Deep Learning to paint like Van Gogh

Author: Stefano D'Angelo,
Supervisor: Frédéric Precioso

[1] Université Côte d'Azur, Nice, France (https://univ-cotedazur.fr)
[2] Inria, Sophia Antipolis (Nice), France (https://www.inria.fr/fr)

Abstract. *The objective of this paper is to transfer the artistic style from artworks, such as paintings, to real images to transform them into new artworks. The principle is to start from real photos and apply the style of Van Gogh to transform the initial photo into a painting-like picture. In this way, new data can be generated when real data is missing, and the dataset containing all of Van Gogh's paintings can be augmented. There have been several techniques proposed to transfer artistic style, from Convolutional Neural Networks to Generative Adversarial Networks. However, most of them do not take into account class information related to the objects present in both images. Hence, this paper tries to improve the state-of-the-art results, comparing different approaches involving semantic segmentation.*

Keywords: Neural Style Transfer · Image Segmentation · Image-to-Image Translation.

1 Introduction

When Vincent Van Gogh painted *Starry Night*, he saw none other than a beautiful landscape. Then, using wispy brush strokes and a specific palette, he captured his *impression* of the scene on the painting. Unfortunately, Van Gogh is not with us anymore, and his original style is gone with him.
However, Deep Learning has made big steps in the area of style transfer, and nowadays we have the more disparate methods to apply the style of a painter to a real picture. The problem with most of these methods is that they ignore instance information in an image. For example, if a landscape image contains both the sky and trees, it is very likely that Van Gogh, or a painter in general, used different styles to paint these elements. Without considering this problem, we are simply assuming that the style is uniform everywhere, which is evidently not the case when we look at whatever Van Gogh's painting.

In this paper, we indeed explore three methods to transfer the style from a painting to a real photo, taking into account the semantics of both the images. We first start from a paired dataset, containing couples of samples coming from two different domains, but that in some way share the same content. These domains are respectively the set of Van Gogh's paintings and some real pictures. To extract the semantics of the images the approach followed was the one coined

by Penhouët et al. [1], that, differently from other methods having the goal of segmenting everything [2], uses *image segmentation* and *semantic grouping* to merge minority classes in order for the masks of each pair of images to match.
Since all the pre-trained segmentation models have been trained on real pictures, it is hard to directly segment paintings. For this reason, in this paper we investigate an approach that converts the paintings to real images and then computes segmentation masks. Then, three different pipelines have been tested.
In the first one, all the masks and the corresponding images are cropped. Then, for each patch of the real photo, we retrieve the patch of the painting having the most similar semantic content. Based on this choice, we transfer the style from the corresponding patch of the painting to the patch of the photo. In the end, we recompose the whole image.
The second pipeline is based on Champandard's model [3], where we transfer the style re-arranging the instances of the objects in the style image according to the semantic mask of the content image.
In the third and last approach, the style is transferred in the real photos domain, and the resulting image is then converted back to the paintings domain.

The main goal is hence to generate new data when real data is missing, so that the dataset containing all Van Gogh's paintings can be augmented. For this reason, the results have been evaluated according to how much they look similar to Van Gogh's artworks, or, in other words, how "fake" they are.
All the models tested in this paper can be found in **this** GitHub repository.

2 Neural Style Transfer

Neural Style Transfer is an area of application of Image-to-Image translation, whose goal is to transfer the style of an image, called *style image*, to another image, called *content image*.
This field has been widely explored, and lots of methods have been proposed. The first state-of-the-art model has been developed in 2015 by Gatys et al. [4], and consisted of a VGG-19 network re-adapted to the problem. Then, a further improvement was to combine the CNN architecture with a Markov Random Field [5]. The latter is a regularizer that maintains local patterns of the "style" exemplars.
These models were still struggling to correctly transfer the style, often resulting in overlapping between content and style images.

A branch of architectures for style transfer that brought some improvements is the one that involves the usage of unpaired datasets, which constitutes a point of advantage with respect to the previous architectures since it is always laborious to pair samples. This kind of dataset contains samples coming from two different domains. In this branch, the predominant models are Generative Adversarial Networks (or GANs), whose properties have been exploited to generate fake images mapping together two different domains. This is the case of GANilla [6], an architecture consisting of a CNN containing a Resnet18 as the downsampling network, and CycleGAN's upsampling layers as the upsampling network.

CycleGAN [7] is another of these powerful models, and its novelty consisted in the bidirectional and cyclic translation of samples from one domain to another domain. It outperformed other similar networks such as BiGAN [8].
The problem with these methods is that they are not semantic-aware. In practice, they do not make (or make poor) distinctions among the objects present in an image. Furthermore, they do not leverage the specific mapping that there exists between images of paired datasets.
There have been other models that tried to include semantic information when transferring the style ([9], [10]), achieving impressive results.

Fig. 1: Van Gogh's *Wheatfield with Crows*. As in most of his paintings, shapes are distorted, so it is difficult for current state-of-the-art models to correctly extract semantic information from them.

However, those models were still not dealing with paintings where the shape of the objects is not very far from reality. In Van Gogh's paintings, the line is not used to describe reality, but it has an *expressive* function, transfiguring the reality itself (see Fig. 1). Throughout this paper, we will indeed try to improve state-of-the-art results by exploring different strategies, comparing, and then interpreting their outcomes.

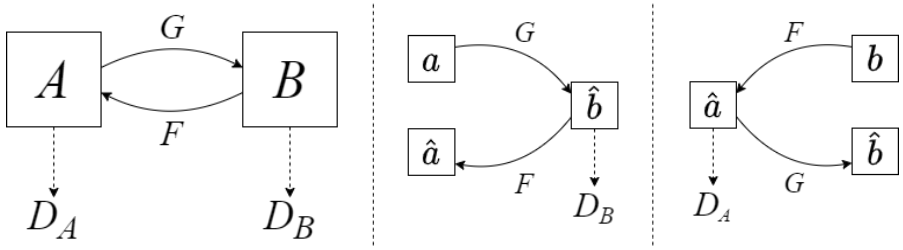
3 Proposed approaches

Differently from other methods, in all the approaches presented here a paired dataset has been used, which consists of pairs of images sharing a similar visual content. Its main strength is based on the fact that it can exploit the one-to-one mapping between the images of each pair to guarantee results more pertinent to the style contained in a specific painting.
The starting point was hence the dataset used by Zhu et al. to train CycleGAN

[7]. Van Gogh's paintings have been retrieved from WikiArt, while real photos have been downloaded from Flickr by using landscapes-related hashtags. Then, images have been manually paired, to match them as better as possible.

