

Virtual Cleaning of Artworks Using a Deep Generative Network^{*}

Morteza Maali Amiri^{1,*†}, David W Messinger^{1,†}

¹Rochester Institute of Technology, Chester F. Carlson Center for Imaging Science, 54 Lomb Memorial Drive, Rochester, NY 14623, USA

Abstract

It is well-known that the varnish applied to artwork yellows with time changing its appearance accordingly. Conservators are then sometimes prompted to physically clean the artwork in an attempt to recover the original look of the work. At times, the conservators only partially clean the artwork first. They then try to virtually clean the rest of the artwork to visualize the result of the cleaning before physically cleaning the entire piece. There have been many different approaches that have been proposed to virtually clean a partially cleaned artwork. All of them have some limitations, the low accuracy of which is the main one. In this paper, a deep generative network is proposed to virtually clean a partially cleaned artwork in the RGB domain. The proposed generative model consists of several up-sampling and down-sampling convolution blocks and skip connections with a symmetric architecture. The loss function is calculated using the part of the artwork that has been physically cleaned for which we have access to both RGB images before and after cleaning. Therefore the network is able to clean the whole artwork using only a small area of it that has already been physically cleaned. A Macbeth ColorChecker and images of the *Mona Lisa* are used to test the approach and the results are compared with a recent approach available in the literature which uses a Convolutional Neural Network (CNN). The results are found to be acceptable given that the approach proposed herein has a potential to be applied in a real situation and there is no need for a large training dataset, on which the CNN method relied on.

Keywords

Deep Generative Network, Virtual cleaning of artworks, Varnish removal

1. Introduction

Artworks are usually varnished for the purpose of protection. Although successful in their main purpose, with time, this application can change the visual qualities of artworks [1, 2, 3, 4]. Therefore, physically removing the aged varnish in order to reestablish the original appearance of the artwork becomes of great importance [5, 6]. There have been two major approaches to clean artwork, namely, physical and virtual cleaning. In the physical approach, the conservator physically removes the varnish layer using a solvent and gel system. These types of cleaning are very time-consuming and can also be detrimental to the artwork [7, 8, 9]. Virtual cleaning,

The 11th Colour and Visual Computing Symposium 2022, Sep 8–9, 2022, Gjøvik, Norway *

Supported by Xerox chair at Rochester Institute of Technology.

*Corresponding author.

† Morteza Maali Amiri

✉ mm2391@rit.edu (M. M. Amiri); dwmpci@rit.edu (D. W. Messinger)

ORCID 0000-0002-0391-3310 (M. M. Amiri); 0000-0002-2273-9194 (D. W. Messinger)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

on the other hand, refers to the outcome of the physical approach using simulation. Virtual cleaning could provide the conservator with the likely appearance of the cleaned artwork, helping them to see if the physical cleaning is necessary and potentially guide their work.

Most of the studies done in the area of virtual cleaning are based on first cleaning a small part of the painting physically. and they use an RGB image of the painting before and after cleaning. Using that small part, for which they have data belonging to both cleaned and unclean state, they attempt to virtually clean the entire painting producing a visualization of the cleaned work. They typically do that through fitting some type of regression to the data obtained from the small area before and after cleaning. They then apply the same regression model to the rest of the painting which leads to the artwork being virtually cleaned [10, 11]. Papas and Pitas (2000) stated that the RGB color space of the camera does not work well and they proposed to use the CIELAB color space instead, asserting that CIELAB works better than RGB color space. Virtually cleaning the *Mona Lisa* was another breakthrough in the field of virtual cleaning [12]. Having access to the classical paints used in 16th century Italy, the authors were able to make a varnished and unvarnished color chart out of them. They were able to extract the relationship between the varnished and unvarnished color chart enabling them to estimate the unvarnished version of *the Mona Lisa* [12]. Palomero and Soriano (2011) developed the first neural network approach trying to virtually clean artworks [13]. They also first cleaned a part of the artwork and then trained a shallow network using that small part. They then used the same model to clean the rest of the artwork [13]. Trumpy, *et al.* (2015) developed the first physics-based model in order to virtually clean artworks [14] through making a few simplifying assumptions, such as that a dark site on the painting is a “perfect” black that absorbs all incident light (perfect meaning not grayish) and the varnish spectral reflectance is wavelength independent. Through first finding the darkest and lightest part of the painting and cleaning them, they were able to estimate the spectral transmittance of the varnish layer which would be used to estimate the cleaned spectral reflectance of the entire painting [14]. Kirchner, *et al.* (2018) used Kubelka-Munk trying to estimate the virtually cleaned artworks [15]. In order to do that, they first characterized the varnish layer through cleaning the artworks at a few spots that appeared white allowing them to compute the spectral transmittance of the varnish. Characterizing the varnish layer enabled them to estimate the cleaned version of the whole painting [15]. Linhares, *et al.* (2020) did a similar work as [15] through characterizing the varnish layer first. However, they characterized the varnish layer through removing the whole varnish and measuring the spectral reflectance of the painting before and after varnish removal [16]. The latest work in the area of virtual cleaning of artworks belong to Maali Amiri and Messinger (2021) [17]. They first developed a Convolutional Neural Network (CNN) model. The network was trained on images of natural scenes and humans that were artificially yellowed mimicking the impact that varnish has on the artwork visually. They were able to visualize the cleaned version of artworks using their proposed CNN model in a very acceptable manner [17]. The methods proposed until now suffer from a few limitations, namely, the requirement to specify the perfect black and white regions on the painting, the need to have access to spectral data, generalizability of the method to other works, and the need to have access to a large set of data for training.

