Virtual restoration of paintings using adaptive adversarial neural network

Roman Sizyakin[a], Viacheslav Voronin[b], Aleksandr Zelensky[b], Aleksandra Pižurica[a]
[a]Ghent University, TELIN-GAIM, Department Telecommunications and Information Processing, Sint-Pietersnieuwstraat 25, Ghent, Belgium, 9000
[b]Moscow State University of Technology "STANKIN", Center for Cognitive Technologies and Machine Vision "Digital Technologies of Mechanical Engineering", Vadkovsky 1, Moscow, Russia, 127055

Abstract.

Over time, the visual quality of the paintings deteriorates. Cracks and loss of paint are the main types of damages that worsen the visual component of the painting. One of the ways to return the authentic appearance of paintings is a virtual restoration. Virtual restoration consists of two main stages: detecting deterioration and their removal. In this research, we investigate the possibility of applying deep learning-based methods for virtual restoration. To detect cracks we use a combination of convolutional (MCN) and autoencoder neural networks based on U-NET architecture, and to remove them, an adaptive adversarial network (aGAN). Also, in this work, we propose an original way of training an adversarial neural network, which allows us to apply it more successfully in practice. A series of experiments shows encouraging results compared to known methods and confirms the high efficiency of deep learning.

Keywords: Virtual restoration of paintings, crack detection, segmentation, deep learning, convolutional neural network, U-Net, adaptive adversarial neural networks.

1 Introduction

Virtual restoration is often the only plausible way to restore the original appearance of master paintings. Over time, aging and various kinds of deterioration dominantly crack, and paint losses become inevitably affected. In physical restoration treatments, painting cracks are typically left untouched unless at places where more severe painting losses are present. Although this conservation practice secures the authenticity of paintings, the aging cracks still reduce the overall quality of visual perception and may hinder full appreciation of the artist's original content.

In this paper, we will focus on detecting and virtually inpainting cracks. Accurate automatic crack detection can provide invaluable support to art restorers, facilitating an objective insight into the current state of the painting and the evolution of deteriorations over time. Moreover, virtual inpainting serves as a simulation to support the decisions that need to be made during the actual restoration process.

This paper focuses on the problems associated with the virtual restoration of paintings using adaptive adversarial neural networks. The primary contributions of our paper include a novelty:

1) The method for virtual restoration of paintings using deep learning to detect cracks and their removal.

2) Fusion of two neural network models for cracks detection: convolutional and U-Net segmentation neural network.

3) The adaptive feedback through the trend estimation coefficient for the adversarial network for a higher-quality reconstruction result of sharpness and the global structure. The coefficient allows for the dynamic evaluation of the loss function trend in the learning process for adaptive balancing of the loss function.

The paper is organized in the following manner: Section II presents the image processing of the painting's background information. Section III defines a virtual restoration of paintings algorithm using deep learning. Section IV presents some experimental results of crack detection and removal. Finally, Section V gives some concluding comments.

2 Related work

The earlier cracks detection methods are based on simple thresholding.[1] In this case, the threshold value is chosen using a histogram to divide pixels belonging to the defected region from the undamaged pixels. In papers,[2] a modification of the thresholding method was proposed based on an adaptation of the threshold value. The main drawback of the threshold-based methods is a dependence of correct detection on a threshold value.

Crack detection methods often employ morphological filtering as top-hat transform, region-growing algorithm, erosion, and dilation with the pre-selected structural element[3,4] due to its low computational complexity and high "Recall" metric. However, the detected crack maps typically contain many false positives, and therefore morphological filtering is rarely used as an independent crack detection method but rather as a preprocessing step. The computational complexity of more advanced techniques can be significantly reduced with a practical preprocessing step that eliminates large areas where painting cracks are absent.

Some of the methods are based on a combination of texture analysis (Gabor filtering, Markov random field) and morphological processing.[5] These methods require a priori information about the threshold value and parameters of algorithms.

a) b) c)

Fig 1: Part of *Annunciation to virgin Mary* panel from the *Ghent Altarpiece*, a) Color image, b) Infrared image, c) X-Ray image

Another group of crack detection methods is based on the processing in the frequency domain.[6] There are still some unsolved problems in this group of methods, such as properly selecting a system of basis functions and detecting a crack on a texture having a similar brightness.

Most of the current crack detection methods are based on machine learning. A Bayesian approach[7,8] form feature vectors from the available image modalities and applies Bayesian Conditional Tensor Factorizations (BCTF) classifier .[9] The functional imaging modalities often include optical macrophotography, infrared macrophotography, infrared reflectography, and X-ray images (Figure 1). Other modalities, like macro-X-ray fluorescence or hyperspectral images, are worked in some cases, but these are still relatively rare as they require expensive equipment. The available imaging modalities are sometimes expanded artificially, creating virtual modalities, e.g., by applying various filters. The corresponding set of filters is typically optimized for each processed painting, which poses limitations in practice.

In general, the main problem with the existing multimodal crack detection approaches is their low resistance to inter-modal shifts, which leads to an increase in false-positive responses. The difficulties arising from intermodal shifts can be alleviated by using patch-based convolutional neural networks (CNN).[10,11] By operating on small image patches, the convolutional neural network can effectively use both spatial and intermodal correlation to improve the crack detection accuracy and improve the robustness to intermodal shifts. Most importantly, as with all deep learning methods, we now enjoy the advantage of not having to hand-engineer any filters. The feature maps are now automatically synthesized inside the network during the training process. However, these methods yield excessive thickening of the actual crack boundaries.[12–14] A possible solution to this problem is a combination of patch-based and vector-based techniques.[15] However, this approach does not permanently eliminate the problem of false crack thickening. Additionally, there is uncertainty with the choice of the patch size, which must be selected for each processed painting individually.

More precise classification (with pixel-level precision) can be achieved with segmentation convolutional autoencoders and modifications.[16–19] Such neural network architectures receive an entire image as input data and output a segmentation map with pixel-level precision. During the training process, the filters of such an autoencoder adapt to texture features that can be linked/combined into a local group, for example, by color or texture features. Those texture areas of the image that cannot be linked/combined into a local group are smoothed. In the expansion process (deconvolution), they are ignored on the resulting segmentation map. The main disadvantage of such networks is a complex learning process requiring many labeled training samples. Also, in some cases, this type of neural network may require a significant amount of time for training or may not converge at all due to poor-quality labeling of training data.

In the case of virtual restoration, the detection of cracks is only the first stage. The second stage is virtual restoration (inpainting) of the areas detected in the first stage. The simplest way to fill in the damaged areas is the usual polynomial interpolation of undamaged boundary pixels. This group of methods includes the work in which the Navier-Stokes equations are used as an interpolating function.[20] This method can be helpful if the fill rate is a priority requirement. However, if the area to fill is extensive, the absence of texturing of the filled area can be a significant disadvantage. Methods based on the search of self-similar patches on an entire area of the image cope with this problem more successfully. After that, the found patches are used to reconstruct the damaged area.[21–23] The most difficult cases for this group of methods are cases when the lost area includes a semantically important object in the image. Semantically important areas can include, for example, the wheels of a car, the windows of a house, or the mouth and eyes on the face. Such areas cannot be restored using this group of methods because undamaged areas may not contain duplicates of such semantically important objects. Reconstructing variational autoencoders (VAE)[24] and adversarial neural networks (GAN)[25–27] can partially and in some cases completely solve this problem. The main advantage of generating neural networks is restoring areas containing important semantic information, even if duplicates of such areas are partially or completely absent in undamaged areas of the image. The ability to restore such image areas is achieved during training ("memorization"), using training images. Subsequently, these neural networks use parts of these "memorized" training images to fill in the damaged areas in the reconstruction model. The main disadvantage of VAE generating methods is the blurriness of the reconstructed area, while GANs suffer from an unstable training process.