3.1 Pre-processing

Before applying style transfer, a pre-processing phase has been studied and applied. Since all the pre-trained models for image segmentation have been trained on real pictures, expecting to see good results would be too much pretentious if segmentation is directly applied to paintings. Hence, a strategy to overcome this issue consists in converting paintings into real photos.
For this purpose, a CycleGAN has been trained on two sets: the set of Van Gogh's paintings, consisting of 400 samples (set **A**), and the set of photographs (set **B**), containing 6853 images. The training process is visually explained in Fig. 2. The model has been trained with the default parameters used in the original paper [7] for a total of 120 epochs. Results and benefits of this pre-processing phase are discussed in Section 4.



(a) (b) (c)

Fig. 2: Paintings to photographs conversion using *CycleGAN* architecture. (a) A and B are respectively Van Gogh's paintings and photographs domains. CycleGAN's goal is to learn the two functions $G : A \to B$ and $F : B \to A$. Each domain is associated to a discriminator, respectively D_A and D_B: (b) D_B encourages G to translate every image $a \in A$ into a new image $\hat{b}$ indistinguishable from those belonging to domain B, and (c) vice versa for D_A and F.

CycleGAN has been chosen for this task because it ensures **cycle consistency**: when we translate from one domain to the other and back again we should arrive at where we started. In formulas:

$$\hat{a} = a \to G(a) \to F(G(a)) \approx a \tag{1}$$

$$\hat{b} = b \to F(b) \to G(F(b)) \approx b. \tag{2}$$

Let us now introduce two losses: the **adversarial loss** and **cycle consistency loss**. The first is used to improve the quality of fake images generated from one domain to the other. There are two adversarial losses, one for G:

$$\mathcal{L}_{GAN}(G, D_B, A, B) = \mathbb{E}_{b \sim p_{data}(b)}[\log D_B(b)] + \mathbb{E}_{a \sim p_{data}(a)}[\log(1 - D_B(G(a)))], \tag{3}$$

and one for F:

$$\mathcal{L}_{GAN}(F, D_A, B, A) = \mathbb{E}_{a \sim p_{data}(a)}[\log D_A(a)] + \mathbb{E}_{b \sim p_{data}(b)}[\log(1 - D_A(F(b)))]. \tag{4}$$

Functions G and F aim to respectively minimize these losses, while their adversaries D_A and D_B try to respectively maximize them. The second loss, instead, incentivizes the cycle consistency, and it is expressed as:

$$\mathcal{L}_{cyc}(G, F) = \mathbb{E}_{a \sim p_{data}(a)}[\|F(G(a)) - a\|_1] + \mathbb{E}_{b \sim p_{data}(b)}[\|G(F(b)) - b\|_1]. \tag{5}$$

The full objective of CycleGAN is then to solve:

$$G^*, F^* = \arg\min_{G,F} \max_{D_A, D_B} \mathcal{L}_{GAN}(G, D_B, A, B) + \mathcal{L}_{GAN}(F, D_A, B, A) + \lambda \mathcal{L}_{cyc}(G, F), \tag{6}$$

where λ controls the relative importance of the two objectives. For further details on CycleGANs refer to [7].

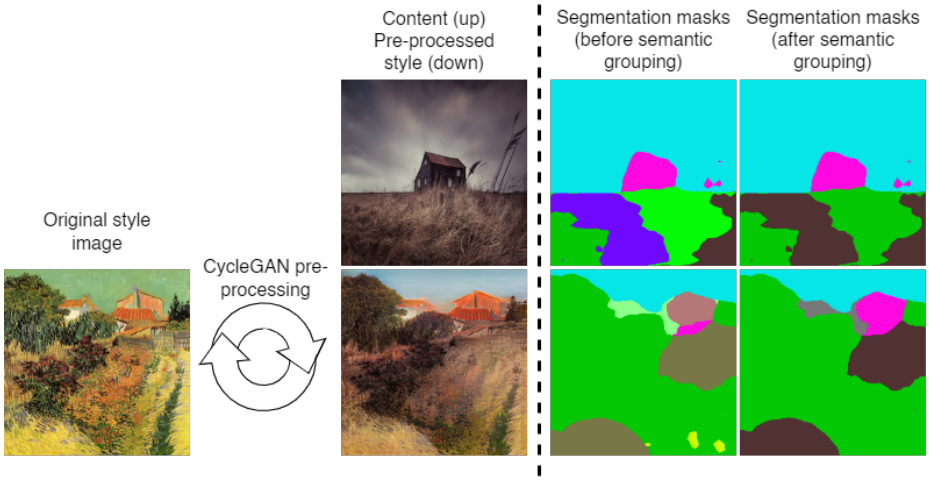


Fig. 3: Visual summary of the pre-processing phase.

Once converted all the images, a subset of 21 Van Gogh's paintings has been selected and paired with real photographs.
Then, all the selected images have been segmented using the approach of Penhouët et al. [1]. The latter is composed of two parts: first, a pre-trained CNN

called *Pyramid Scene Parsing Network (PSPNet)* creates a segmentation image; secondarily, a Knowledge Graph from a Python library called *Sematch* measures the similarity between two class words (e.g.: *sky* and *ground*). The purpose of this second part is to group semantically similar classes into a wider class, so that both the content and the style image will have the same classes. Fig. 3 shows how the pre-processing phase looks like for one single pair of images.
The semantic grouping is regulated by a parameter $\theta \in [0, 1]$ called the *semantic threshold*. When $\theta = 1$ no semantic grouping is applied, while with $\theta = 0$ all the classes are merged into one. The value used here is $\theta = 0.6$, which is the same used in the original paper [1].

Here is where the pre-processing phase reaches its end. From now on, when we talk about segmentation masks we are referring to the semantically grouped masks, and not to the "raw" ones.

3.2 Patch-based Neural Style Transfer

To pursue the goal of this paper, different strategies have been explored in an iterative process. The starting point was the *Neural Style Transfer* (NST) model introduced by Gatys et al. [4], that has been here used to transfer the style from each patch of the style image to the corresponding most similar patch of the content image.
In practice, each pair of content and style images is cropped into 9 patches, as well as their respective segmentation masks. The number of patches has been chosen in order to have more diversified matches, and all the parameters used to train the model are the same used in the original paper [4].

Then, for each patch of the content mask, it is retrieved the most similar patch of the style mask, according to a similarity measure. The final chosen measure was the **Euclidean Distance** since it overall showed more appealing results than **Manhattan Distance**. Another metric, called **Vector Cosine Angle Distance** has been tried and its implementation has been kept in the code, but it was discarded since it computed completely wrong matches. Fig. 4 illustrates a comparison between the influence of Euclidean and Manhattan distance on the final results.