In this work, we propose a Deep Generative Network (DGN) to virtually clean a partially cleaned artwork. The generative model we use herein has been used in the area of remote sensing for the purpose of denoising of the hyperspectral image and single image super-resolution

[18, 19]. The authors developed a convolutional generative network that was able to take in a noise cube and output a super-resolved remotely sensed image. The network is deep and symmetrical and has also borrowed the idea of skip connections from U-Net enabling it to use residual information as best as possible. In this work, we have modified the network to fit our purpose. Instead of feeding the network with a random noise image, we feed in the RGB image of the uncleaned artwork. To be more specific, we have information of a small area of the painting before and after cleaning. The RGB image of the artwork is first changed into CIELAB, and the a^*b^* channels are used to train the network. The loss function, on the other hand, is computed between the uncleaned and the corresponding cleaned area of the artwork (the small area for which we have access to both cleaned and uncleaned data). The model is tested on Macbeth ColorChecker and the *Mona Lisa* that are partially cleaned. The results show that our approach here has done a better job compared to the model proposed by [17] when it comes to the *Mona Lisa* but has done slightly worse when compared to [17] for the Macbeth ColorChecker. Overall, the method proposed herein is more applicable to a real situation where the conservator has no access to a large set of data with which to train the model. Comparing our model to that of [17] seems fair as in their paper they showed that their model had outperformed the only physics-based model proposed for artwork virtual cleaning [14, 17].

This paper is laid out as follows: the next section will present the specifications of the data, while the method will be explained in a more detailed manner along with the evaluation metrics and experimental environment. After that, results are presented along with discussions in the next section. Finally the conclusions are presented.

2. Methodology

In this section, the data used are explained and the proposed algorithm is described in detail.

2.1. Data

One of the datasets used to test the proposed method is the Macbeth ColorChecker spectral reflectance data. The spectral reflectances were artificially yellowed using the same formula suggested by [17] in the spectral domain. The artificially yellowed spectral reflectance mimics the visual impact varnish has on the painting. So, due to the Macbeth ColorChecker having a wide range of colors along with neutral patches, we use it as an initial test for our approach. The Macbeth ColorChecker is simulated in a way that it is “varnished” with a layer of a particular spectral reflectance and transmittance (generally speaking, varnish is yellow, and its spectral reflectance and transmittance should represent that [14, 15]), as explained by [17]. The yellowed spectral reflectances and the originals were converted into sRGB data afterwards. The Macbeth ColorChecker was primarily used to assess the feasibility of the proposed methods before application to a well-known work of art. Consequently, we apply the network to the *Mona Lisa* to further test the network. The varnished and cleaned versions of the *Mona Lisa* are taken from [12].

2.2. Deep Generative Network (Architecture and Application)

In this section, the Deep Generative Network (DGN) that has been developed in this work is described. This method requires only a small area of the artwork to be cleaned. Then using the data of both the cleaned and varnished conditions of the same area, the network learns how to map from the uncleaned condition to the clean one. It then applies the same map to the rest of the artwork resulting in a virtually cleaned artwork.

The idea behind a DGN is to learn the relationship $x = f_{\theta}(z)$, which maps an image z to another image x . This approach is used here to recover the virtually cleaned artwork from the unclean one in the RGB color domain. The goal here is to generate image X , which is the virtually cleaned image of the varnished artwork. Through feeding the varnished image Z into the generator, image X with this characteristic will be attained. Z is the RGB image of the artwork before cleaning. As mentioned above, only a small area of the painting is cleaned and we have the RGB image of that area for both cleaned and uncleaned conditions. Let us call the area of the painting for which we have both the cleaned and uncleaned data A . The RGB image of this area that is physically cleaned is called A_c and the corresponding RGB image of this area that belongs to Z (that is unclean) is A_u . It makes sense that A_u belongs to Z as Z is the RGB image of the uncleaned artwork. When Z goes through the network, the part corresponding to A_u is taken out and the pixel-wise error between A_u and A_c is calculated to compute the loss, which is then back-propagated to the generator, through which the parameters θ of the mapping function are optimized. Fig. 1 shows the process described. It should be noted that there is no training in a traditional sense using this approach. The error computed between the A_u and A_c is back propagated to the generator and the generator will clean the whole image using this error coming from the loss function. This cleaning process is taken place step by step at each epochs, until the network reaches the maximum number of epochs.