In this research, we investigate the possibility of virtual restoration of paintings using a combination of convolutional (MCN) and autoencoded neural network based on U-NET architecture for detecting cracks, as well as a novel adaptive adversarial network (aGAN) for removing detected cracks.

3 Proposed method

Cracks in the paintings are dark or light elongated curves with a complex shape. The main difficulty for detecting cracks is their similarity to some textural features found in paintings, for example: brush strokes, hair, complex painted patterns, etc. Due to the impossibility to distinguish such objects from cracks, often even with visual analysis, in the tasks of virtual restoration, images in the infrared, X-ray and other wave ranges are used as an addition to the main color image. In our work, we use multimodal acquisitions of the *Ghent Altarpiece*[28][1].

The challenge of detecting cracks is to construct a binary map on which the cracks are marked with value 1, and the undamaged areas are marked with 0. The input image $Y_{h,v}$ can be represented as:

$$Y_{h,v} = (1 - d_{h,v}) \cdot S_{h,v} + d_{h,v} \cdot c_{h,v} \tag{1}$$

where h, v are the spatial coordinates, $S_{h,v}$ is the undamaged content, $d_{h,v} \in \{0, 1\}$ a binary crack map of defects, and $c_{h,v}$ is the crack color.

3.1 Crack detection

To create a crack map, we combine the results from two different neural network models: a segmenting autoencoder U-Net based[17] and a convolutional neural network MCN.[11] The architecture of this hybrid network is illustrated in Figure 2.

[1]Image Gallery: Closer to Van Eyck, Rediscovering the Ghent Altarpiece, http://closertovaneyck.kikirpa.be/

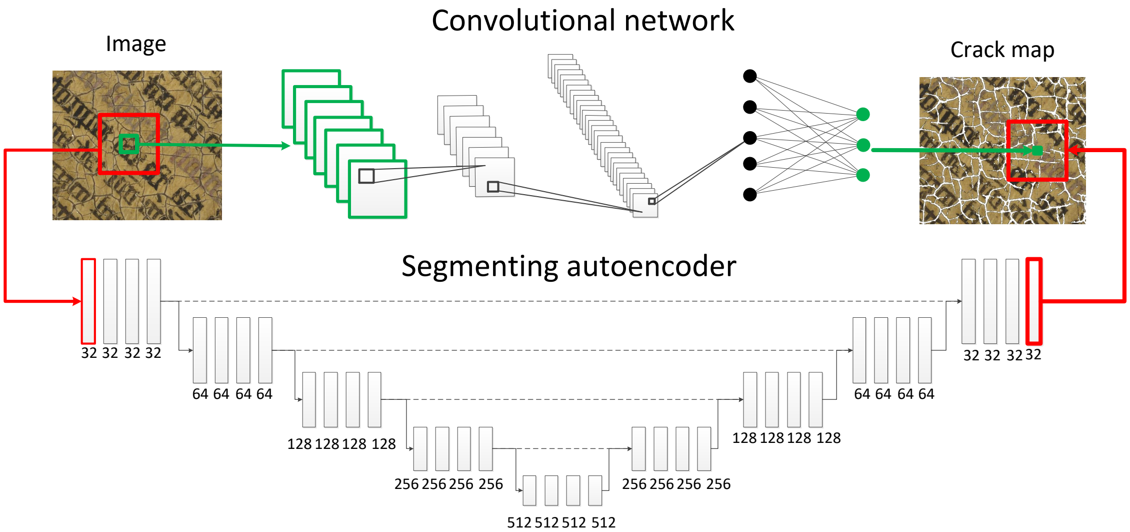


Fig 2: The proposed architecture of the combining segmenting autoencoder and convolutional neural network.

All convolutional layer for autoencoder and convolutional network are based on the operation of N-dimensional convolution of input data and filters. Equation for this operation can be defined as:

$$x_{h,v}^{l,c} = f(\sum_{h}\sum_{v}\sum_{c} x_{h+m,v+n}^{l-1,c} \cdot k_{h,v}^{l,c} + b), \tag{2}$$

where $x_{h,v}^{l,c}$ is the feature map at layer l from modality c, $k_{h,v}^{l,c}$ is the corresponding convolution kernel, $x_{h+m,v+n}^{l-1,c}$ is the feature map from the previous layer, f is the activation function of the hidden layer, and b is a bias.

The training process consists in setting up the filters for convolution so that when the input data passes through all the layers of the neural network, the loss function is minimal. For convolutional neural network we use the binary cross-entropy function, defined as:

$$Loss(y_\kappa, y'_\kappa) = -\frac{1}{\mathcal{K}}\sum_{\kappa=1}^{\mathcal{K}}\left[y_\kappa \cdot log(y'_\kappa) + (1 - y_k) \cdot log(1 - y'_\kappa)\right] \tag{3}$$

where y' is the label predicted by our classifier, and y is the ground truth label.

For autoencoder we use Sörensen–Dice coefficient[29,30] for loss estimation, which shows the measure of the area of correctly marked segments and can be defined as:

$$Loss = \frac{2|x \cap d|}{x + d} \tag{4}$$

where x and d - is estimated and ground truth crack maps, respectively.

The architecture of autoencoder has following parameters: $C^0 5$, $C^1 32$, $C^2 32$, $C^3 32$, $C^4 32$, MP^5, $C^6 64$, $C^7 64$, $C^8 64$, $C^9 64$, MP^{10}, $C^{11} 128$, $C^{12} 128$, $C^{12} 128$, $C^{14} 128$, MP^{15}, $C^{16} 256$, $C^{17} 256$, $C^{18} 256$, $C^{19} 256$, MP^{20}, $C^{21} 512$, $C^{22} 512$, $C^{23} 512$, $C^{24} 512$, US^{25}, $C^{26} 256$, $C^{27} 256$, $C^{28} 256$, $C^{29} 256$, US^{30}, $C^{31} 128$, $C^{32} 128$, $C^{33} 128$, $C^{34} 128$, US^{35}, $C^{36} 64$, $C^{37} 64$, $C^{38} 64$, $C^{39} 64$, US^{40}, $C^{41} 32$, $C^{42} 32$, $C^{43} 32$, $C^{44} 32$, $C^{45}(sigm)3$ where C^h - denotes a convolutional layer with index h, digit after C^h denotes a number of feature maps for current layer, MP^h - Max-pooling operation, US^h - Up-sampling operation and $(sigm)$ is denote logistic sigmoid activation function. All other layers use the exponentially linear unit (ELU)[31] as activation function, which is a more efficient version of the activation function ReLU[32,33] and Leaky ReLU,[34] and allows to achieve convergence of the neural network faster and higher accuracy, as well as exclude the process of batch normalization.[35] The equation can be written as:

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ a(e^x - 1) & \text{if } x \leq 0, \end{cases} \tag{5}$$

where $a > 0$ is a hyperparameter that controls the value at which the ELU saturates for negative inputs.

