A two-stream neural network architecture for the detection and analysis of cracks in panel paintings

Roman Sizyakin[a], Bruno Cornelis[b], Laurens Meeus[a], Viacheslav Voronin[c], and Aleksandra Pižurica[a]

[a]Department Telecommunications and Information Processing, TELIN-GAIM, Ghent University, Ghent, Belgium
[b]Department of Electronics and Informatics, Vrije Universiteit Brussel, Brussels, Belgium
[c]Center for Cognitive Technology and Machine Vision, Moscow State University of Technology "STANKIN", Moscow, Russian Federation

ABSTRACT

Museums all over the world store a large variety of digitized paintings and other works of art with significant historical value. Over time, these works of art deteriorate, making them lose their original splendour. For paintings, cracks and paint losses are the most prominent types of deterioration, mainly caused by environmental factors, such as fluctuations in temperature or humidity, improper storage conditions and even physical impacts. We propose a neural network architecture for the detection of crack patterns in paintings, using visual acquisitions from different modalities. The proposed architecture is composed of two neural network streams, one is a fully connected neural network while the other consists of a multiscale convolutional neural network. The convolutional neural network plays a leading role in the crack classification task, while the fully connected neural network plays an auxiliary role. To reduce the overall computational complexity of the proposed method, we use morphological filtering as a pre-processing step to safely exclude areas of the image that do not contain cracks and do not need further processing. We validate the proposed method on a multimodal visual dataset from the *Ghent Altarpiece*, a world famous polyptych by the Van Eyck brothers. The results show an encouraging performance of the proposed approach compared to traditional machine learning methods and the state-of-the-art Bayesian Conditional Tensor Factorization (BCTF) method for crack detection.

Keywords: Crack detection, convolutional neural network (CNN), fully connected neural network, multimodal data, panel paintings.

1. INTRODUCTION

Most museums acquire and store information on their art collections in digital form. While most of this information is used for documentation and archival purposes, it also constitutes a crucial source of information for in-depth studies of these artworks. Most historical paintings accumulate cracks in their varnish layer, which are formed as a result of excessive pressure within one or more paint layers. The causes of this pressure in paintings are improper storage conditions, physical damage, and ageing of the painting materials. Cracks come in many shapes, from regular to completely random patterns, and various colors, ranging from dark to light.

Automatic crack detection is a complex and non-trivial task, but is a crucial component in a wide range of applications. For example, automatic crack detection can help restorers analyze the state of conservation of a painting, determine whether storage conditions are adequate, and confirm or deny a painting's authenticity.[1] In addition, the detected crack patterns can be digitally inpainted, providing a simulation, or at least a rough idea, of the restored painting. Such a study was done in an earlier work by some of the authors,[2] where digital inpainting of cracks proved useful for the deciphering of the content in a book, depicted in one of the panels of the *Ghent Altarpiece*.

The main challenges we face when detecting cracks are: (i) cracks, and other types of paint loss, may have low contrast compared to their background, (ii) crack patterns form complex ragged structures, and (iii) cracks can sometimes be difficult to distinguish from other thin and elongated painted content, such as eyelashes or hair (see Figure 1). These challenges can partially be alleviated by jointly using different modalities, often made available

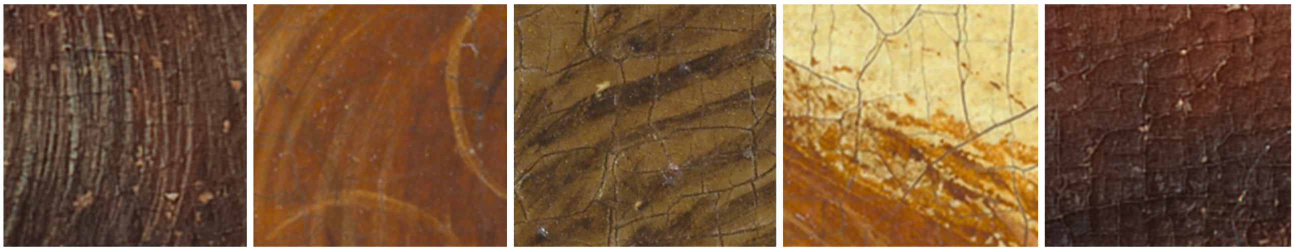
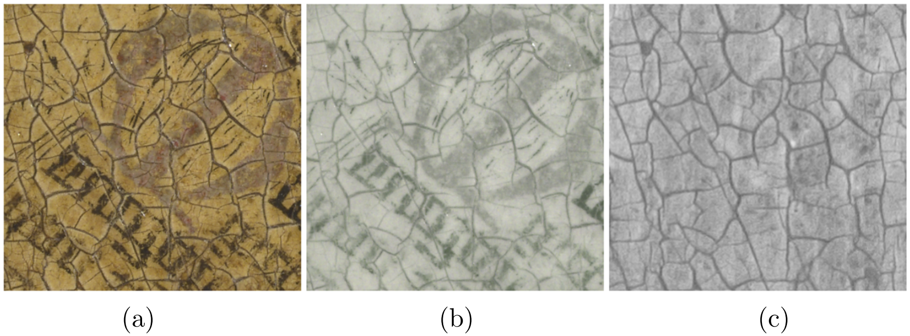