Through trial and error we come to know that the network works better in CIELAB color space than in RGB. This improvement in the neural network performance by changing the color space to CIELAB has been reported in the literature as well [11]. Therefore, we first convert the RGB image, Z , into the CIELAB color space. The L^* channel is then set aside and the a^*b^* channels, as input, go through two main modules of the network, consisting of several blocks as follows:

1) The down-sampling block $d^{(i)}$: Each $d^{(i)}$ is composed of convolutional layer $C_d^{(1)}(i)$ also performing the down-sampling operation through setting the stride $S = 2$. After that, batch normalization and the LeakyReLU activation layer are performed. The output is then fed into the next convolutional layer $C_d^{(2)}(i)$ with the same stride. Similar to the first convolutional layer, this operation is followed by a batch normalization layer and the LeakyReLU activation function. $C_d^{(1)}(i)$ and $C_d^{(2)}(i)$ can be set to different kernel sizes and different numbers of filters shown as $k_d^{(1)}(i)$, $k_d^{(2)}(i)$, $n_d^{(1)}(i)$ and $n_d^{(2)}(i)$.

2) The up-sampling block $u^{(i)}$: Each $u^{(i)}$ consists of a few stacked layers. Opposite to the down-sampling blocks, batch normalization is the first layer. Afterwards, the first convolutional layer $C_u^{(1)}(i)$ with $S = 1$ and a batch normalization and LeakyReLU activation function are used. The output is then fed into the next convolutional layer $C_u^{(2)}(i)$. The output, after batch normalization and non-linear activation, is input into the bilinear up-sampling layer with factor

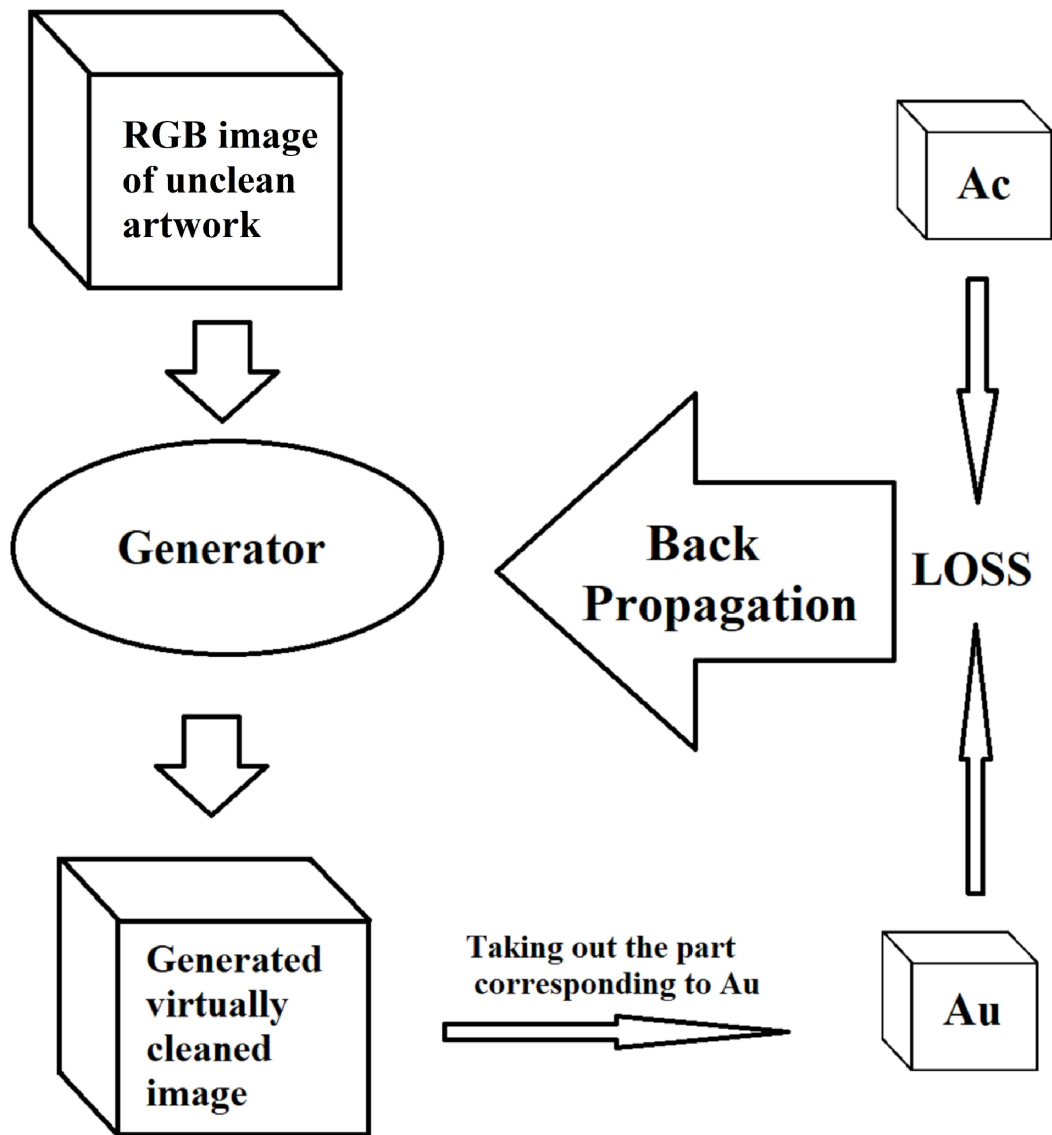


Figure 1: The overall algorithm of the proposed deep generative network. It should be noted that the generator actually takes in the error and based on that, it generates a new image, which would be the virtually cleaned image. There is no training in the traditional sense here, and the generator only learns to clean the whole image using the error it is computing based on the cleaned parts.