Convolutional network has following layer parameters: $C^0 5$, $C^1 100$, MP^2, $C^3 200$, MP^4, $C^5 300$, $FC^6 300$, $FC^7(softmax)$, where FC - denotes a fully connected layer. In final layer has used "softmax" activation function:

$$y(z_\iota) = \frac{e^{z_\iota}}{\sum_\kappa e^{z_\kappa}}, \tag{6}$$

All convolutional layers for both networks have a spatial filter size of 3×3 pixels. For training, the optimization of Adam[36] was used with a learning rate of 0.00002. Additionally, it should be noted that the convolutional network has the spatial size of the input tensor $8 \times 8 \times 5$ pixel, while

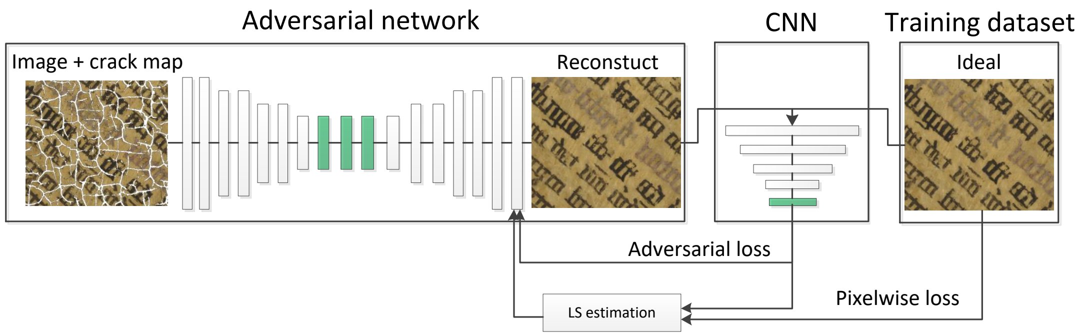


Fig 3: The proposed GAN-based model for virtual restoration.

the autoencoder uses the tensor $20 \times 20 \times 5$ pixels as input. The resulting crack map is formed using the logical operator "and", which combines predictions from two neural networks.

3.2 Crack removal

For the virtual restoration of paintings, we use a generative adversarial neural network.[37] This network usually includes at least 2 neural networks: a generative network based on an auto-encoder and a discriminative network based on a convolutional neural network. The two networks are set up in an adversarial style. This means that with the improvement of the results of one network, the opposing network will receive more losses, and vice versa. The key advantage of such a network is sharper generated images, in comparison with an autoencoder that uses pixel-by-pixel difference as a loss function. The disadvantages of such an architecture include an unstable training process. This means that the network may not converge if one of the networks included in the GAN learns earlier than the opposing one. The architecture of proposed the adaptive generating adversarial neural network that we use is illustrated in Figure 3.

The reconstructing network has the following architecture: $C^{0}4$, $C^{1}64$, $C^{2}64$, $C^{3}64$, $C^{4}64$, MP^{5}, $C^{6}128$, $C^{7}128$, $C^{8}128$, $C^{9}128$, MP^{10}, $C^{11}256$, $C^{12}256$, $C^{12}256$, $C^{14}256$, US^{15}, $C^{16}128$, $C^{17}128$, $C^{18}128$, $C^{19}128$, US^{20}, $C^{21}64$, $C^{22}64$, $C^{23}64$, $C^{24}64$, $C^{25}(sigm)3$ and global discriminator: $C^{0}4$, $C^{1}64$, $C^{2}64$, $C^{3}64$, MP^{4}, $C^{5}128$, $C^{6}128$, $C^{7}128$, MP^{8}, $C^{9}256$, $C^{10}256$, $C^{11}256$, $FC^{12}(sigm)3$, where FC^{h} - denotes a fully connected layer with logistic sigmoid activation function. As input data for the layer $C^{0}4$, a color image with a randomly deleted area is used together with a binary mask of the deleted area [2]. All convolutional layers of the generating and discriminating networks use an exponentially linear unit (ELU) as an activation function.

The loss function for reconstructing network is determined according to the equation:

$$Loss_G = L_{adv} + \lambda L_{abs} \cdot |\alpha|, \tag{7}$$

$$L_{abs} = |x_{trn} - G(x_{def})|, \tag{8}$$

$$L_{adv} = \mathbb{E}[\log(1 - D(G(x_{def})))] \tag{9}$$

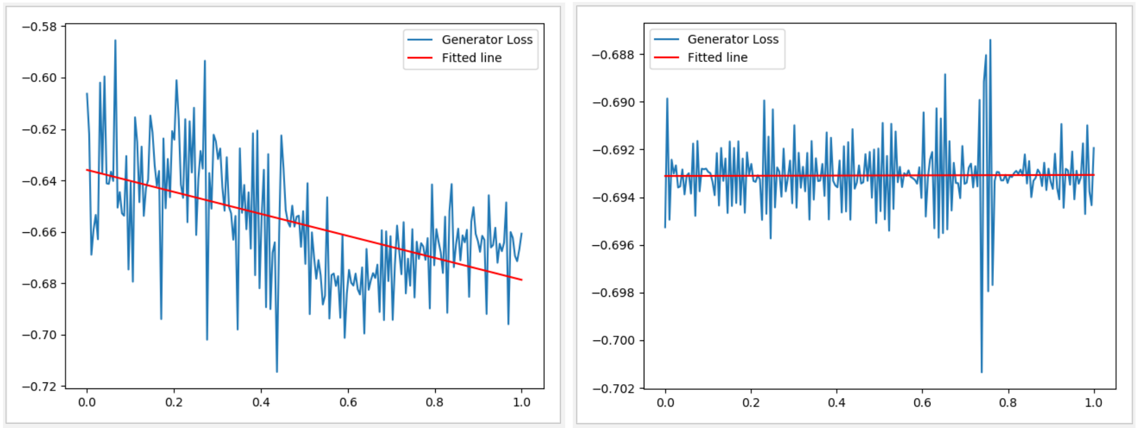
$$\alpha, \beta = LS(Loss_G) \tag{10}$$

where x_{trn} - undamaged image for training, $G(x_{def})$ - reconstructed image, λ - coefficients of proportionality, which is used to align the loss order, α and β - is approximation coefficient for first-oder polynomial obtained by lest squares method.

At this stage, we introduce the tilt coefficient of the approximating curve α, which allows us to estimate the trend dynamics of the loss function. Depending on the value of this curve, we adjust the weight of the pixel-by-pixel loss to make the restored area a more sharp. This coefficient makes it possible to achieve a tradeoff between sharpness and structural accuracy of the reconstructed area. Figure 4(a) and 4(b) show two cases of trend estimation, for starting and finising moment of training.