Once two patches of the segmentation masks have matched, the corresponding patches of content and style images are fed to the architecture used in Neural Style Transfer paper [4]. This process is repeated for every patch in the content image and all the resulting patches are reconstructed into one image (see Fig. 5). The rationale behind the NST architecture is to extract features from a VGG19 for both style and content images, and jointly minimize their losses:

$$\mathcal{L}_{total}(\overrightarrow{p}, \overrightarrow{a}, \overrightarrow{x}) = \alpha \mathcal{L}_{content}(\overrightarrow{p}, \overrightarrow{x}) + \beta \mathcal{L}_{style}(\overrightarrow{a}, \overrightarrow{x}), \tag{7}$$

where α and β are the weighting factors for content and style reconstruction respectively, while $\overrightarrow{p}$, $\overrightarrow{a}$, and $\overrightarrow{x}$ are, in the order given, the photograph, the artwork, and the image that is generated.

The approach presented in this section is close to the CNNMRF method ([5]), but is different in terms of how the similarity between two patches is chosen (here

Fig. 4: A comparison between Euclidean and Manhattan distances for 3 different images. While the second image shows basically no difference between the two metrics, the first and the last images have important divergences. The resulting image in the first row is more close to the correct style transfer when using the Manhattan distance to compute the closest patch. This is visible from the sky, which in the Euclidean case is not referable to the style image. In the third row, instead, the Euclidean distance gives a better result, as it is evident from the style transferred to the children.

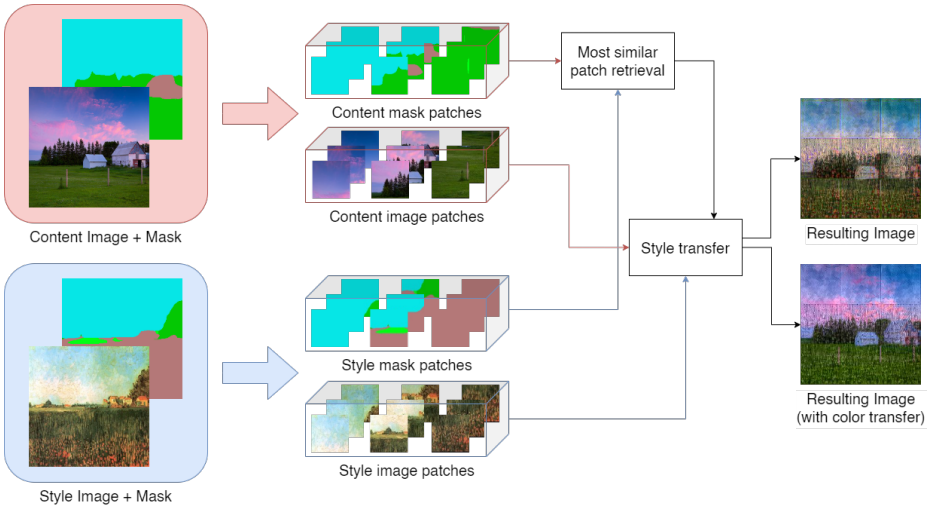


Fig. 5: Style is transferred patch-by-patch from the style image to the content image basing on the Euclidean Distance between the patches of both the content mask and the style mask.

the whole process is based on semantic content).
Nevertheless, this approach is not exploiting all the information contained in the semantic mask of a certain painting. Let's say the *sky* of a photo is cropped into two patches P_{c_1} and P_{c_2}, and the *sky* of the paired painting is cropped into patches *sky* P_{s_1} and P_{s_2}. When finding the most similar patch, it could happen that both patches P_{c_1} and P_{c_2} match with P_{s_1}. In this case, the information held by P_{s_2} is not exploited at all, and the final image shows repetitions of the very same style for various patches. One could constrain the model to pick a different patch if others have already matched. However, in this way we could lead the model to choose a completely semantically different patch, resulting in a wrong style transfer.
Furthermore, the resulting images still show the junctions between every patch, which is visually not good. Hence, it is clear that this model is too limited, and a better model is needed. For this reason, the architecture presented in the next chapter aims at improving the semantic style transfer, exploiting all the information contained in an image.

3.3 Neural Doodle

To ensure full transfer of the style, an improvement has been to use Champandard's model [3] to each pair of masks resulting from the pre-processing phase. This architecture consists of an augmented VGG19 network containing additional semantic channels computed by downsampling a semantic map provided

as input. The objective is to minimize a loss function similar to the one presented in Section 3.2, consisting of a content reconstruction error E_c and a style remapping error E_s, both weighted by two parameters α and β:

$$E = \alpha E_c + \beta E_s. \tag{8}$$

The algorithm first extracts patches from the semantic layers; then, for any patch in the current image and layer, it is computed its nearest neighbor using normalized cross-correlation, that takes into account weighted semantic map. At this point, it is computed the gradient of the loss function defined above, where E_c is the same term used in Gatys et al.'s paper, while E_s is defined as the sum of the Euclidean distances between all the patches of a certain layer in the current image to the closest style patch.

Actually, this strategy led to better results (see Section 7) compared to the model presented in Section 3.2, but the content image was poorly taken into account. The outcomes of this *Neural Doodle* architecture in practice are often too much similar to the style images, and the only difference between them is that elements in the image are rearranged according to the content mask (see Fig. 6).

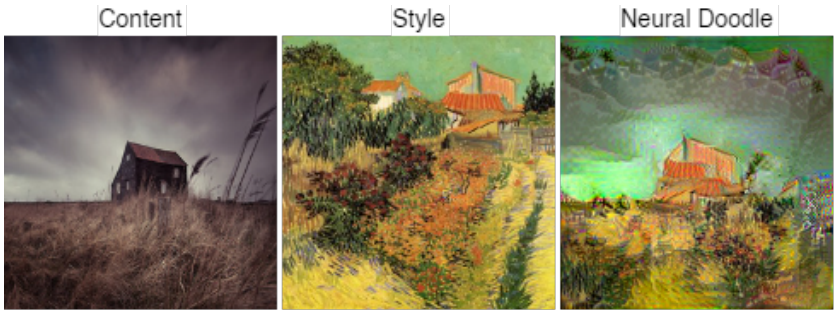


Fig. 6: Drawbacks of *Neural Doodle*. Here, the only information coming from the content image is the position of the house, since the same house in the resulting image looks like a copy of the group of houses in the style image.

3.4 Photo-Realistic into Painting-Like Artistic Style Transfer

The last architecture presented here is also the one having the best results. It is based on both the *Automated Deep Photo Style Transfer* (ADPST) model by Penhouët et al. [1], and the classical *Neural Style Transfer* model [4]. The latter was explained in Section 3.2, while the first one was introduced in Section 3.1.