Figure 1. Example of cracks in panel paintings



(a) (b) (c)

Figure 2. Illustration multimodal data (part of the panel *Annunciation virgin Mary*). a) Visual macrophotograh image, b) infrared digital macrophotography, c) X-radiography

by museums. An example of such a multimodal acquisition is shown in Figure 2, depicting a small sample taken from the *Ghent Altarpiece*. Another challenge that occurs frequently comes from the large resolution of the acquisitions, which may impose additional limitations on some crack detection methods.

Taking into account the challenges described above, we propose a crack detection method based on a two-stream neural network. Additionally, we use morphological filtering as a pre-processing step to strongly reduce the computational complexity of the overall method, making it applicable in real-life scenarios.

2. RELATED WORK

The general goal of crack detection methods is to create a binary map $d_{i,j}$, called crack map, which accurately describes the location of cracks in the painting. The mathematical model of the digitized painting containing the cracks can be represented as follows:

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

where $Y_{i,j}$ is the image with cracks, $i = \overline{1, I}$ and $j = \overline{1, J}$ are spatial coordinates, with I and J the height and width of the image respectively, $S_{i,j}$ is the crack-free image, $d_{i,j} \in \{0, 1\}$ is the crack map, and where $c_{i,j}$ contains the brightness values of cracks.

Methods for automatic crack detection can be divided into three main groups: methods based on spatial image filtering,[3] methods based on machine learning,[4,5] and methods combining both.[6] Spatial filtering methods typically employ a variety of grayscale morphological filters followed by a thresholding step, where the threshold is chosen either manually or automatically, using e.g. Otsu's method.[7] A hybrid approach[6] combines the results of distinctive methods based on directionally sensitive filters, morphological operations, and dictionary learning. The binary images obtained by these different methods are combined through a voting scheme. Machine learning approaches are based on vector classification[4,5,8] or tensor classification.[9–11] An efficient Bayesian crack detection method[8] employed Bayesian Conditional Tensor Factorization (BCTF)[12] to detect cracks on a multimodal dataset and proved excellent results on high-resolution images of the *Ghent Altarpiece*.

It should be noted that most of the earlier reported crack detection methods, except for a few exceptions like the BCTF[8] approach, were developed for detecting cracks using a single modality only. Thus, they cannot exploit the valuable information present in other modalities. Secondly, most methods require crafting features manually. Thirdly, the existing approaches for crack detection in paintings are not able to deal well with situations where continuous learning is desired. Finally, processing high resolution images becomes highly problematic in terms of computational complexity, limiting the practical applicability of most methods.

Several recent works reported using convolution neural network to detect cracks in road surfaces.[10,13,14] In comparison, the problem of crack detection in paintings is considerably more challenging, as paintings have a much more complex background structure. Cracks in paintings are often difficult to distinguish from other background objects, such as brush strokes and other line-like details, which makes their automatic detection more challenging. Furthermore, the aforementioned deep learning based methods suffer from excessive thickening of crack boundaries, making their accurate detection difficult. An example of this excessive thickening will be illustrated in the experimental section of this paper.