The figures show that at the initial moment of training, the red curve has a significant slope, so the losses from the pixel-by-pixel difference have a significant weight, while at the final stage of training, the slope of the curve tends to zero, which in turn leads to a decrease in the impact of the pixel-by-pixel loss. The L_{adv} error allows to achieve a higher sharpness of the reconstructed area and the L_{abs} loss allows to achieve a more stable learning process.

[2]To form a binary mask, a random section from the full map of cracks obtained at the crack detection stage is used



a) b)

Fig 4: Example of loss function trend estimation using first-oder approximation, a)Trend estimation for the starting moment of training, b)Trend estimation for the finising moment of training

The task of the discriminator is to determine which of the images is the original and which is reconstructed. Loss function for discriminator are calculated according to the equation:

$$Loss_D = \mathbb{E}[\log(D(x_{trn}) + \log(1 - D(G(x_{def})))] \tag{11}$$

where x - source image, the size of which depends on what discriminator is used.

This configuration of loss functions leads to a adversary between two neural networks. Since the generator has a larger number of layers, one iteration of the training includes two steps of the generator and one step of the discriminator. Additionally we use RMSProp optimization with a different learning rate of 0.0002 and 0.0001, generator and discriminator, respectively. The training batch includes 100 samples with the size of 24×24 pixels for *Annunciation virgin Mary* panel and 12×12 pixels for *Singing Angels* panel.

Due to the fact that in our work we apply a generating adversarial network to small patches independently, there is a problem of their incoherence at the edges, when combined into a full restored image. This problem is shown in Figure 5.

To solve this problem, we process the full image several times using a small shift of 3 pixels for each iteration of the restoration. For example, if the first time the starting position for processing was the upper-left corner with the beginning of [0,0], then at the second iteration of processing, the starting position will be the value [3,3]. Example of such shift for 0,9 and 18 pixels illustrate in Figure 5(a,b,c) respectively.

a) b) c) d)

Fig 5: An example of the edge coherence problem in the independent processing of small patches of a large image. a,b,c) An example of removing cracks, provided that each subsequent processing begins with a shift of 0, 9 and 18 pixels, respectively, d) The result of combining all the images into one using the median filter.

a)

Fig 6: Example of training dataset for crack detection

Since we use the patch size of 24×24, we have 8 versions of the restored images in total. After that, the 8 versions of the reconstructed images are combined into one using the median filter. As a result, the final image contains only the pixels that received the highest probability among the 8 images, while the abnormal pixel values are rejected. The result of this operation shown in Figure 5(d)

For form training dataset in this work, we use intact areas between cracks as training data. This decision is explained by the fact that fragments for training are highly correlated with damaged areas that will need to be removed in the future.

4 Experimental results

To assess the quality of the restoration of paintings, we use two paintings *Annunciation virgin Mary* and *Singing Angels* from *Ghent Altarpiece*.[28] These paintings have high resolution and are presented in three modalities: color macrophotograph, infrared macrophotograph and X-ray macrophotograph. Such rich visualization is extremely useful for virtual crack detection, as it allows to get more useful information for classification.

This section is divided into two parts: crack detection and crack removal subsections. To obtain numerical results in the first subsection, we use the following metrics:

$$FA = \frac{FP}{AlPx - DfPx}, \quad FM = \frac{FN}{AlPx - UdPx} \tag{12}$$

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F_1 = \frac{2 \cdot P \cdot R}{P + R} \tag{13}$$

where FA - probability of false alarm, FM - probability of false missing pixels containing cracks, P - precision, R - recall, F_1 - F_1-measure, TP - true positive, FP - false positive, FN - false negative, $DfPx$ - total amount of pixels belonging to a crack, $UdPx$ - total amount of pixels not belonging to a crack, and $AlPx$ - total amount of pixels in the image.

Additionally, as well-known methods for comparison, we use: MCNC method with improved crack boundary localization,[11] Bayesian Conditional Tensor Factorization method (BCTF),[7] CNN-based method that was proposed for crack detection in roads[10] and a deep feature fusion network (DFFN) classifer from.[38] All methods use the same set of training data that is used in the works [Cornelis B. et al.][7] and [Sizyakin R. et al.][11] An example of data with label from this set is illustrated in Figure 6(a) for *Singing angels* panel.

For the second subsection, as well-known methods for comparison, we use: exemplar based method (EBM)[21] and context-aware image inpainting using MRF.[22] Since the paintings *Virgin Annunciate* and *Singing Angels* currently have no intact versions, numerical metrics are not provided.

4.1 Crack detection

The use of crack detection methods based on the U-Net architecture has both advantages and disadvantages. From our previous research, we can note the following advantages of such an architecture: high accuracy of crack localization without excessive expansion of their boundaries, high speed of network learning. The disadvantages we can attribute to the high demands on the quality of the markup of training data. So if the cracks in the training set are not completely marked or inaccurately, then the network may not converge or have a low accuracy of localization of cracks. To solve this problem and also to make the model more applicable in practice, we use an input tensor with a small spatial size of $20 \times 20 \times 5$ pixels. Where 5 corresponds to 3 modalities

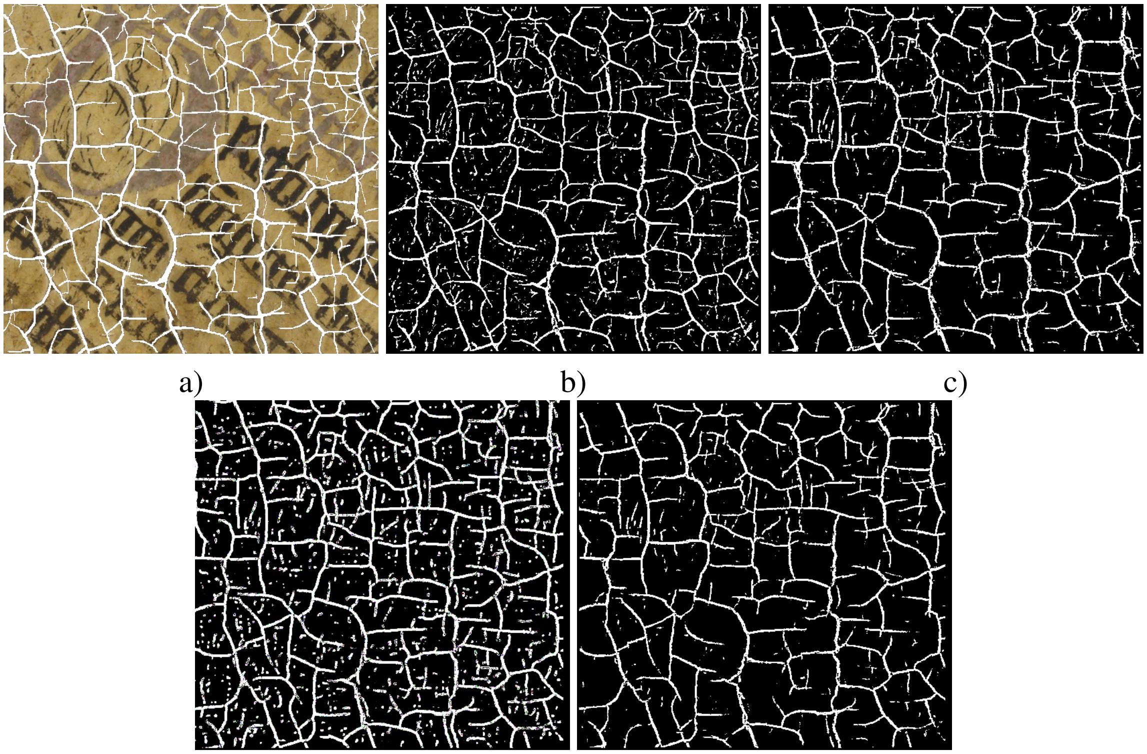


Fig 7: Example of crack detection: a) Part of *Annunciation virgin Mary* panel, b) Crack map of BCTF, c) Crack map of MCNC, d) Crack map of UNET, e) Crack map of UMCNC

Table 1: Comparison of different methods for crack detection on a panel from the *Ghent Altarpiece*.