ADPST model is based on *Deep Photo Style Transfer* ([11]), with the difference that in ADPST segmentation masks are created automatically. Its objective

is to minimize the following loss:

$$\mathcal{L} = \sum_{l=1}^{L} \alpha_l \mathcal{L}_c^l + \Gamma \sum_{l=1}^{L} \beta_l \mathcal{L}_s^l + \lambda \mathcal{L}_m + \eta \mathcal{L}_a, \tag{9}$$

where:

- $\mathcal{L}_c$ is the *content* loss ([4])
- $\mathcal{L}_s$ is the *augmented style* loss, which improves NST style loss ([4]) by taking semantic regions into account
- $\mathcal{L}_m$ is the *affine* loss, based on locally affine transformations in color space that preserves edges in small patches
- $\mathcal{L}_a$ is the *image assessment* loss, that ensures image quality based on ratings of the aesthetics of images
- Γ, α_l, and β_l are constant weights that can be used to configure layer $l = 1, ..., L$ preferences or to weigh the two loss functions
- λ is the photo-realism regularization term
- η is a weight to scale the image assessment loss

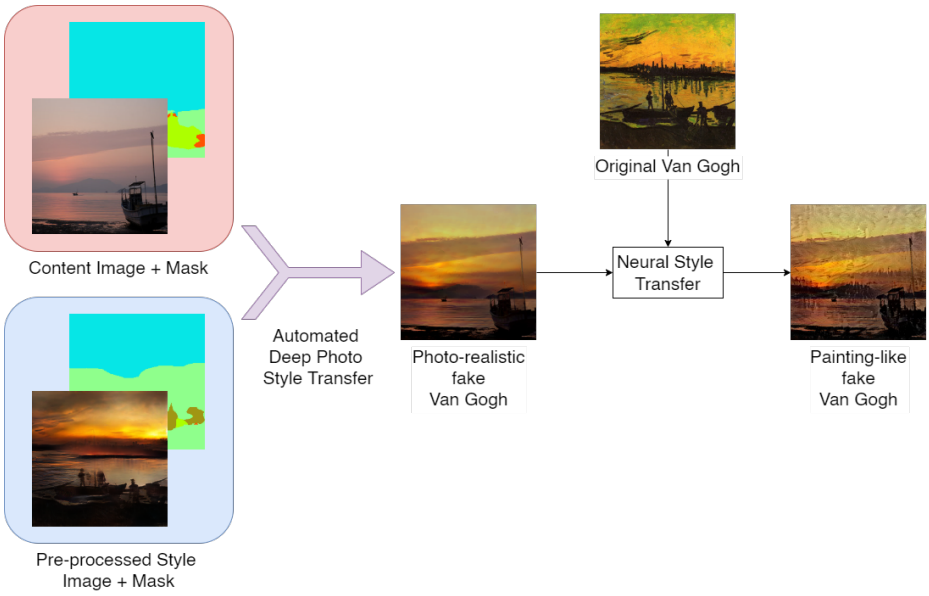


Fig. 7: In this workflow, the style is first transferred in the photo-realistic domain from the painting to the photo. The resulting image and the original painting are then fed into the *NST* model as content and style images, respectively.

The loss is minimized for a certain number of iterations, that, according to Penhouët et al., should be a minimum of 1000. The authors also observed

that good results are achieved with about 2000 iterations and improvements after 4000 iterations are most of the time negligible; therefore, the number of iterations was here set to 3000. For further details on the loss and the training process, refer to the original paper [1].

As already mentioned, the workflow presented in this Section combines both ADPST and NST models, and it is shown in Fig. 7. Each pair of content and pre-processed style images is fed into the ADPST architecture, which transfers the style in the photo-realistic domain. In this way, we are able to exploit the segmentation masks of both the images, mainly transferring the palette and the semantic content of the painting.
However, we still need to map the resulting image to the paintings domain. To accomplish this task, we resorted to the Neural Style Transfer architecture, because its loss function is basically the foundation of ADPST's loss. Hence, the final painting-like fake Van Gogh is the result of a style transfer from the original (not pre-processed) painting to the photo-realistic fake Van Gogh obtained from ADPST.

4 Experimental results

All the models presented in Section 3 have been run on the selected subset of Van Gogh's paintings. Also, to evaluate the effect of coupling, certain images have been paired with more than one photograph. Results from three baselines were generated in order to compare the outcomes of the architectures presented in this paper with the state-of-the-art. Fig. 8 illustrates some of them, while the rest can be found in Section 7. The baselines that were chosen are the classical Gatys et al.'s Neural Style Transfer (NST) [4], Li and Wand's CNNMRF, because it is a patch-based improvement of the previous one [5], and CycleGAN, which involves the usage of an unpaired dataset [7].
Let's now call PbP (Patch-by-Patch transfer) the method presented in Section 3.2, Neural Doodle the model based on the homonym architecture by Champandard [3], and PRPLAST the Photo-Realistic into Painting-Like Artistic Style Transfer method explained in Section 3.4. Generally speaking, results of PRPLAST are at first sight those that make the most of the trade-off between content and style, and this justifies why we previously stated that this was the best among the three methods presented. Indeed, apart from the problem of the visible junctions interconnecting each patch, PbP shows a sort of repetition of the style, lacking uniformity in the resulting image. On the other hand, Neural Doodle's results are much too style-oriented. As we can indeed see in the last row of Fig. 8, the algorithm transfers the style of the sky without differentiating between the portion of sky that is visible and the clouds, ending up completely ignoring them. Still, some results of this model are visually captivating (see Section 7 for more examples).

Concerning the three baselines, we can easily observe that CNNMRF's results are too close to the content image, in the sense that the style is in general poorly transferred. CycleGAN, instead, shows impressive results, but they are