In this paper, we propose a neural network architecture that combines two neural network streams. The first stream, composed of a fully connected neural network, is used to limit excessive thickening of the crack boundaries. The second stream, a convolutional neural network, performs the actual pixel classification function. The main advantage is that the proposed architecture is implemented in a joint learning stream, this allows to deal well with situations where continuous training is required. This is very important because manually marking data for each painting is impractical. Furthermore, our method is designed for efficient processing of high resolution multimodal datasets. Therefore, we use a two-staged approach, where morphological filtering is performed prior to the classification by our two-stream neural network. Morphological filtering allows to effectively and safely eliminate areas that do not contain any cracks, and hence do not need to go through the classification process. A thorough validation of the overall method was performed on a multimodal visual dataset from the *Ghent Altarpiece*[*] by the Van Eyck brothers, a world famous polyptych. We demonstrate that the proposed approach is capable of accurately localising crack pixels in paintings and produces better results compared to traditional machine learning methods as well as the Bayesian Conditional Tensor Factorization (BCTF) approach, considered to be the state of the art at the moment.

3. PROPOSED METHOD

To reduce the computational complexity that is involved when working on high resolution data, we opt for the approach of eliminating areas in the painting that can be confidently declared not to contain any cracks, by means of simple spatial filtering. This pre-processing step significantly accelerates the overall crack detection process. The overall method thus consists of two processing stages: i) a morphological filtering stage and ii) a classification stage using a two-stream neural networks.

3.1 Morphological filtering

Morphological image filtering is a popular technique for the preliminary localization of details in images. In our case, it is used to reduce the computational complexity of the overall crack detection method and, in some cases, to reduce false positives.[15] We use both the "top" and "bottom hat" transforms, which are constructed by means of four binary mathematical morphology operations, namely closing, opening, erosion, and dilation:

$$BottomHat(Y_{i,j}, B) = ((Y_{i,j} \oplus B) \ominus B) - Y_{i,j}, \quad TopHat(Y_{i,j}, B) = Y_{i,j} - ((Y_{i,j} \ominus B) \oplus B), \tag{2}$$

where B is called a structuring element, $(Y_{i,j} \oplus B) \ominus B$ is morphological closing, $(Y_{i,j} \ominus B) \oplus B$ is morphological opening, $\ominus$ and $\oplus$ are the erosion and dilation operations, respectively. The result of morphological filtering is illustrated in Figure 3(b). The size and shape of the structuring element are important, as they determine the morphology of the objects within the image that will be highlighted. For our experiments, a disk-shaped structuring element B of 3×3 pixels ensures the proper detection of all cracks within the image. We set the threshold value based on work[7] (see Figure 3(c)). This procedure is applied for all the involved modalities. The obtained binary maps are then combined into one crack map using the logical "OR" operator.

[*] Link: http://closertovaneyck.kikirpa.be/ghentaltarpiece/

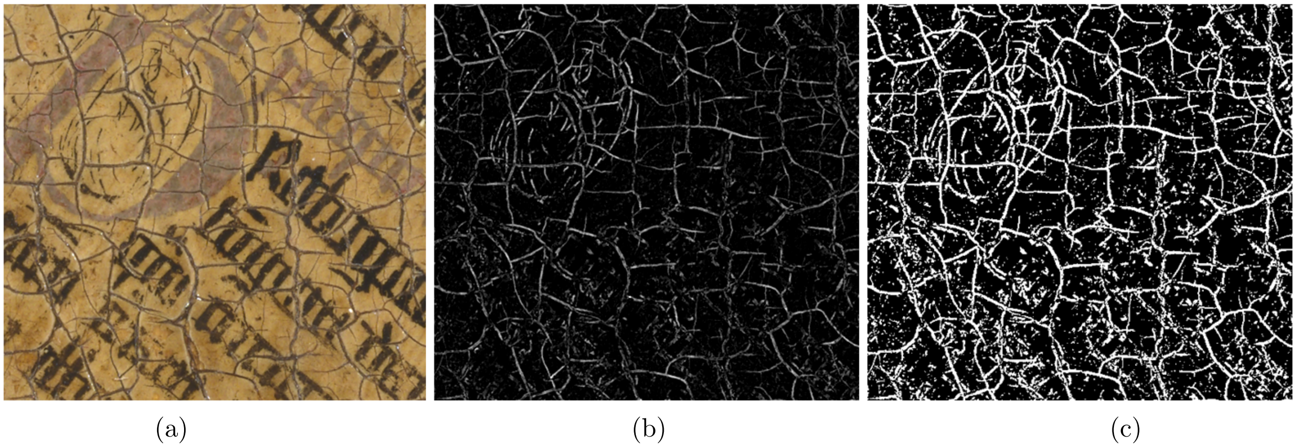


Figure 3. Preliminary localization of cracks using morphological filtration. a) source image, b) filtered image,) thresholded image