Annunciation virgin Mary panel					
Method	Recall	False alar.	False miss.	Precision	F_1-m.
CNN[10]	**0.8481**	0.0777	**0.1519**	0.5989	0.7020
DFFN[38]	0.7488	0.0422	0.2512	0.7081	0.7279
BCTF[7]	0.7896	0.0535	0.2104	0.6686	0.7241
MCN[11]	0.8161	0.0540	0.1839	0.6741	0.7383
MCNC[11]	0.7673	0.0375	0.2327	0.7365	0.7516
UNET	0.8356	0.1109	0.1644	0.5076	0.6315
UMCN	0.7928	0.0436	0.2072	0.7134	0.7510
UMCNC	0.7541	**0.0320**	0.2459	0.7630	**0.7585**

of the color and 2 modality from infrared and X-ray photograph. Obviously, marking up a tensor with a spatial size of 20×20 is easier to mark up than, for example, a 256×256 patch. And due to the fact that the training data set should include as many textural features of the painting as

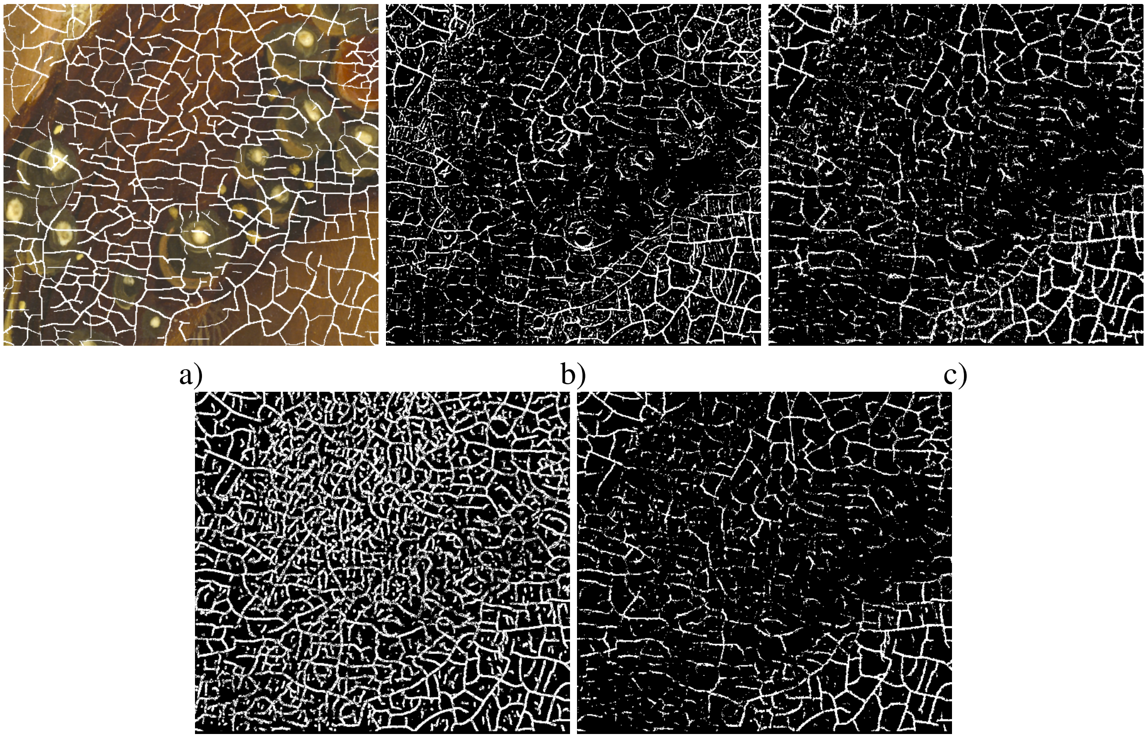


Fig 8: Example of crack detection: a) Part of *Singing Angels* panel, b) Crack map of BCTF, c) Crack map of MCNC, d) Crack map of UNET, e) Crack map of UMCNC

Table 2: Comparison of crack detection methods on a second selected panel from the *Ghent Altarpiece*, where * denotes an extended dataset and $+C$ the use of a technique for suppressing excessive thickening of the crack boundaries.[11]

Singing angels panel					
Method	Recall	False alar.	False miss.	Precision	F_1-m.
CNN[10]	0.6119	0.0999	0.3881	0.4680	0.5304
DFFN[38]	0.6242	0.0966	0.3758	0.4814	0.5436
BCTF[7]	0.6150	0.0905	0.3850	0.4941	0.5479
MCN[11]	0.6340	0.0894	0.3660	0.5048	0.5621
MCNC[11]	0.6083	0.0681	0.3917	0.5622	0.5843
UNET	**0.7412**	0.2089	**0.2588**	0.3376	0.4639
UMCN	0.6080	0.0655	0.3920	0.5713	0.5891
UMCNC	0.5833	**0.0528**	0.4167	**0.6134**	**0.5980**

possible, the number of tensors can increase significantly, which ultimately may call into question the reasonableness of automatic crack detection.

Further, in addition to the completeness of the tensor marking for training (i.e., all cracks in the tensor must be marked), the UNet architecture strongly depends on the accuracy of the crack

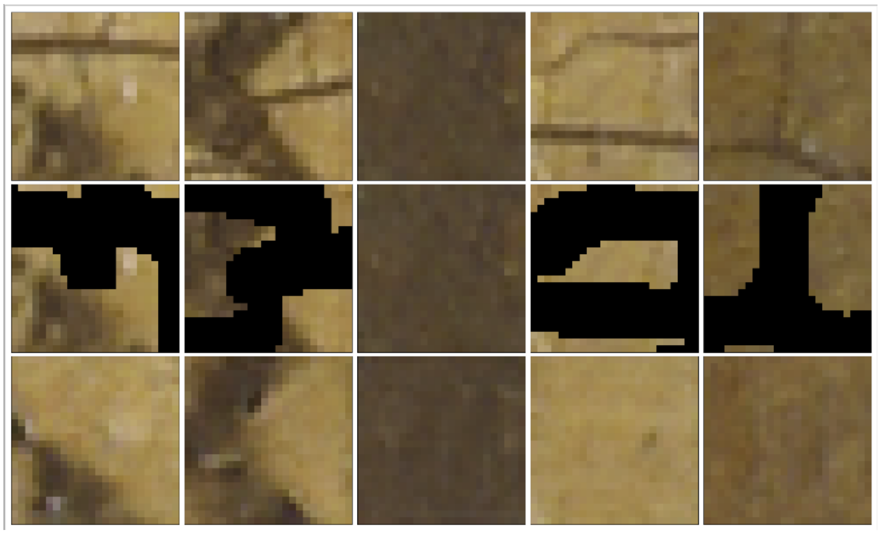


Fig 9: Example of removing detected cracks. First row: images with cracks, Second row: images with mask, Third row: removed cracks by proposed aGAN

covering. So an inaccurate coating can lead to excessive thinning or thickening of the actual crack boundaries. Therefore, in order to improve the quality of crack detection, we use the technique proposed in work.[11] This technique allows to deal with excessive thickening of the crack boundaries on the resulting map.