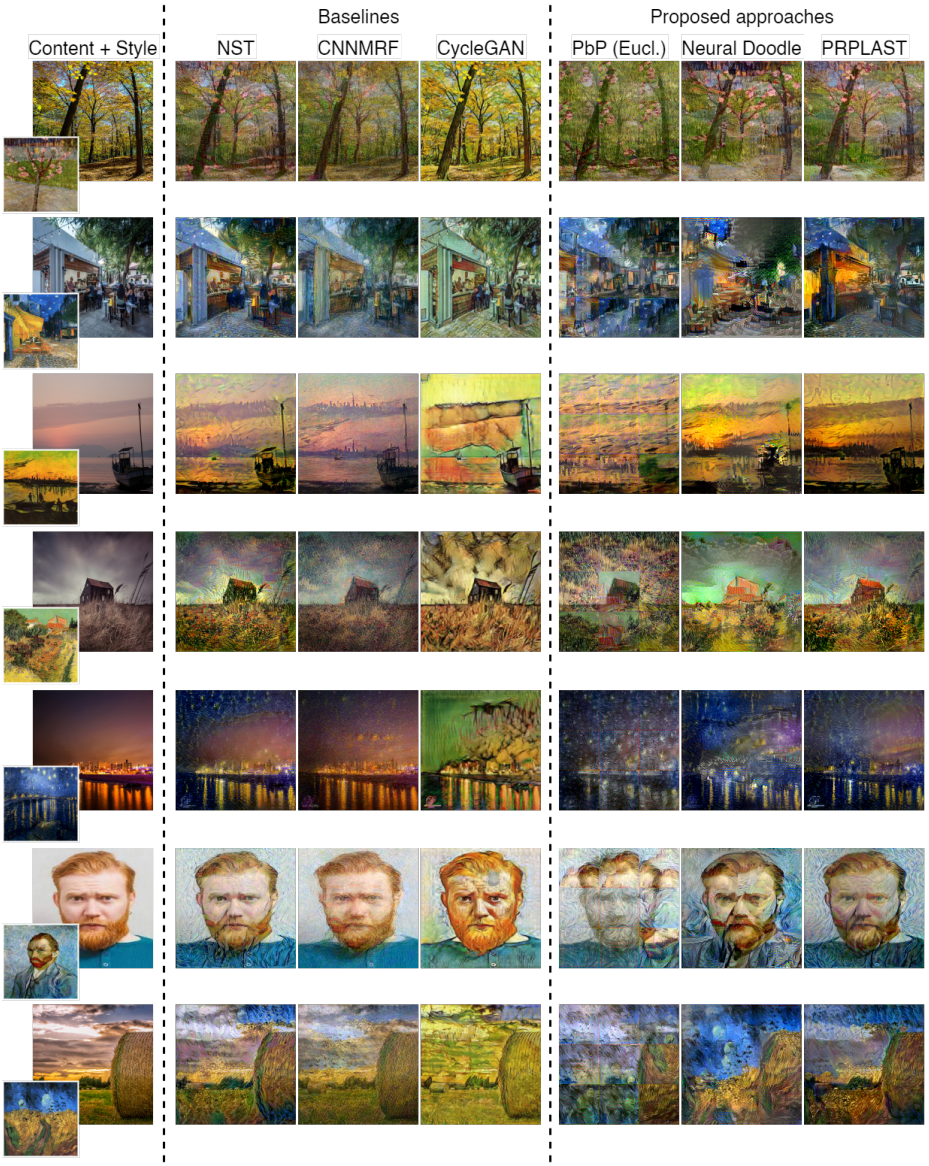


Fig. 8: Some examples of results. For the PbP model, only results obtained using the Euclidean Distance are shown, since it proved to be the visually best metric.

generalized to the whole dataset of photos. Indeed, due to the nature of GANs, the style is not captured by a precise correlation between content and style image, but it is dispersed throughout every image. This can be also seen by the fact that its results are the most different among all those presented. Classical Neural Style Transfer's results are amazing as well, especially because this is the eldest model, but they are too simplistic. Since the objective of this model is to jointly minimize only content and style loss, the resulting image tends to be an overlap between content and style image. In any case, NST can be defined as the main competitor of PRPLAST, which actually is the one among the three proposed that improves the most the state-of-the-art. This improvement is inherent in the pre-processing phase, which allows to locally map semantic information from one image to the other.

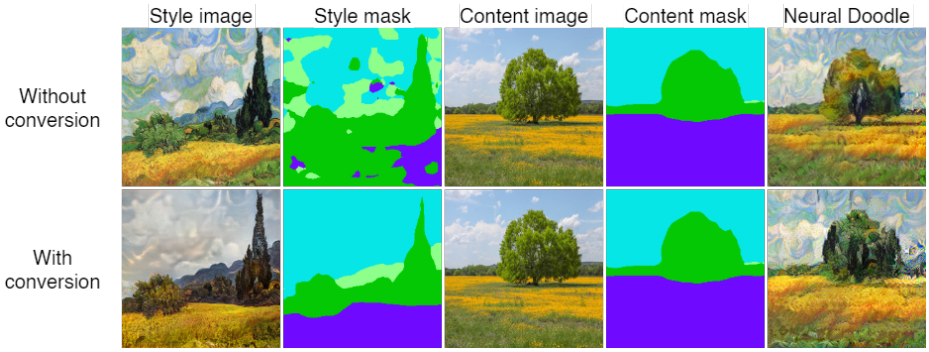


Fig. 9: Benefits of painting to photo conversion. As an example, in the first row it is shown the result for *Neural Doodle* when the style image is the original Van Gogh's painting. The second row shows the same result when the style image is first converted into a real picture. The main point of difference is here the style mask, which is more precise in the second case.

Intermediate results to interpret come from the pre-processing phase itself, whose advantages are easily visible in the quality of semantic masks that have been generated from style images. In Fig. 9, the style mask extracted from the photo-realistic painting is clearly closer to the actual content of the painting itself. Indeed, in the mask generated directly from the original artwork the sky is segmented in multiple instances. Furthermore, the ground is confused with the greenery, as we can see from the mixture of purple and green in the style mask. This is not the case when the mask is instead generated from the converted style image, whose result is more pertinent to the content of the painting.

5 Limitations and Discussion

Drawbacks of the Patch-by-Patch transfer and the Neural Doodle models have been investigated in Sections 3.2 and 3.3, respectively. For what concerns the Photo-Realistic into Painting-Like Artistic Style Transfer architecture, its limitations are mainly related to the quality of the semantic masks. If on the one hand paintings of landscapes are giving the best results, on the other hand we get worse results when it comes to paintings containing objects or people (Fig. 10).