2. $C_u^{(1)}(i)$ and $C_u^{(2)}(i)$, similar to the down-sampling block, can be set to different kernel sizes and different numbers of filters shown as $k_u^{(1)}(i)$, $k_u^{(2)}(i)$, $n_u^{(1)}(i)$ and $n_u^{(2)}(i)$, respectively.

The skip connection, shown as $s^{(i)}$, is also utilized to connect the down-sampled data to the up-sampled data (the up-sampling and down-sampling blocks are symmetrical), so the residual

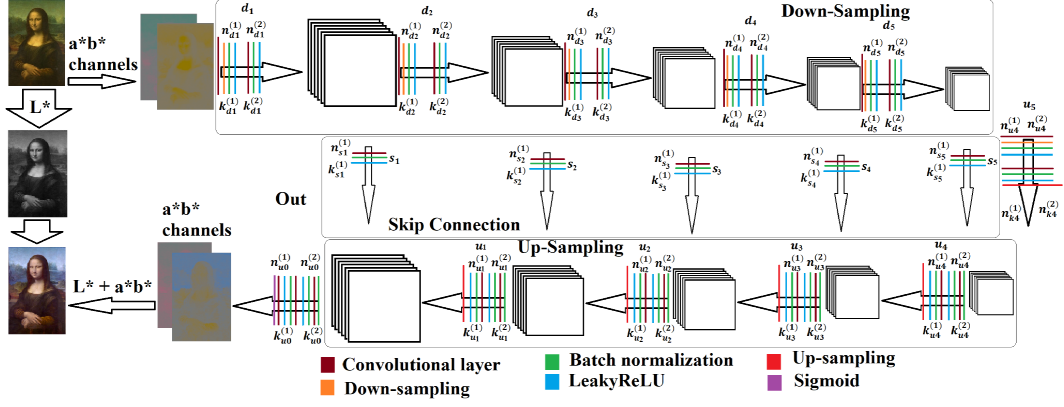


Figure 2: The architecture of the work along with how the input and output are processed.

information can be fully employed. $o^{(0)}$ denotes the output block. It is indeed the up-sampling block that is modified so that the up-sampling layer is superseded with one convolutional layer which is followed by one Sigmoid activation layer.

The network has an hourglass architecture as shown in Fig. 2. Each down-sampling and up-sampling sections are comprised of 5 blocks and 5 skip connections. The filter size is 3×3 in the up-sampling and down-sampling blocks but it is 1×1 in the last convolutional layer. There are 128 filters in the convolutional layers in the down-sampling and up-sampling blocks and there are only 2 (to be equal to the a^*b^* channels) filters in the last convolutional layers. As it was mentioned, only a^*b^* channels of the image Z are input into the network. The output from the network is also the a^*b^* of the image X . This output will be combined with the L^* channel of the image Z that was first set aside, constructing the CIELAB image of output X . The CIELAB image is then converted back into RGB image following standard formulae for sRGB.

As mentioned, the input to the network is the a^*b^* image of the uncleaned artwork Z and the generated image is X . The cost function is defined as the pixel-wise difference between A_u and A_c . A_u belongs to Z and therefore, it changes in each iteration. Consequently, the cost function is given as

$$\min \|A_u - A_c\|^2 \quad (1)$$

It should be noted that the input to the model should be replaced with the output of the model after each iteration. The overall algorithm is shown in Algorithm 1.

2.3. Evaluation Metrics and Experimental Environment

Visualization of the results, per-pixel spectral Euclidean Distance (ED) and Spectral Angle (SA), between the original (cleaned) image and the virtually cleaned image, are the metrics used in this work for accuracy evaluation[20]. The color space used is RGB and each pixel is considered a vector in this space, with the vector tip located at a particular point in the color space according to the RGB values. The spectral Euclidean distance is obtained through calculating the Euclidean distance between two pixels in that color space. The spectral angle is calculated between two

Algorithm 1 Deep Generative Network Algorithm

Procedure: Virtual Cleaning (A_c)

Input: a*b* image of the uncleaned artwork Z

while epoch < max_epoch **do**

$X = Model(Z)$ (Model here stands for the deep generative model.)