Based on the analysis of the results obtained, it can be seen that the UMCN/UMCNC hybrid network is superior to known crack detection methods. The combination of a convolutional network and an autoencoder based on the U-Net network can significantly reduce the probability of a false positive, which leads to an increase in the F_1 metric. It is also clear from the results that the pure U-Net network has a large number of false positives. These errors are mainly related to inaccurate descriptions of the actual crack boundaries. This is due to the fact that the training data set was created with an emphasis on user convenience, and did not take into account the limitations that may arise when using the U-Net model. That is, when marking cracks to form a training set, the user did not cover the whole crack, but only its central part. Additionally, it can be seen from the results that the technique of improving the boundaries of cracks confirms its effectiveness.

4.2 Crack removal

In this section we present the result of virtual restoration of two paintings: *Annunciation virgin Mary* and *Singing Angels*. As well-known methods, we use exemplar based method (EBM)[21] and a context-aware method based on Markov Random Fields (MRF),[22] both of which proved to be

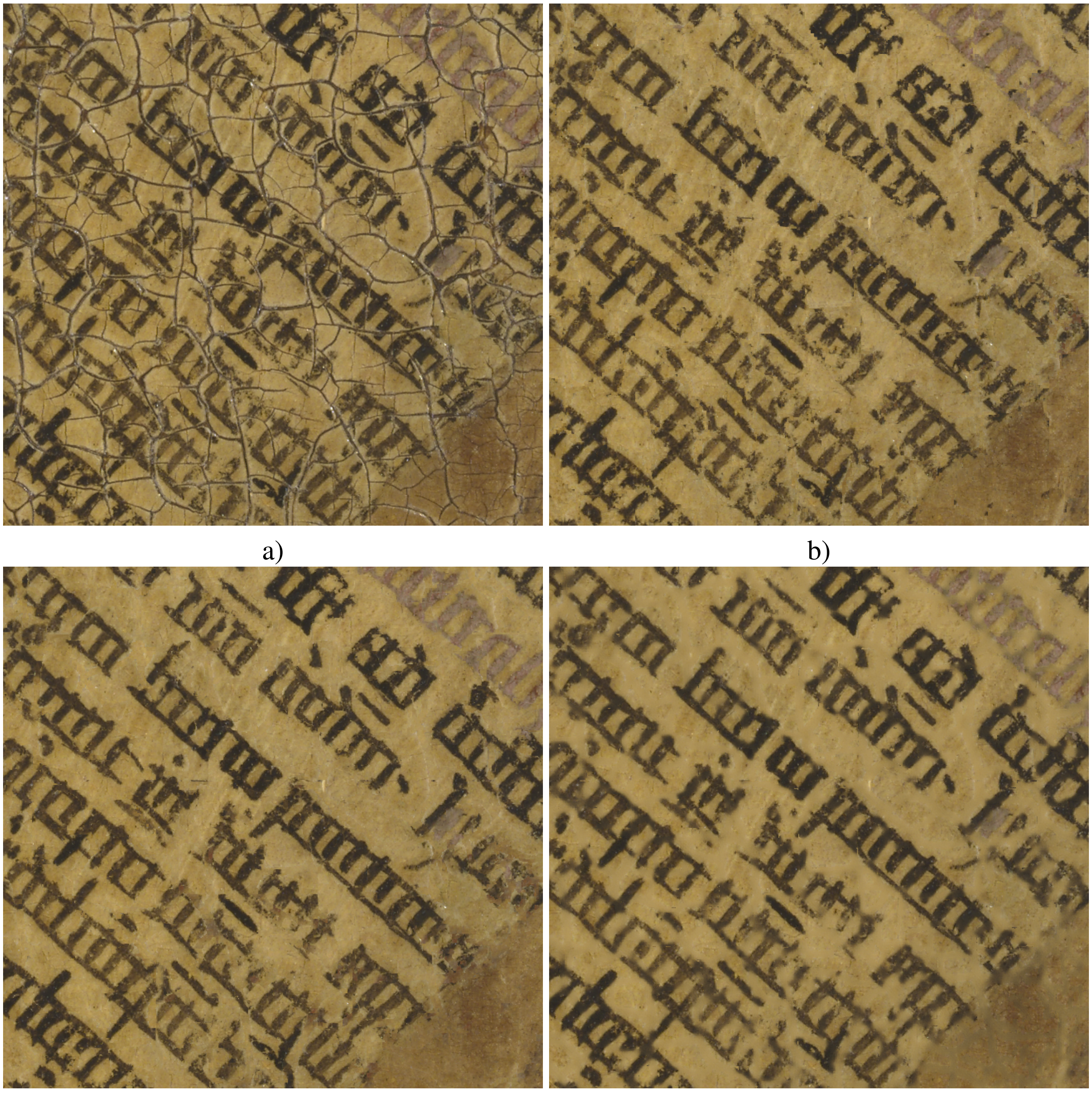


Fig 10: Example of removing detected cracks of the panel *Annunciation virgin Mary* a) Parts of the original painting, b) The result of EBM, c) The result of context-aware MRF, d) The proposed aGAN technique.

successful in virtual restoration of paintings.[8] These methods are based on searching for patches on undamaged areas of the image and then filling in the damaged area with them. The main challenge for such methods are cases when an undamaged area does not contain a semantically connected object to an object that has been deleted. This can occur when the lost area is large. Nevertheless, in the problems of crack removal, such situations are rare, so such methods can be