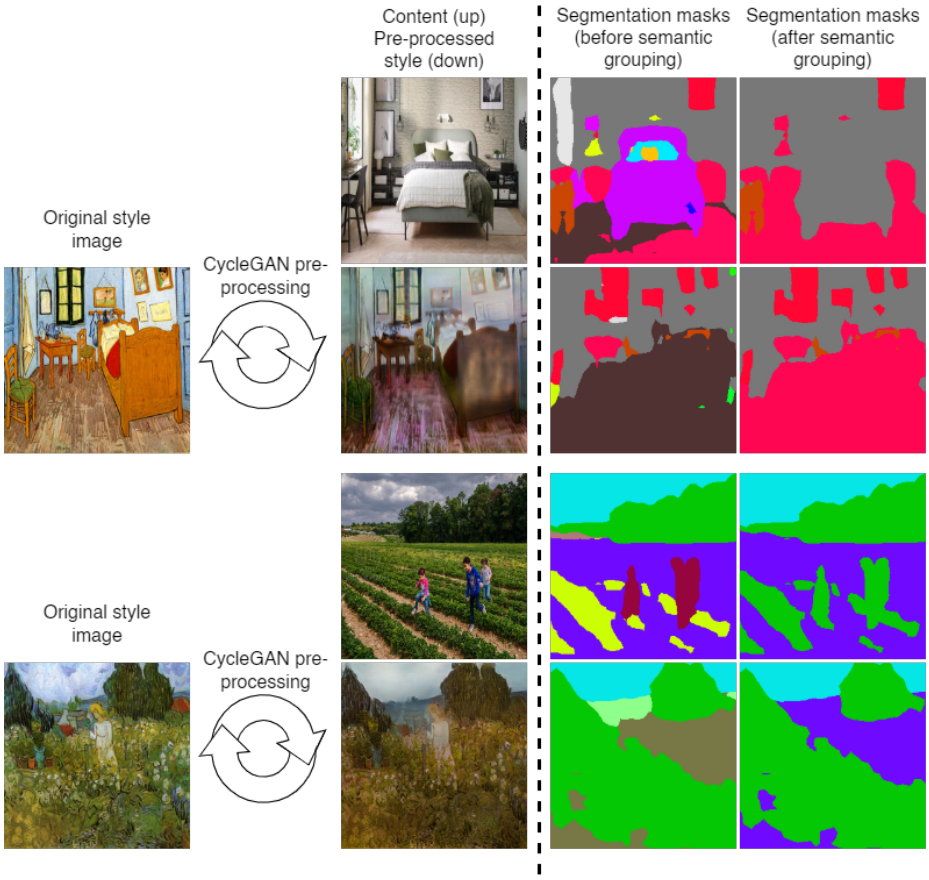


Fig. 10: Challenging scenarios for the pre-processing step. In the first pair of content and style images, it is evident that the model struggles in distinguishing the bed from the hardwood floor. Other elements in the style image are poorly identified as well, for example the chair. In the second example, instead, people in the pre-processed style image are confused with the ground.

To improve the quality of these masks, a first possible solution is to include photos of people and common objects in the appropriate dataset. Doing so, CycleGAN will be aware of how to pre-process those kinds of images, and results may get better. Another solution, which is not mutually exclusive with the first one, can be to tune the parameters of Penhouët et al.'s Automated Deep Photo Style Transfer (ADPST) model, in order to see what set of parameters better adapts to photo-realistic Van Gogh's artworks.
Anyway, these improvements are constrained by the fact that ADPST's segmentation is optimized for "Scene Parsing", hence a re-training of the model might be necessary.

Nevertheless, not all the landscapes paintings have been well pre-processed neither. As shown in Fig. 11, when elements of the scenery are depicted too abstractly it is more challenging for the model to map them to the real photos domain and then generate segmentation masks. With the current pre-processing pipeline it could be difficult and/or time-consuming to try to improve the quality of segmentation masks in such a scenario. Consequently, it may be more convenient to use a different technique to convert paintings into real photos.
Current literature offers *Art2Real* architecture [12], that uses CycleGAN to map patches of each painting to sub-domains, called *memory banks*. Essentially, each photograph is segmented and cropped, and patches sharing the same semantic information (e.g. patches containing the sky) are stored in a specific memory bank. Then, each patch of the paintings is mapped to the right sub-domain (*sky*, *ground*, etc.) and is converted independently from the other patches. In such a way, semantic information can be exploited in the pre-processing step as well, and results might refine.

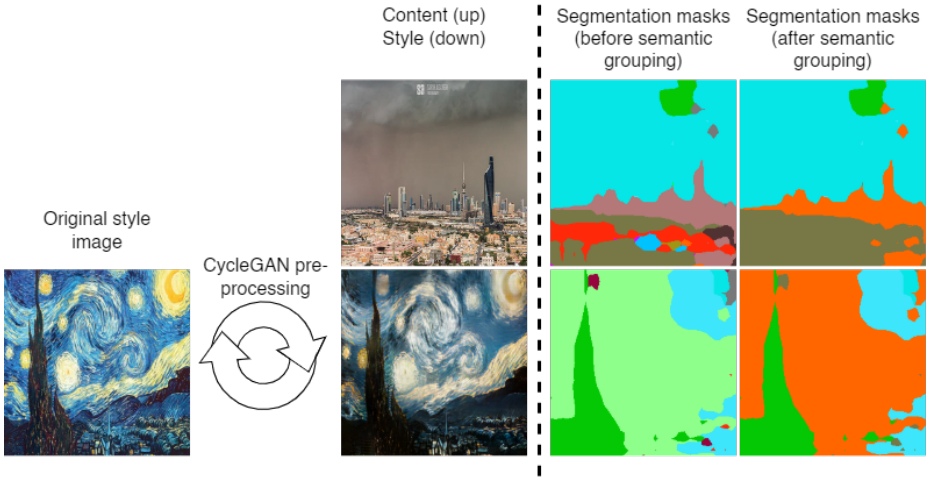


Fig. 11: A challenging example for the pre-processing step that portrays a landscape.

By improving the quality of semantic masks, it is likely that style transfer of each of the models presented in Section 3 returns better results. However, the patch-based method of Section 3.2 has more margin for improvement. In fact, the main issue to solve is the one related to the visible junctions in the reconstructed image. One solution to smooth the transition between the boundaries could involve the usage of a 2-dimensional weighted mean over each patch. Anyway, other solutions are left to the reader either for the pre-processing or for the style transfer phases.

One last crucial part to optimize is the dataset generation. In fact, images have been manually paired before starting experimenting. However, finding a new method to automatically pair each image would be preferable. In such a way, the amount of time needed to build a paired dataset will be reduced, and a bigger dataset can be created. With enough images, it could be also possible to train a classifier that detects fake artworks among the results obtained from PRPLAST and the real Van Gogh's paintings. Its results can be then used as a numerical metric to compare the performance of PRPLAST with the state-of-the-art. In addition to the previous one, another metric to use could be to conduct a user study where users give their impressions on each image, according to *painting-likelihood* scores and *style faithfulness* preferences [11].

6 Conclusion

In this paper, we presented three new ways of approaching to Neural Style Transfer. The novelty is mainly given by the pre-processing phase, whose goal is to convert paintings into real photos in order for current segmentation networks to deliver better results. For what concerns the style transfer process, three algorithms have been studied: Patch-based Neural Style Transfer, Neural Doodle, and Photo-Realistic into Painting-Like Artistic Style Transfer. The first one tried to exploit semantic information of each pair of photograph and artwork patch-by-patch. The second algorithm tried to re-arrange the content of each painting according to the semantics of the associated photo. Lastly, the third model wanted to transfer the style by exploiting the power of current Neural Networks in processing real photos. To pursue this objective, it transferred the style twice: first, in the photo-realistic domain, then in the painting-like domain. Generally, results of the last model showed an improvement of current state-of-the-art. Still, even if not all the results of this and the other models led to improvements, there is important room for further refinements, and Section 5 well-explains what the future work may be.