$A_u = X$ (The part of the X corresponding to A_c is taken out)

$min\|A_u - A_c\|^2$

$Z = X$ (replace the input with the output of the model in each iteration)

end while

Return X

End Procedure

vectors and is reported in radians in the range $[0, 3.142]$, defined as

$$SA_k = \cos^{-1} \left(\frac{\mathbf{t}_k \cdot \mathbf{r}_k}{|\mathbf{t}_k| |\mathbf{r}_k|} \right) \quad (2)$$

where k denotes the k^{th} pixel, \mathbf{t}_k and \mathbf{r}_k denote the two pixels belonging to the test and reference images, and SA_k denotes the spectral angle between these two pixels.

Python 3.9.7 |Anaconda, Inc. is used as a base coding environment for the DGN algorithm. More specifically, the DGN codes were written and run in the TensorFlow environment, which was installed onto the Anaconda. In terms of hardware, the programs are run on a GPU (NVIDIA GeForce MX350). The training of the DGN is performed using only one image and is consequently referred to as an unsupervised learning method [18]. As mentioned before, only a small area of the image is used to compute the loss function, and the same loss is then used for the whole image to virtually clean it. 1500 epochs are used to train the model. MATLAB R2022a, the package of mathematical software was also used for evaluation computations, making the Macbeth ColorChecker and yellowing it.

3. Results and Discussions

In this section, the results of applying DGN to virtually clean the Macbeth ColorChecker and the *Mona Lisa* are presented and examined. First, we consider the Macbeth ColorChecker.

The Macbeth ColorChecker was simulated as varnished and unvarnished and is used to test the approach, similarly to previous work by [17]. The Macbeth ColorChecker has 24 different color patches, including a range of neutral samples. As mentioned, the DGN needs only a small area of the painting to be physically cleaned and then, using that small part to learn the transfer function describing the varnish effect, the whole painting is virtually cleaned. Given that the Macbeth ColorChecker has different color patches, we empirically identified that the number of patches necessary to be physically cleaned is at least three. Therefore, we applied the method, using the following combination of three patches: a) red, green and blue, b) black and white and a neutral patch, and c) all of the neutral patches, i.e., six neutral patches that exist on the standard Macbeth ColorChecker. The combination in c obviously contains more than three

patches, but is presented as an alternate approach to training the network for testing. The results are visually compared to the method proposed by [17], as shown in Fig. 3 and quantitatively compared in Table 1.

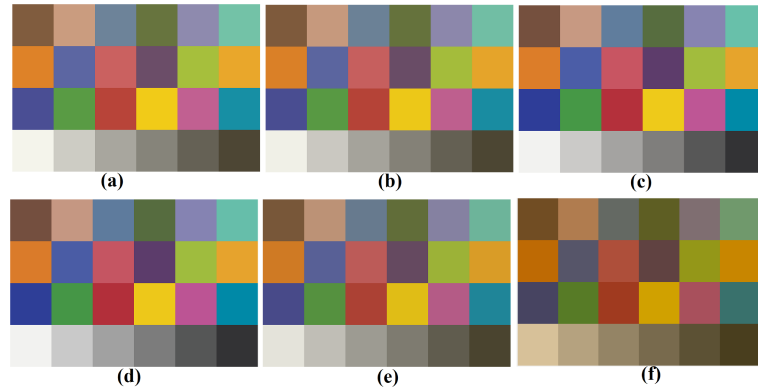


Figure 3: a) all neutral patches, b) black and white and a neutral patch, c) CNN output, d) original Macbeth, e) red, green and blue patches and f) unclean (i.e., yellow) Macbeth.

We observe that the DGN has done an acceptable job compared to the CNN proposed by [17], even though the number of training samples required by the DGN is significantly smaller than that of the CNN. To have a better understanding of the results, Table 1 shows the quantitative results in terms of the mean values of ED and SA for the whole ColorChecker. These metrics are computed between the virtually cleaned color chart and the original one.

Table 1

Euclidean distance and SA mean and standard deviation (SD) values between the original and virtually cleaned Macbeth color chart.

Method	Euclidean distance		SA	
	Mean	SD	Mean	SD
All neutral patches	0.06	0.022	0.06	0.021
Black, white and a neutral patch	0.056	0.026	0.062	0.024
Red, green and blue patches	0.074	0.042	0.070	0.034
CNN proposed by [17]	0.021	0.002	0.014	0.004

As it is observed from Table 1, the CNN model has done a slightly better job in terms of cleaning the Macbeth ColorChecker. This is not too concerning as the method proposed herein is more practical than the CNN proposed by [17]. The DGN proposed herein only needs a small area of the painting to be cleaned, while the CNN needs a significantly larger number of training samples to work. While the end goal of each approach is the same, a virtually cleaned work of art, the operational aspects of the two methods are significantly different.

Finally, we also applied the DGN to clean the *Mona Lisa*. The results are shown in Fig. 4.