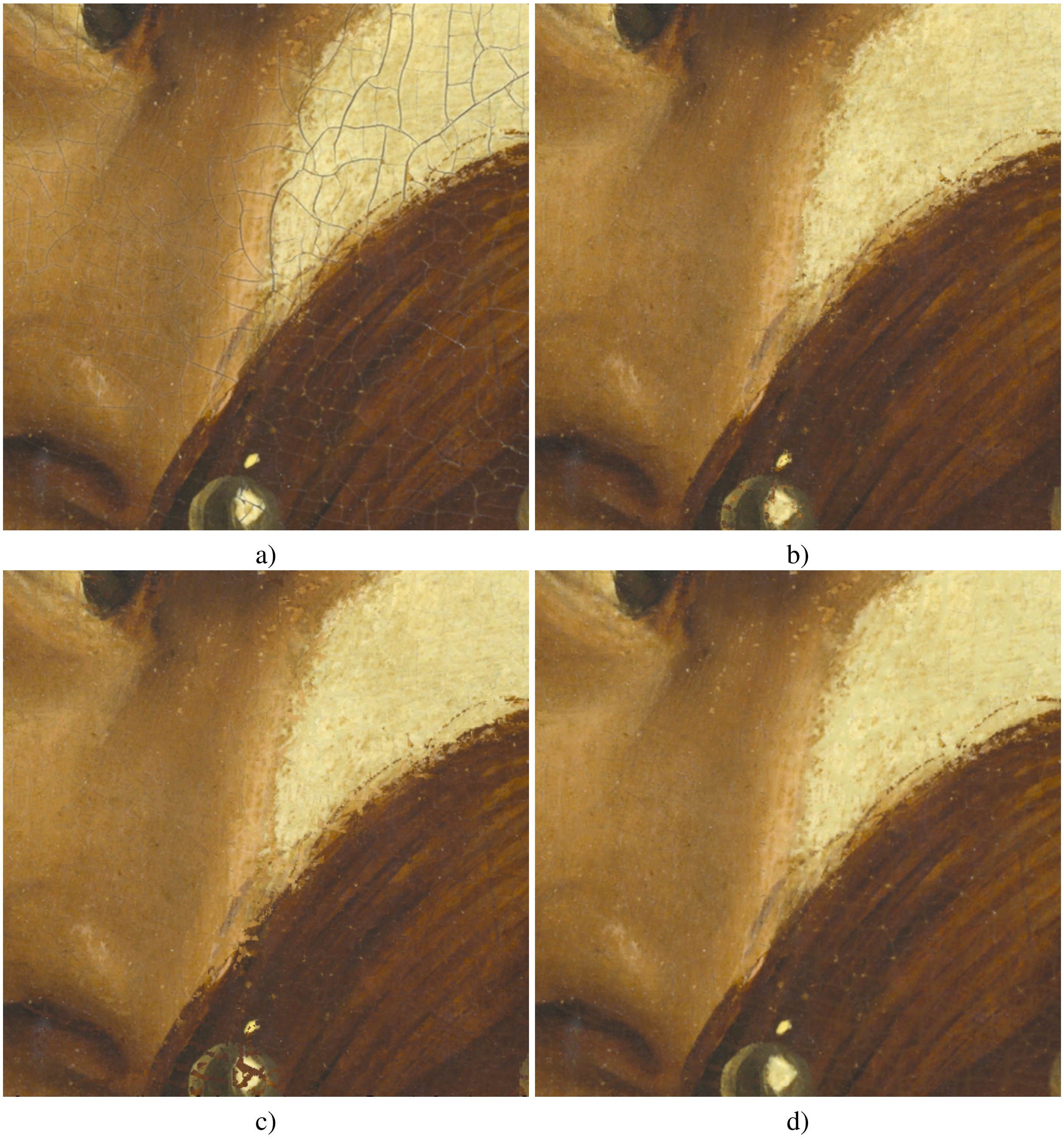


Fig 11: Example of removing detected cracks of the panel *Singing Angels* a) Parts of the original painting, b) The result of EBM, c) The result of context-aware MRF, d) The proposed aGAN technique.

successfully applied.

An example of crack removal for the proposed adaptive adversarial network (aGAN) illustrated on Figure 9. Figure 10 shows the result of removing cracks in the *Annunciation virgin Mary* painting. Figure 10 shows that the EBM method has some spikes that degrade the overall perception of the restored image. The painting restored using the context-aware method based on Markov Random Fields method looks much better. However, upon closer looks, some objects are visible that have been inpainted in areas where they should not be. The restoration result using the proposed aGAN approach has no such drawbacks. However, some areas of the restoration do not look sharp enough.

Figure 11 shows the result of removing cracks in the *Singing Angels* painting. The analysis of the results confirms the effectiveness of restoration methods based on adversarial neural networks. As before, the EBM and CA-MRF methods have a certain amount of structurally incorrectly reconstructed regions. This is especially visible in the pearl area as well as in areas of swift contrast changes.

5 Conclusion

In this paper, a study was performed aimed at investigating the possibility of virtual restoration of paintings using deep learning. This study consists of two parts: the detection of cracks and their removal. To detect cracks, we use a combination of two neural network models: a convolutional neural network and a segmenting neural network based on the U-Net architecture. This combination has a number of advantages that are presented for crack detection methods for their successful application in practice. This includes easy creation of a training set without the need for excessively painstaking and accurate labeling of cracks, the ability to use an online training model when new training data becomes available, high speed of training and creating a crack map, as well as the absence of the need for hand-engineering texture descriptors. Additionally, the proposed architecture provides significant accuracy of crack localization, which is confirmed by numerical results. The second part of this paper is dedicated to the problem of removing detected cracks. To do this, we use an adaptive adversarial network. The key novelty is the coefficient that allows to dynamically evaluate the trend of the loss function in the learning process. In our work, we use this coefficient for adaptive balancing of the loss function and finding a tradeoff between sharpness and the global structure of the restored area. The results obtained show encouraging results in comparison with known methods.

Generally based on the results obtained, it can be concluded that combining different neural network architectures can improve the result if compared with the result from each architecture separately. Also, the use of adaptive feedback through the trend estimation coefficient for the generative network has the potential for further study to obtain a higher-quality reconstruction result. Therefore, further work will be carried out in these directions.

Acknowledgments

The research was supported in part by the Flanders AI research program and part of Russian Science Foundation under grant No 21-79-10392 (https://rscf.ru/project/21-79-10392). This manuscript is a further research of the article (accepted and will be published in the proceedings of the 14th SPIE, ICMV Conference in Rome, Italy): Virtual Restoration of Paintings Based on Deep Learning, Roman Sizyakin, Viacheslav Voronin, Aleksandra Pizurica.

References

1 N. Otsu, "A threshold selection method from gray-level histogram," *IEEE Transaction on Systems, Man, and Cybernatics* **9**, 62–66 (1979).

2 H. Oliveira and P. Correia, "Automatic road crack segmentation using entropy and image dynamic thresholding," *7th European Signal Processing Conference (EUSIPCO)* (2009).

3 A. Gupta, V. Khandelwal, A. Gupta, *et al.*, "Image processing methods for the restoration of digitized paintings," *International Journal of Science and Technology* **13**(3), 66–72 (2008).

4 I. Giakoumis, N. Nikolaidis, and I. Pitas, "Digital image processing techniques for the detection and removal of cracks in digitized paintings," *IEEE Transactions on Image Processing* **15**, 178–188 (2006).

5 D.-M. Tsai, C.-P. Lin, and K.-T. Huang, "Defect detection in coloured texture surfaces using gabor filters," *Imaging Science Journal* **53**(1), 27–37 (2005).

6 K. C. P. Wang, Q. Li, and W. Gong, "Wavelet-based pavement distress image edge detection with "Àtrous"algorithm," *Transportation Research Record* **2024**(1), 73–81 (2007).

7 B. Cornelis, Y. Yang, J. T. Vogelstein, *et al.*, "Bayesian crack detection in ultra high resolution multimodal images of paintings," *IEEE, 18th International Conference on Digital Signal Processing* (2013).