3.2 Classification of cracks using a deep neural network

After the pre-processing stage, every candidate crack pixel is further classified using the proposed two-steam convolutional neural network. Convolutional neural networks (CNN) have two main advantages over traditional machine learning methods. First, there is no need to manually select texture descriptors since the neural network itself synthesises them during training, and second, CNN models generally demonstrate superior classification accuracy compared to other machine learning methods.[13, 16]

Our deep learning model, consisting of two separate streams, is depicted in Figure 4. Each stream of the proposed model requires a specific input. For the first stream (at the top of Figure 4), which is essentially a fully connected neural network, a vector is constructed by traversing every pixel in all available modalities. The second stream on the other hand expects a tensor, constructed by stacking patches from all the modalities. Both streams have their specific purpose. The first stream improves localization of the classification task and reduces thickening observed at the edges of cracks. The second stream, constructed with convolutional layers, is more resistant to false positives caused by inter-modal shift, noise and other distortions. The combination of both sub networks ensures proper localization as well as robust classification.

At the initial stage of neural network training, the kernels in all convolutional layers are initialised randomly. The convolution operation at each stage of the network can be defined as:

$$x_{h,v}^{l,c} = f(\sum_{h}\sum_{v}\sum_{c} x_{h+m,v+n}^{l-1,c} \cdot k_{h,v}^{l,c} + b), \tag{3}$$

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. An exponential linear unit (ELU)[17] is used as activation function for all convolutional and fully connected layers:

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ a(e^x - 1) & \text{if } x \leq 0, \end{cases} \tag{4}$$

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

To ensure that the output of our network sums to one, the fully connected layers 6 and 7 are followed by a softmax function, defined as:

$$y^{'} = \lambda_1 \frac{e^z}{\sum_r e^{z_r}} + \lambda_2 \frac{e^z}{\sum_r e^{z_r}}, \tag{5}$$

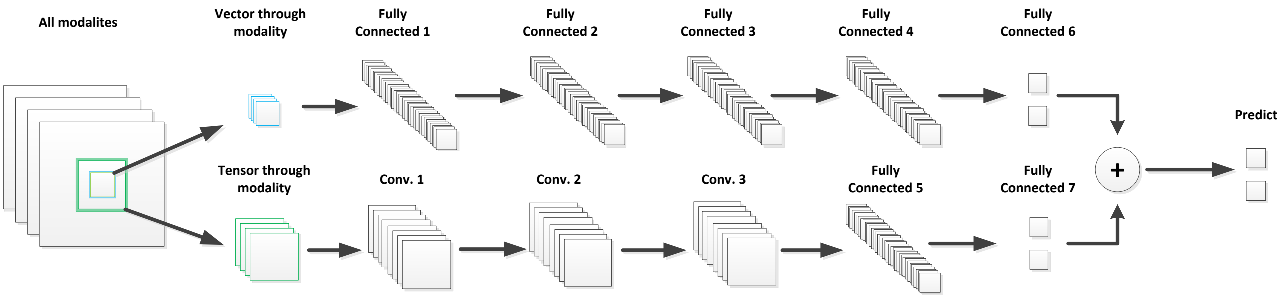


Figure 4. The proposed architecture of the convolutional neural network

where z is the output value for the current descriptor after passing the neural network, the coefficients λ_1 and λ_2 sets the proportional contribution from each stream of the proposed neural network to the total losses. In our work we use equal coefficients $\lambda_1 = \lambda_2 = 0.5$, which is equivalent to the same contribution to the total losses from two streams. To determine the losses, we use the binary cross-entropy function. This is defined according to the expression:

$$H(y, y^{'}) = -\frac{1}{m}\sum_{l=1}^{m}[y \cdot log(y^{'}) + (1 - y) \cdot log(1 - y^{'})], \tag{6}$$

where y is the network prediction and $y^{'}$ is the ground truth.

The layers of the neural network are parameterized as follows: C1-16, C2-32, C3-64, FC1-16, FC2-32, FC3-64, FC4-128, FC5-128, FC6-2, FC7-2, where C denotes a convolutional layer, FC denotes a fully connected layer, the first digit corresponds to the layer number and the second to the amount of filters/neurons. All convolutional layers have a spatial filter size of 3×3 pixels. Adam optimization is used for training,[18] with a learning rate of 0.0001. The training process took approximately 50 to 70 epochs, with a batch size of 100 samples.

