Cezanne**	**Van Gogh**
2048 dimensions			
Original paintings	74.45	176.51	166.72
Style Transfer [2]	58.02	91.23	101.54
CycleGAN [29]	55.26	83.62	86.82
Our Model	**54.43**	**77.01**	**81.74**
768 dimensions			
Original paintings	0.52	1.26	1.39
Style Transfer [2]	0.50	1.01	1.18
CycleGAN [29]	0.41	0.49	0.48
Our Model	**0.34**	**0.37**	**0.41**
192 dimensions			
Original paintings	0.94	1.67	3.96
Style Transfer [2]	0.71	1.49	3.33
CycleGAN [29]	0.31	0.28	0.19
Our Model	**0.16**	**0.13**	**0.11**

Table 2. Results of the user tests on realism and coherency. Values are reported as the percentage of images chosen with respect to the total.

Test	**Scale**	**CycleGAN [29]**	**Our method**
Realism of the generation	256×256	41.6%	58.4%
Coherency with the painting	256×256	41.2%	58.8%

5 Conclusions

We presented a novel method for artistic-to-realistic domain translation. Since paired training data is not available for this task, our approach is based on an unpaired framework. In particular, we built upon the CycleGAN architecture, and enriched it with multi-scale memory banks of real images, to drive the generation at the patch level. To make the approach computationally feasible, we also provided an approximated version of the association strategy. Results, presented both qualitatively and quantitatively, show that our method outperforms the CycleGAN baseline, leading to more realistic results. Despite the increased quality, failure cases are still frequent, and the task is still far from being solved. In particular, we noticed that the method often fails to translate portraits and images with blurry foreground objects. Future works will explore this direction, also tackling the generation of higher resolution images.

Table 3. Results of the multi-scale comparison with real images. Values are reported as the percentage of images chosen with respect to the total.

Scale	Random real image	Generated image
256×256	95.1%	4.9%
170×170	88.2%	11.8%
128×128	88.0%	12.0%

Acknowledgments

This work was supported by the CultMedia project (CTN02_00015_9852246), co-founded by the Italian MIUR. We also acknowledge the support of Facebook AI Research with the donation of the GPUs used for this research.

Fig. 6. Sample results generated by our method.

References

1. Baraldi, L., Cornia, M., Grana, C., Cucchiara, R.: Aligning text and document illustrations: towards visually explainable digital humanities. In: International Conference on Pattern Recognition (2018)
2. Gatys, L.A., Ecker, A.S., Bethge, M.: A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576 (2015)
3. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: IEEE International Conference on Computer Vision and Pattern Recognition (2016)
4. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems (2014)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE International Conference on Computer Vision and Pattern Recognition (2016)
6. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Klambauer, G., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a Nash equilibrium. Advances in Neural Information Processing Systems (2017)
7. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: IEEE International Conference on Computer Vision and Pattern Recognition (2017)
8. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with gpus. arXiv preprint arXiv:1702.08734 (2017)
9. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision (2016)
10. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
11. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE International Conference on Computer Vision and Pattern Recognition (2017)
12. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: European Conference on Computer Vision (2016)
13. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Advances in Neural Information Processing Systems (2017)
14. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in Neural Information Processing Systems (2016)
15. Maaten, L.v.d., Hinton, G.: Visualizing data using t-SNE. Journal of Machine Learning Research **9**(Nov), 2579–2605 (2008)
16. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. In: International Conference on Learning Representations (2016)
17. Mechrez, R., Talmi, I., Shama, F., Zelnik-Manor, L.: Learning to maintain natural image statistics. arXiv preprint arXiv:1803.04626 (2018)
18. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: European Conference on Computer Vision (2016)
19. Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. In: International Conference on Machine Learning (2016)

20. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Advances in Neural Information Processing Systems (2016)
21. Sangkloy, P., Lu, J., Fang, C., Yu, F., Hays, J.: Scribbler: Controlling deep image synthesis with sketch and color. In: IEEE International Conference on Computer Vision and Pattern Recognition (2017)
22. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
23. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: IEEE International Conference on Computer Vision and Pattern Recognition (2016)
24. Taigman, Y., Polyak, A., Wolf, L.: Unsupervised cross-domain image generation. In: International Conference on Learning Representations (2017)
25. Ulyanov, D., Lebedev, V., Vedaldi, A., Lempitsky, V.S.: Texture Networks: Feed-forward Synthesis of Textures and Stylized Images. In: International Conference on Machine Learning (2016)
26. Vaserstein, L.N.: Markov processes over denumerable products of spaces, describing large systems of automata. Problemy Peredachi Informatsii **5**(3), 64–72 (1969)
27. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: Advances in Neural Information Processing Systems (2016)
28. Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J.: Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: Advances in Neural Information Processing Systems (2016)
29. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: IEEE International Conference on Computer Vision (2017)