Fig. 4 (c) shows the area of the painting that was used to compute the loss; in other words, that area is used to train the network to go from the unclean to the clean version of the artwork. As shown in Fig. 4 (e), the DGN has again done a visually acceptable job of cleaning the artwork, considering that the area of the painting used to train the network is fairly small. The ED and

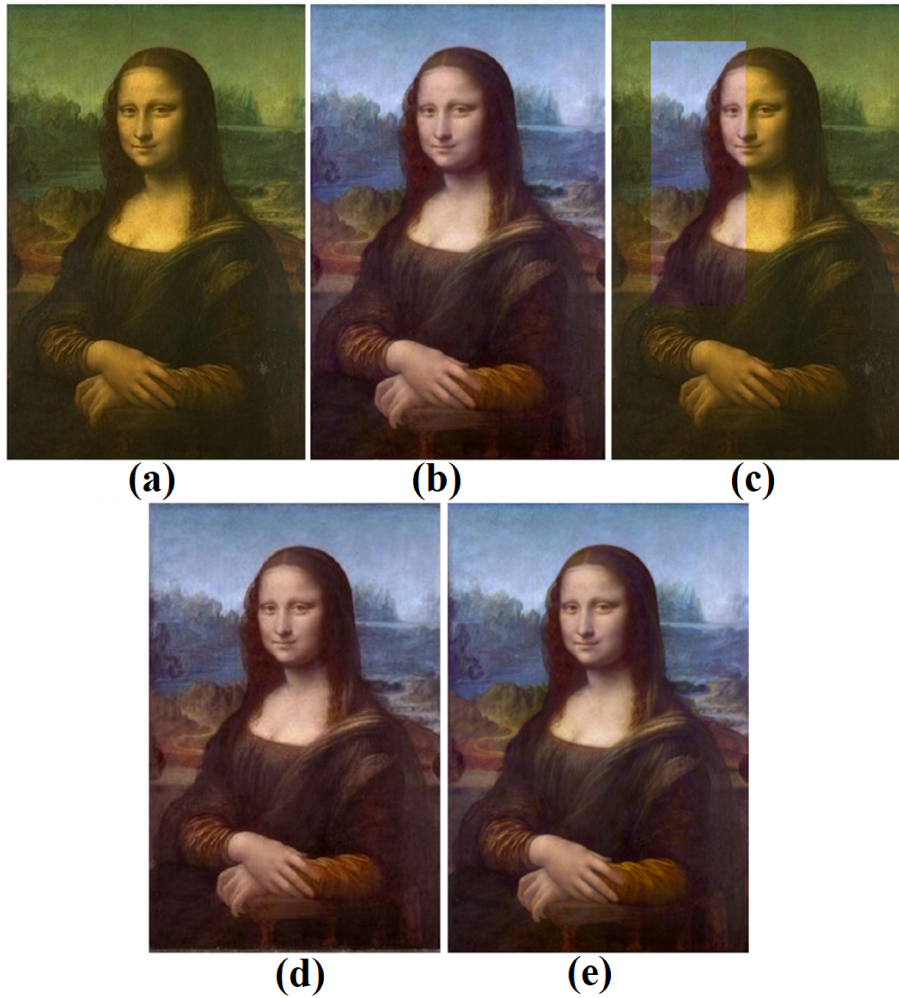


Figure 4: a) Unclean *Mona Lisa*, b) original clean *Mona Lisa*, c) The area of *Mona Lisa* that is assumed to be physically cleaned, d) virtually cleaned using CNN proposed by [17], e) virtually cleaned using DGN.

SA are also computed between the original clean *Mona Lisa* and the virtually cleaned one. The results are both visualized (Fig. 5) and reported in terms of the mean values across the whole image (Table 2). The visualization of the ED and SA values show specific areas of the work that are not well cleaned (note that in Figure 5 all four results are normalized to 1). To better understand the absolute performance, the mean values of the ED and SA are also reported which clarifies which method has outperformed the other. As it is observed from Fig. 5, the CNN has not done a good job especially predicting the cleaned color of the sky, and overall the error is higher and more widespread in the CNN.

We see from Table 2, the proposed method here has surprisingly outperformed the CNN proposed by [17]. It is surprising as the CNN outperformed our proposed method when the Macbeth ColorChecker was the object of interest, but the results here are the opposite in the case of the *Mona Lisa*. This could be because of the richness of the colors and structural features