4. EXPERIMENTAL RESULTS

To evaluate and compare the proposed method, we use a dataset from the *Ghent Altarpiece*, composed of high resolution images from the following modalities: macrophotography, infrared macrophotography (IRP) and X-ray images. Our deep neural network model was trained with ground truth data from.[8] Since the original images are of large resolution, and for the sake of clarity, we only report results for small parts of the following panels: *Annunciation virgin Mary* and *Singing Angels*.

A total of 9 imaging modalities are used as input data; i.e. the three color channels of the visual macro-photograph, the X-Ray image and single-channel infrared macrophotograph, as well as a fourth "modality" obtained by morphological filtering ("bottom hat" for all modalities and "top hat" for visual macro photography only).

For comparison, we use the following machine learning methods: AdaBoost,[19] support vector machine (SVM),[20] standard fully connected neural network (NN),[21] the CNN method proposed in,[13] and the BCTF method.[8] We denote our two-stream architecture as TsCNN.

The standard fully connected neural network (NN) has 3 layers with sigmoidal activation functions with 100 neurons in each layer. Backpropagation was used for training. The boosting method uses a decision tree as a weak classifier. The minimum classification error was achieved after 2000 iterations. For the support vector machine (SVM) a "linear function" is used as the kernel.

For the AdaBoost, SVM, and NN methods, an input vector is constructed with the LBP texture descriptor,[22] mean, and standard deviation of 7×7 pixel patches, taken from each modality. The vector for one modality

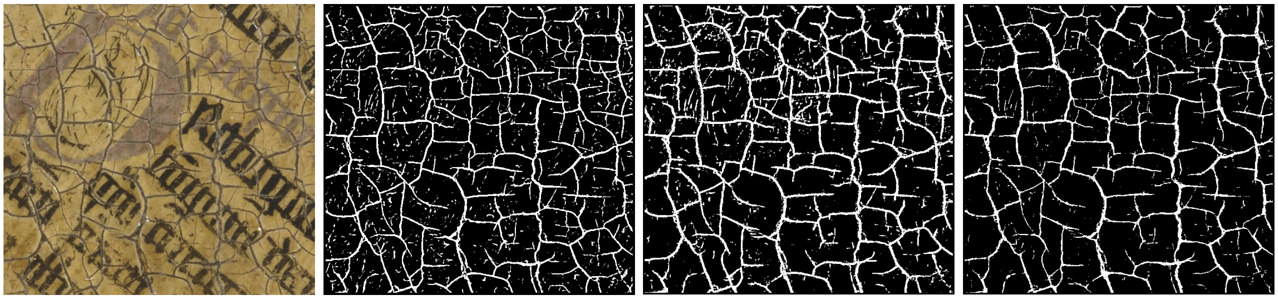

(a) (b) (c) (d)

Figure 5. Illustration of the crack detection maps for the first test image (part of the panel *Annunciation virgin Mary*). a) Visual macrophotograph image, b) BCTF method, c) CNN method, d) TsCNN method

Table 1. Experimental results for the first test image

Method	Recall	False alar.	Precision	F_1-m.
ADA	**0.8695**	0.1183	0.5013	0.6360
SVM	0.8471	0.0832	0.5822	0.6901
NN	0.8468	0.0840	0.5796	0.6882
CNN	0.8481	0.0777	0.5989	0.7020
BCTF	0.7896	0.0535	0.6686	0.7241
TsCNN	0.7078	**0.0284**	**0.7733**	**0.7391**

has a size of 20 values, where 18 values are obtained from the LBP descriptor and 2 values from the mean and standard deviation. The total resulting vector has 180 values.

The CNN method proposed by Lei et al[13] has a standard convolutional neural network architecture, which has 4 convolutional layers and two fully connected ones. Convolutional layers produce 48 feature maps and the first fully connected one has 200 neurons. For training, the stochastic gradient descent method is used with a rectification linear unit (ReLU) as the activation function.

The BCTF method uses a vector for classification with 208 elements composed of raw multimodal data, as well as their pre-processed versions, obtained using various spatial filters. BCTF outputs a probability for each pixel, thus classification is performed by thresholding these probabilities with a threshold of 0.5.