8 A. Pižurica, L. Platiša, T. Ružić, *et al.*, "Digital image processing of the Ghent altarpiece: supporting the painting's study and conservation treatment," *IEEE Signal Processing Magazine* **32**, 112–122 (2015).

9 Y. Yang and D. B. Dunson, "Bayesian conditional tensor factorizations for high-dimensional classification," *Journal of the American Statistical Association* **111**(512), 1–32 (2013).

10 Z. Lei, Y. Fan, D. Yimin, *et al.*, "Road crack detection using deep convolutional neural network," *IEEE International Conference on Image Processing (ICIP)* , 3708–3712 (2016).

11 R. Sizyakin, B. Cornelis, L. Meeus, *et al.*, "Crack detection in paintings using convolutional neural networks," *IEEE Access* **8**, 74535–74552 (2020).

12 Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Computer - Aided Civil and Infrastructure Engineering* **32**, 361–378 (2017).

13 Y. Li, H. Li, and H. Wang, "Pixel-wise crack detection using deep local pattern predictor for robot application," *MDPI and ACS Style* (2018).

14 B. Kim and S. Cho, "Automated vision-based detection of cracks on concrete surfaces using a deep learning technique," *MDPI and ACS Style* (2018).

15 R. Sizyakin, B. Cornelis, V. V. Meeus, L., *et al.*, "A two-stream neural network architecture for the detection and analysis of cracks in panel paintings," *ISO&P, Optic, Photonics and Digital Technologies for Imaging Applications VI* , 1–9 (2020).

16 V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39**(12), 2481–2495 (2017).

17 O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," *Springer,Medical Image Computing and Computer-Assisted Intervention, MICCAI* **9351** (2015).

18 R. Sizyakin, V. Voronin, N. Gapon, *et al.*, "A deep learning approach to crack detection on road surfaces," *ISO&P, Artificial Intelligence and Machine Learning in Defense Applications II* , 1–7 (2020).

19 L. Meeus, S. Huang, N. Zizakic, *et al.*, "Assisting classical paintings restoration: efficient paint loss detection and descriptor-based inpainting using shared pretraining," *SPIE, Optics, Photonics and Digital Technologies for Imaging Applications VI* , 1–13 (2020).

20 M. Bertalmío, A. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001* **1**, I–I (2001).

21 A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on Image Processing* , 1200–1212 (2004).

22 T. Ružic and A. Pižurica, "Context-aware patch-based image inpainting using markov random field modeling," *IEEE Transactions on Image Processing* **24**(1), 444–456 (2015).

23 V. Voronin, V. Marchuk, R. Sizyakin, *et al.*, "Automatic image cracks detection and removal on mobile devices," *Mobile Multimedia/Image Processing, Security, and Applications* (2016).

24 C. Ham, A. Raj, V. Cartillier, *et al.*, "Variational image inpainting," *Third workshop on Bayesian Deep Learning (NIPS)* , 1–6 (2018).

25 S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics* **36**(4) (2017).

26 R. Sizyakin, V. Voronin, N. Gapon, *et al.*, "A deep learning-based approach for defect detection and removing on archival photos," *Electronic Imaging, Society for Imaging Science and Technology,* **10**, 1–7 (2020).

27 Y. Jiahui, L. Zhe, Y. Jimei, *et al.*, "Generative image inpainting with contextual attention," *CoRR* (2018).

28 Hubert and J. V. Eyck, "Image gallery: Closer to van eyck, rediscovering the ghent altarpiece," *http://closertovaneyck.kikirpa.be/* (2011).

29 A. Sörensen T., "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons," *Kongelige Danske Videnskabernes Selskab* **5**(4), 1–34 (1948).

30 R. Dice Lee, "Measures of the amount of ecologic association between species," *Ecology* **26**(3), 297–302 (1945).

31 D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," *ICLR: International Conference on Learning Representations* (2016).

32 A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems* , 1097–1105 (2012).

33 X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, PMLR* **15**, 315–323 (2011).

34 A. L. Maas, "Rectifier nonlinearities improve neural network acoustic models," *International Conference on Machine Learning (ICML)* (2013).

35 S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *Proceedings of the 32nd International Conference on International Conference on Machine Learning* , 448–456 (2015).

36 D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ICLR: International Conference on Learning Representations* (2015).

37 I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," *Advances in Neural Information Processing Systems* , 2672–2680 (2014).

38 W. Song, S. Li, L. Fang, *et al.*, "Hyperspectral image classification with deep feature fusion network," *IEEE Transactions on Geoscience and Remote Sensing* **56**, 3173–3184 (2018).

Roman Sizyakin received the Bachelor of Engineering and Technology degree in radio engineering from the South–Russian State University of Economics and Services, in 2011, and the Master of Engineering and Technology degree in radio engineering from Don State Technical University (DSTU), in 2013. He is currently pursuing the Ph.D. degree with Ghent University, Belgium. Also in parallel, the researcher at laboratory Mathematical methods of image processing and computer vision intelligent systems, DSTU. His research interests include signal and image processing, mathematical statistics, mathematical modeling, and deep learning.

Viacheslav Voronin is the head of the Center for Cognitive Technology and Machine Vision at Moscow State University of Technology "STANKIN", Moscow, Russian Federation. He received his BS (2006), MS (2008) in the communication system from the South-Russian State University of Economics and Service, and his Ph.D. in technics from Southern Federal University (2009). Voronin is a member of the Program Committee of the conference SPIE. His research interests include image processing, inpainting, and computer vision.

Aleksandr Zelensky is the director of institute at Moscow State University of Technology "STANKIN", Moscow, Russia. He received his BS (2005), MS (2007) in the communication system from the South-Russian State University of Economics and Service, and his Ph.D. in technics from Novocherkassk Polytechnic University (2010). Zelensky A. has authored more than 60 scientific papers. His research interests include collaborative robotics, control systems, and computer vision.

Aleksandra Pižurica received the Diploma degree in electrical engineering from the University of Novi Sad, Serbia, in 1994, the Master of Science degree in telecommunications from the University of Belgrade, Serbia, in 1997, and the Ph.D. degree in engineering from Ghent University, Belgium, in 2002. She is currently a Professor in statistical image modeling with Ghent University. Her research interests include the area of signal and image processing and machine learning, including multiresolution statistical image models, Markov random field models, sparse coding, representation learning, and image and video reconstruction, restoration, and analysis. She has served as an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, from 2012 to 2016, the Senior Area Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, from 2016 to 2019. She is currently an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. She was also the Lead Guest Editor for the EURASIP JOURNAL ON ADVANCES IN SIGNAL PROCESSING for the Special Issue Advanced Statistical Tools for Enhanced Quality Digital Imaging with Realistic Capture Models, in 2013. The work of her team has been awarded twice the Best Paper Award of the IEEE Geoscience and Remote Sensing Society Data Fusion contest, in 2013 and 2014. She received the scientific prize de Boelpaepe, from 2013 to 2014, awarded by the Royal Academy of Science, Letters, and Fine Arts of Belgium for her contributions to statistical image modeling and applications to digital painting analysis.

List of Figures

List of Tables