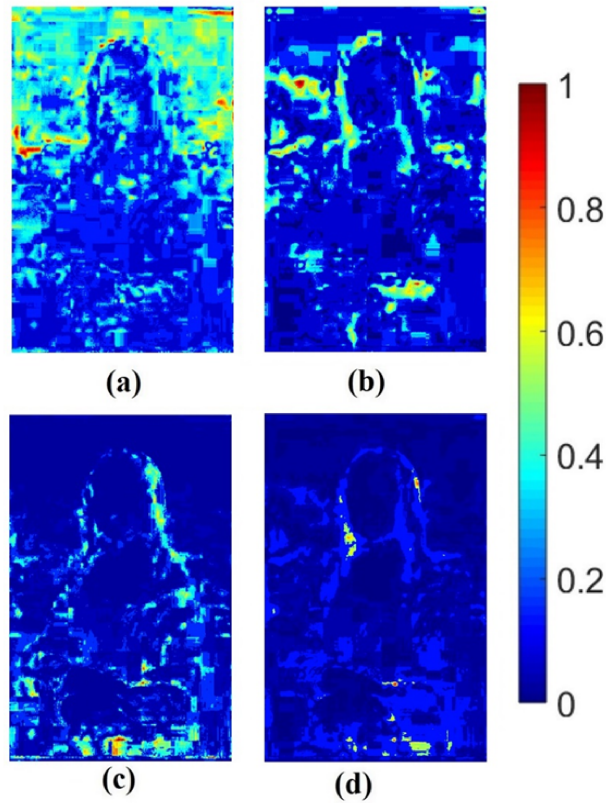


Figure 5: a) ED calculated between the original clean *Mona Lisa* and the virtually cleaned one using CNN, b) ED calculated between the original clean *Mona Lisa* and the virtually cleaned one using DGN, c) SA calculated between the original clean *Mona Lisa* and the virtually cleaned one using CNN, d) SA calculated between the original clean *Mona Lisa* and the virtually cleaned one using DGN.

Table 2

Euclidean distance and SA mean and standard deviation (SD) values between the original and virtually cleaned *Mona Lisa*.

Method	Euclidean distance		SA	
	Mean	SD	Mean	SD
DGN	0.0167	0.0015	0.1045	0.0139
CNN [17]	0.0371	0.0024	0.1489	0.0209

that are present in the *Mona Lisa*, as opposed to the Macbeth ColorChecker, which is a simple color chart. This would also confirm that the method proposed herein is more practical than CNN, as asserted above. The method proposed herein has a potential of being applied to a wider type of artworks compared to the CNN, which requires a large set of training data with content similar to the artwork itself.

It is important to note that the small area chosen in the artwork should be a representative of all the features and material present on the painting. Looking at Figure 4 (c), one could see that the small area contains a small part of the sky, human eye and skin, and her dress. This will

strengthen the performance of the DGN. To examine this point further, another experiment is performed in which, the small area varies from what has been chosen in Figure 4 (c). The new small area only comprises the person (part of her face, her dress, her hair and her skin) as shown in Fig. 6. In this Figure, the top row shows the results of Figs 4 and 5 combined in the case of the DGN and the bottom row shows the results of the DGN when a different and smaller area is chosen.

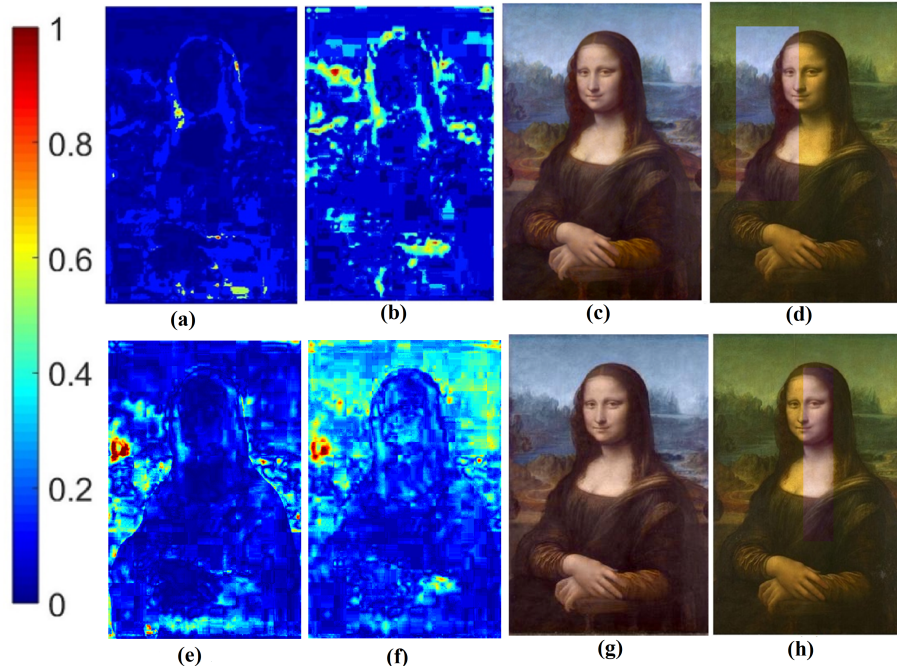


Figure 6: First experiment: a) SA calculated between the original clean *Mona Lisa* and the virtually cleaned one using DGN, b) ED calculated between the original clean *Mona Lisa* and the virtually cleaned one using DGN, c) virtually cleaned using DGN, d) The area of *Mona Lisa* that is assumed to be physically cleaned. Second experiment: e) SA calculated between the original clean *Mona Lisa* and the virtually cleaned one using DGN, f) ED calculated between the original clean *Mona Lisa* and the virtually cleaned one using DGN, g) virtually cleaned using DGN, h) The area of *Mona Lisa* that is assumed to be physically cleaned.

As seen from the bottom row of Fig. 6, the sky and everything around the person, has not been cleaned as well as the top row where the area chosen is a better representative of everything in the image. It is also worthwhile noting that the DGN has not done terribly, however, with a more thorough area, it could lead to a better result.