The following quantitative metrics are used:

$$FA = \frac{FP}{AlPx - DfPx}, P = \frac{TP}{TP+FP}, R = \frac{TP}{TP+FN}, F_1 = \frac{2 \cdot P \cdot R}{P+R} \tag{7}$$

where FA is the probability of false alarm, P is the precision, R is the recall, TP are the number of true positives, FP are the number of false positives, FN is the number of false negative, $DfPx$ is the total amount of pixels belonging to a crack, and $AlPx$ is the total amount of pixels in the image.

The first test image (shown in Figure 5) is particularly challenging as the painted letters are very similar to cracks. Figure 5(b), (c) and (d) show the crack detection results obtained with BCTF, CNN and our TsCNN, respectively. Analysis of the results in Table 1 shows that the proposed TsCNN significantly reduced the amount of false alarms compared to the other methods. However, TsCNN has the smallest *Recall* value. This result can be explained by the fact that the proposed method, in addition to a significant reduction in the false thickening of the boundaries, in some cases also leads to a certain decrease in the correctly detected boundaries of these cracks.

The main challenge for crack detection in the second test image (depicted in Figure 6(a)) is the poorly visible cracks at the bottom of the image. Figure 6(b), (c) and (d) show the corresponding crack maps obtained with BCTF, the CNN method and the TsCNN method, respectively.

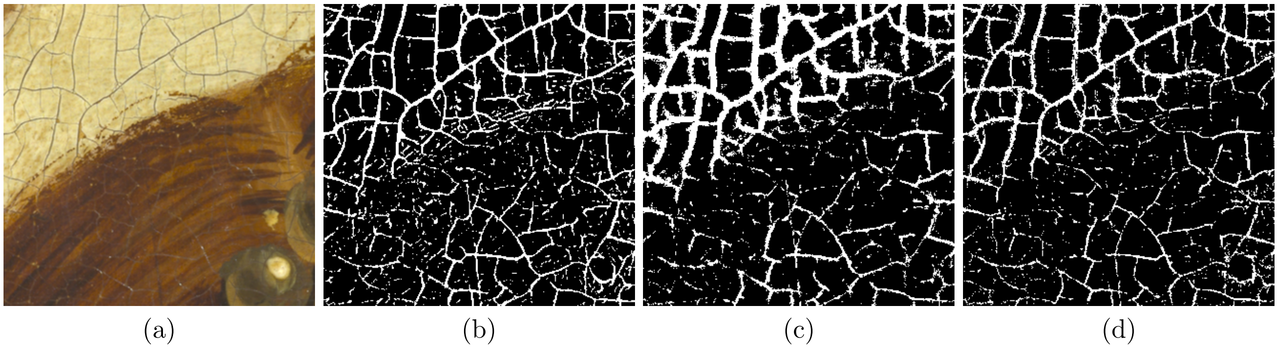


Figure 6. Illustration of the crack detection maps for the first test image (part of the panel *Singing Angels*). a) Visual macrophotograh image, b) BCTF method, c) CNN method, d) TsCNN method

Table 2. Experimental results for second test image

Method	Recall	False alar.	Precision	F_1-m.
ADA	**0.6475**	0.1391	0.4008	0.4951
SVM	0.5111	0.0756	0.4927	0.5017
NN	0.5655	0.0877	0.4809	0.5198
CNN	0.6119	0.0999	0.4680	0.5304
BCTF	0.6150	0.0905	0.4941	0.5479
TsCNN	0.5765	**0.0709**	**0.5387**	**0.5569**

Analysis of the results obtained for the second test image confirmed the effectiveness of the proposed method compared with the standard CNN and BCTF methods. As can be seen from Figure 6, the CNN method demonstrates excessive thickening of the crack boundaries. The proposed method also has a slight thickening of the boundaries, in comparison with the BCTF method. In summary, the quantitative metrics for the second test image present in Table 2 confirmed the results illustrated on Figure 6.

Based on the results obtained, it can be concluded that the methods relying on deep learning show superior results for crack detection in paintings compared to methods based on traditional machine learning. The competing BCTF method yields a much higher number of false positives caused by incorrect classification of some painted objects falsely detected as cracks. The proposed method has the highest value of the $F1 - measure$ in comparison with the reference methods, and most of its false positives are associated only with some thickening of the true boundaries of cracks.