7 Annexes

Fig. 12: Results comparison pt.1

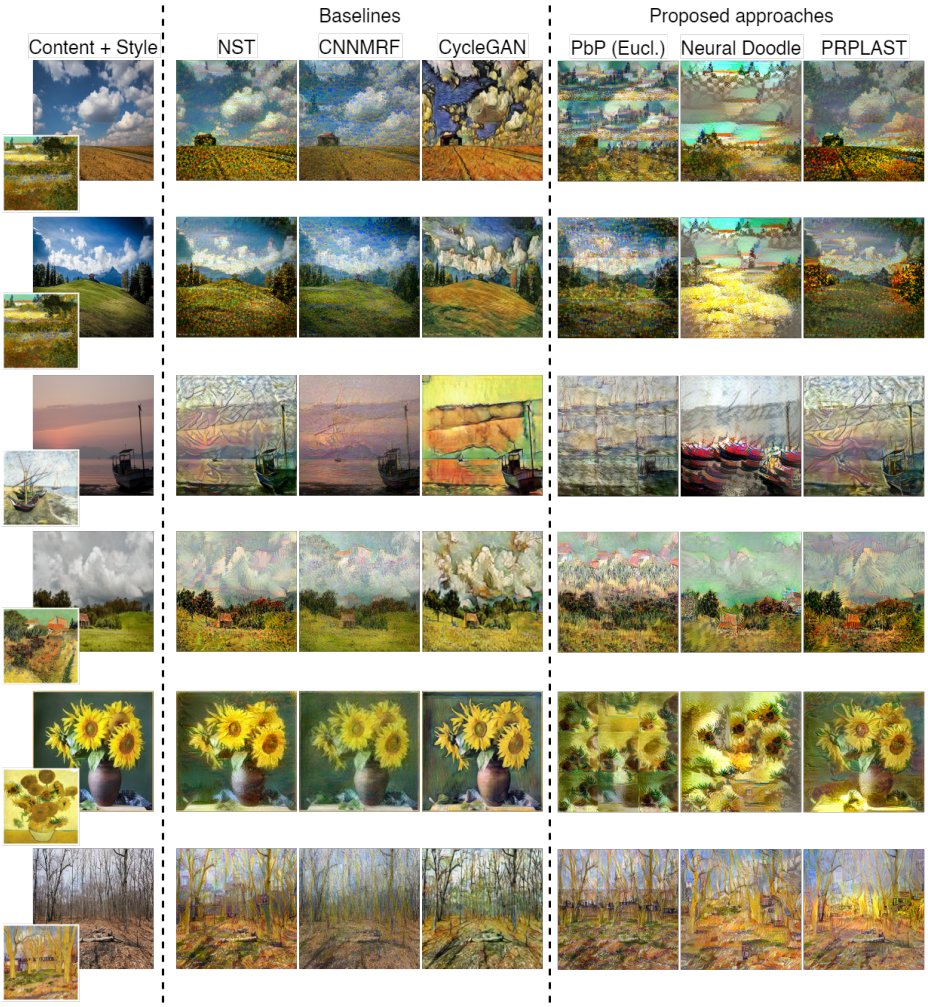


Fig. 13: Results comparison pt.2

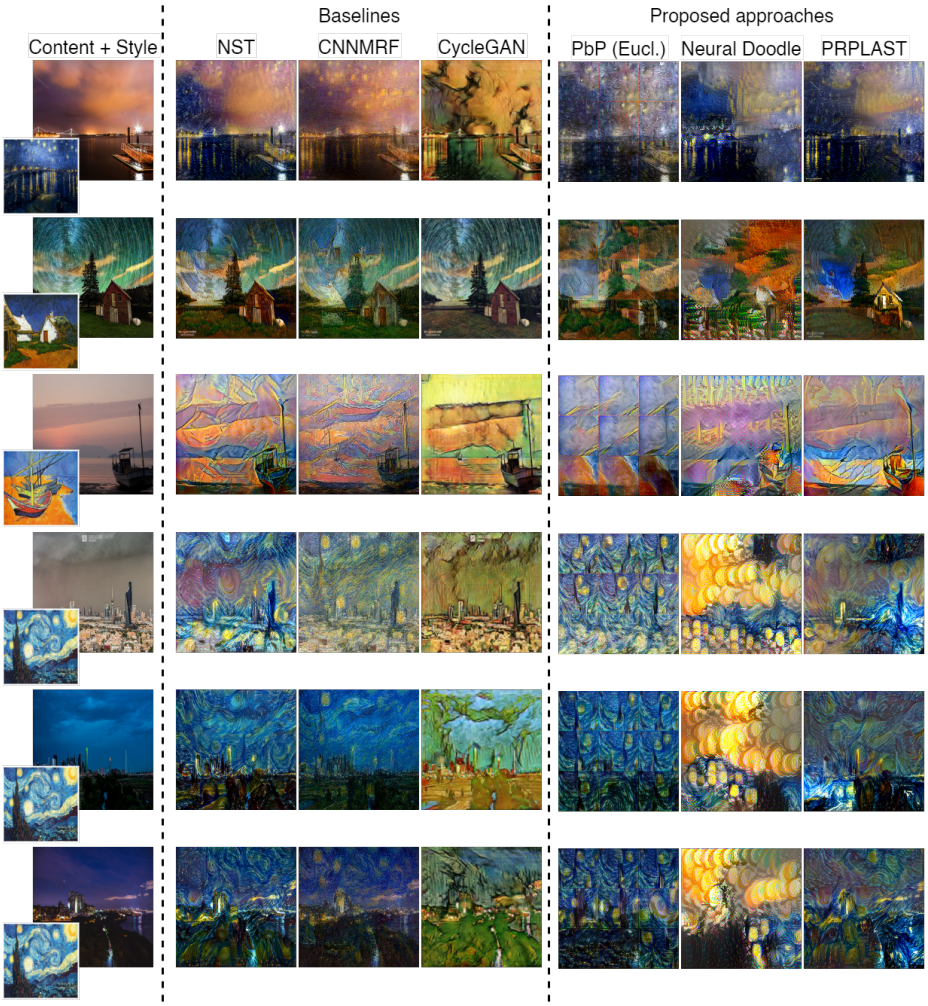


Fig. 14: Results comparison pt.3

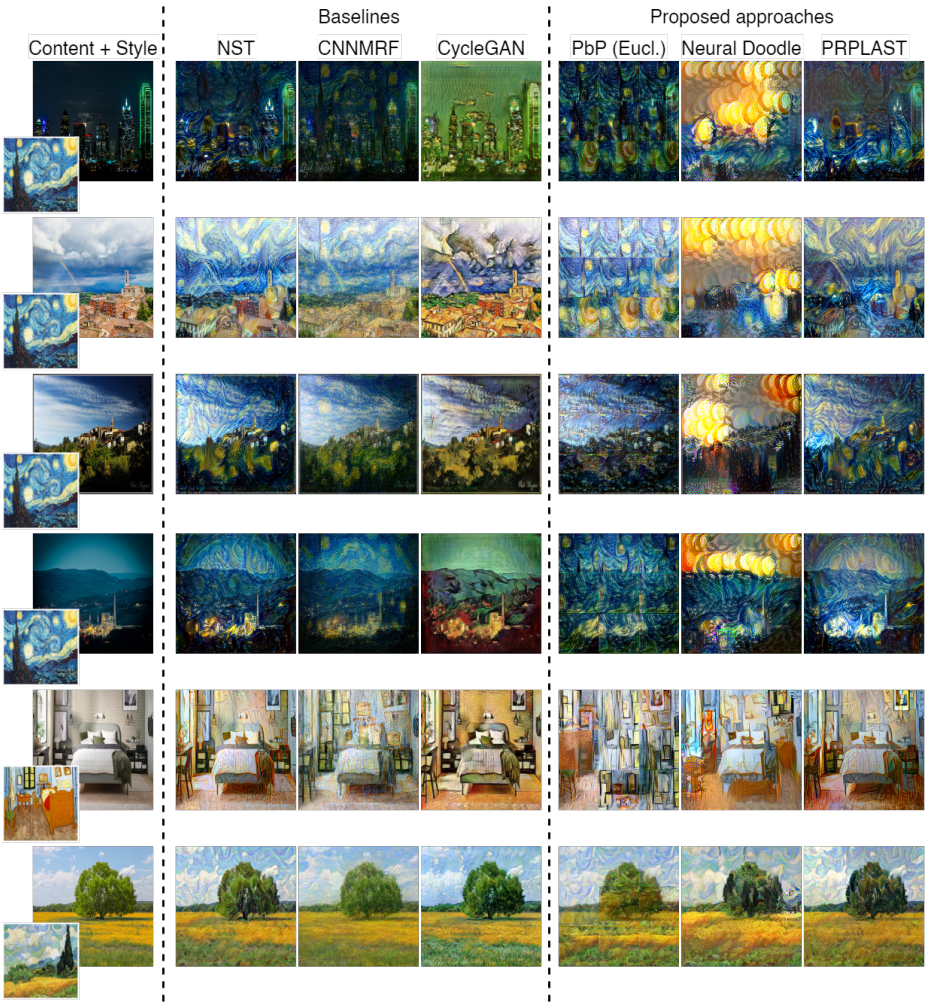


Fig. 15: Results comparison pt.4

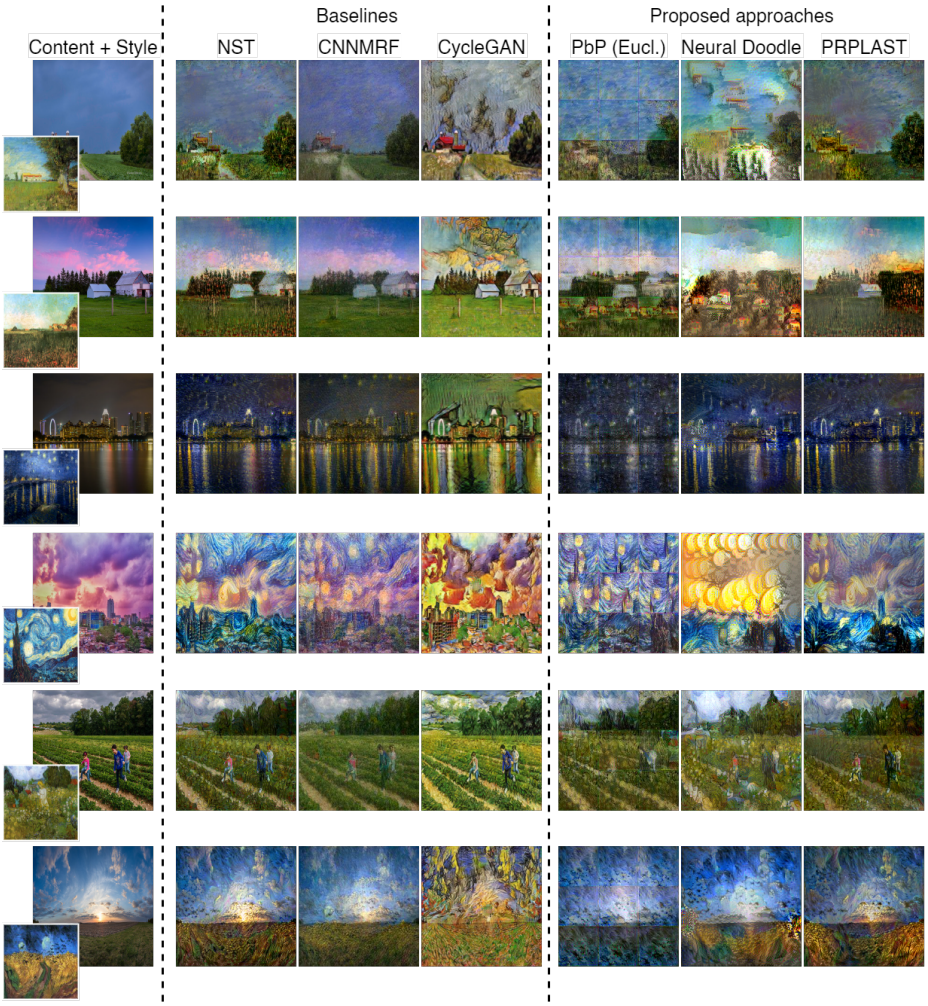


Fig. 16: Results comparison pt.5

References

[1] Sebastian Penhouët et al. *Automated Deep Photo Style Transfer*. 2019. arXiv: 1901.03915 [cs.CV].

[2] Ronghang Hu et al. *Learning to Segment Every Thing*. 2018. arXiv: 1711.10370 [cs.CV].

[3] Alex J. Champandard. *Semantic Style Transfer and Turning Two-Bit Doodles into Fine Artworks*. 2016. arXiv: 1603.01768 [cs.CV].

[4] Leon A. Gatys et al. *A Neural Algorithm of Artistic Style*. 2015. arXiv: 1508.06576 [cs.CV].

[5] Chuan Li and Michael Wand. *Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis*. 2016. arXiv: 1601.04589 [cs.CV].

[6] Samet Hicsonmez et al. *GANILLA: Generative Adversarial Networks for Image to Illustration Translation*. 2020. arXiv: 2002.05638 [cs.CV].

[7] Jun-Yan Zhu et al. *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks*. 2020. arXiv: 1703.10593 [cs.CV].

[8] Jeff Donahue et al. *Adversarial Feature Learning*. 2017. arXiv: 1605.09782 [cs.LG].

[9] Huihuang Zhao et al. *Automatic Semantic Style Transfer using Deep Convolutional Neural Networks and Soft Masks*. 2017. arXiv: 1708.09641 [cs.CV].

[10] Joo Hyun Park et al. "Semantic-aware neural style transfer". In: *Image and Vision Computing* 87 (2019), pp. 13–23. ISSN: 0262-8856. DOI: https://doi.org/10.1016/j.imavis.2019.04.001. URL: https://www.sciencedirect.com/science/article/pii/S0262885619300435.

[11] Fujun Luan et al. *Deep Photo Style Transfer*. 2017. arXiv: 1703.07511 [cs.CV].

[12] Matteo Tomei et al. *Art2Real: Unfolding the Reality of Artworks via Semantically-Aware Image-to-Image Translation*. 2019. arXiv: 1811.10666 [cs.CV].