4. Conclusions

In this work, we developed a Deep Generative Network (DGN) to tackle the problem of virtual cleaning of artwork for visualization. We compared our method to the latest method in this area which used a Convolutional Neural Network (CNN). We used the Macbeth ColorChecker and the *Mona Lisa* to test our method. We found that the proposed model did not outperform

the CNN in the case of the Macbeth ColorChecker, but it did outperform the CNN in the case of the *Mona Lisa*. This shows the high potential of the work proposed herein to be applied in the real case and to a wider range of artworks. The method proposed herein could potentially help the conservators with seeing how a painting would look if it were to be physically cleaned, or aid them in choosing from different options they have for a physical cleaning and so forth.

5. Acknowledgments

This research was funded by the Xerox Chair in Imaging Science in the Chester F. Carson Center for Imaging Science at the Rochester Institute of Technology.

References

- [1] S. Constantin, The barbizon painters: a guide to their suppliers, *Studies in conservation* 46 (2001) 49–67.
- [2] A. Callen, The unvarnished truth: Mattness, 'primitivism' and modernity in french painting, c. 1870-1907, *The Burlington Magazine* 136 (1994) 738–746.
- [3] R. Bruce-Gardner, G. Hedley, C. Villers, *Impressionist and post-impressionist masterpieces: The courtauld collection*, 1987.
- [4] M. Watson, A. Burnstock, An evaluation of color change in nineteenth-century grounds on canvas upon varnishing and varnish removal, in: *New Insights into the Cleaning of Paintings: Proceedings from the Cleaning 2010 International Conference*, Universidad Politecnica de Valencia and Museum Conservation Institute, Smithsonian Institution, 2013.
- [5] L. Baij, J. Hermans, B. Ormsby, P. Noble, P. Iedema, K. Keune, A review of solvent action on oil paint, *Heritage Science* 8 (2020) 1–23.
- [6] S. Prati, F. Volpi, R. Fontana, P. Galletti, L. Giorgini, R. Mazzeo, L. Mazzocchetti, C. Samorì, G. Sciutto, E. Tagliavini, Sustainability in art conservation: a novel bio-based organogel for the cleaning of water sensitive works of art, *Pure and Applied Chemistry* 90 (2018) 239–251.
- [7] E. Al-Emam, H. Soenen, J. Caen, K. Janssens, Characterization of polyvinyl alcohol-borax/agarose (pva-b/ag) double network hydrogel utilized for the cleaning of works of art, *Heritage Science* 8 (2020) 1–14.
- [8] M. El-Gohary, Experimental tests used for treatment of red weathering crusts in disintegrated granite–egypt, *Journal of cultural heritage* 10 (2009) 471–479.
- [9] D. Gulotta, D. Saviello, F. Gherardi, L. Toniolo, M. Anzani, A. Rabbolini, S. Goidanich, Setup of a sustainable indoor cleaning methodology for the sculpted stone surfaces of the duomo of milan, *Heritage Science* 2 (2014) 1–13.
- [10] M. Barni, F. Bartolini, V. Cappellini, Image processing for virtual restoration of artworks, *IEEE multimedia* 7 (2000) 34–37.
- [11] M. Pappas, I. Pitas, Digital color restoration of old paintings, *IEEE Transactions on image processing* 9 (2000) 291–294.
- [12] M. Elias, P. Cotte, Multispectral camera and radiative transfer equation used to depict leonardo's sfumato in mona lisa, *Applied optics* 47 (2008) 2146–2154.

- [13] C. M. T. Palomero, M. N. Soriano, Digital cleaning and “dirt” layer visualization of an oil painting, *Optics express* 19 (2011) 21011–21017.
- [14] G. Trumpy, D. Conover, L. Simonot, M. Thoury, M. Picollo, J. K. Delaney, Experimental study on merits of virtual cleaning of paintings with aged varnish, *Optics express* 23 (2015) 33836–33848.
- [15] E. Kirchner, I. van der Lans, F. Ligterink, E. Hendriks, J. Delaney, Digitally reconstructing van gogh’s field with irises near arles. part 1: varnish, *Color Research & Application* 43 (2018) 150–157.
- [16] J. Linhares, L. Cardeira, A. Bailão, R. Pastilha, S. Nascimento, Chromatic changes in paintings of adriano de sousa lopes after the removal of aged varnish, *Conservar Património* 34 (2020) 50–64.
- [17] M. Maali Amiri, D. W. Messinger, Virtual cleaning of works of art using deep convolutional neural networks, *Heritage Science* 9 (2021) 1–19.
- [18] J. M. Haut, R. Fernandez-Beltran, M. E. Paoletti, J. Plaza, A. Plaza, F. Pla, A new deep generative network for unsupervised remote sensing single-image super-resolution, *IEEE Transactions on Geoscience and Remote sensing* 56 (2018) 6792–6810.
- [19] D. Ulyanov, A. Vedaldi, V. Lempitsky, Deep image prior, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9446–9454.
- [20] B. Park, W. Windham, K. Lawrence, D. Smith, Contaminant classification of poultry hyperspectral imagery using a spectral angle mapper algorithm, *Biosystems Engineering* 96 (2007) 323–333.