5. CONCLUSION

In this paper, we present a new method for detecting cracks in paintings based on deep learning. As a preprocessing stage, we use morphological filtering, which allows to reduce the computational complexity, as well as reduce the number of false positives. The proposed neural network combines the advantages of two different neural network architectures. A fully connected neural network avoids excessive thickening of the crack boundaries, and a convolutional network increases robustness to various distortions of input data. Combining the two neural network architectures into one joint learning stream provides an efficient learning model in situations where continuous training is required. The analysis of the experimental results confirmed the efficiency of the proposed architecture, in comparison with the classification method based on a convolutional neural network alone and in comparison with the state-of-the-art method BCTF. Future work will include further improving the exact localization of the crack boundaries.

ACKNOWLEDGMENTS

The reported study was funded by the Russian Ministry of Education and Science in accordance with the state assignment according to the research project №075-03-2020-079/2 and by RFBR and NSFC according to the research project №20-57-53012 and in part by the Research Foundation Flanders (FWO), project 528 G.OA26.17N.

REFERENCES

[1] Mohen, J., Menu, M., and Mottin, B., "Mona Lisa: Inside the painting," *Harry N. Abrams, New York, NY, USA* (2006).

[2] Pižurica, A., Platiša, L., Ružić, T., Cornelis, B., Dooms, A., Martens, M., Dubois, H., Devolder, B., Mey, M. D., and Daubechies, I., "Digital image processing of the Ghent altarpiece: supporting the painting's study and conservation treatment," *IEEE Signal Processing Magazine* **32**, 112–122 (2015).

[3] Gupta, A., Khandelwal, V., Gupta, A., and Thammasat, M. C. S., "Image processing methods for the restoration of digitized paintings," *International Journal of Science and Technology* **13**(3), 66–72 (2008).

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

[5] Spagnolo, G. S. and Somma, F., "Virtual restoration of cracks in digitized image of paintings," *International Conference on Defects in Insulating Materials, Journal of Physics* **249** (2010).

[6] Cornelis, B., Ružić, T., Gezels, E., Dooms, A., Pižurica, A., Platiša, L., Cornelis, J., Martens, M., Mey, M. D., and Daubechies, I., "Crack detection and inpainting for virtual restoration of paintings: The case of the Ghent altarpiece," *Signal Processing* **93**(3), 605–619 (2013).

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

[8] Cornelis, B., Yang, Y., Vogelstein, J. T., Dooms, A., Daubechies, I., and Dunson, D. B., "Bayesian crack detection in ultra high resolution multimodal images of paintings," *IEEE, 18th International Conference on Digital Signal Processing* (2013).

[9] Sizyakin, R., Voronin, V., Gapon, N., Zelensky, A., and Pižurica, A., "Automatic detection of welding defects using the convolutional neural network," *International Society for Optics and Photonics, Automated Visual Inspection and Machine Vision III* (2019).

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

[11] Voronin, V., Marchuk, V., Sizyakin, R., Gapon, N., Pismenskova, M., and Tokareva, S., "Automatic image cracks detection and removal on mobile devices," *Mobile Multimedia/Image Processing, Security, and Applications* (2016).

[12] Yang, Y. and Dunson, D. B., "Bayesian conditional tensor factorizations for high-dimensional classification," *Journal of the American Statistical Association* **111**, 656–669 (2016).

[13] Lei, Z., Fan, Y., Yimin, D., and Ying, J. Z., "Road crack detection using deep convolutional neural network," *IEEE International Conference on Image Processing (ICIP)* , 3708–3712 (2016).

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

[15] Sizyakin, R., Cornelis, B., Meeus, L., Martens, M., Voronin, V., and Pižurica, A., "A deep learning approach to crack detection in panel paintings," *Image Processing for Art Investigation (IP4AI)* , 40–42 (2018).

[16] Simonyan, K. and Zisserman, A., "Very deep convolutional networks for large-scale image recognition," *International Conference on Learning Representations* (2015).

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

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

[19] Freund, Y. and Schapire, R., "A short introduction to boosting," *Journal of the Japanese Society For Artificial Intelligence* **14**(1612), 771–780 (1999).

[20] Scholkopf, B. and Smola, A., "Learning with kernels: Support vector machines, regularization, optimization and beyond, adaptive computation and machine learning," *Cambridge, MA: The MIT Press* (2002).

[21] Rumelhart, D., Hinton, G., and Williams, R., "Learning internal representations by error propagation," *Parallel Data Processing* **1**, 318–362 (1986).

[22] Ojala, T., Pietikainen, M., and Maenpaa, T., "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(7), 971–987 (2002).