IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 29, 2020, accepted April 15, 2020, date of publication April 20, 2020, date of current version May 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2988856

Crack Detection in Paintings Using Convolutional Neural Networks

ROMAN SIZYAKIN[1], BRUNO CORNELIS[2], (Member, IEEE), LAURENS MEEUS[1], (Member, IEEE), HÉLÈNE DUBOIS[3,5], MAXIMILIAAN MARTENS[3], VIACHESLAV VORONIN[4], AND ALEKSANDRA PIŽURICA[1], (Senior Member, IEEE)

[1]Department Telecommunications and Information Processing, TELIN-GAIM, Ghent University, 9000 Ghent, Belgium
[2]Department of Electronics and Informatics, ETRO-imec, Vrije Universiteit, 1050 Brussels, Belgium
[3]Department of Art History, Musicology, and Theatre Studies, Ghent University, 9000 Ghent, Belgium
[4]Center for Cognitive Technology and Machine Vision, Moscow State University of Technology STANKIN, 127994 Moscow, Russia
[5]Royal Institute for Cultural Heritage (KIK-IRPA), 1000 Brussels, Belgium

Corresponding authors: Roman Sizyakin (roman.sizyakin@ugent.be) and Aleksandra Pižurica (aleksandra.pizurica@ugent.be)

This work was supported in part by the Russian Ministry of Science and Education for Support Under the Grant for Conducting Basic Scientific Research by Educational Organizations in 2020–2022 Project under Grant №075-03-2020-079/2, in part by the Research Foundation Flanders (FWO) under Project 528 G.OA26.17N, and in part by the Artificial Intelligence Research Flanders Program funded by the Flemish Government.

ABSTRACT The accurate detection of cracks in paintings, which generally portray rich and varying content, is a challenging task. Traditional crack detection methods are often lacking on recent acquisitions of paintings as they are poorly adapted to high-resolutions and do not make use of the other imaging modalities often at hand. Furthermore, many paintings portray a complex or cluttered composition, significantly complicating a precise detection of cracks when using only photographic material. In this paper, we propose a fast crack detection algorithm based on deep convolutional neural networks (CNN) that is capable of combining several imaging modalities, such as regular photographs, infrared photography and X-Ray images. Moreover, we propose an efficient solution to improve the CNN-based localization of the actual crack boundaries and extend the CNN architecture such that areas where it makes little sense to run expensive learning models are ignored. This allows us to process large resolution scans of paintings more efficiently. The proposed on-line method is capable of continuously learning from newly acquired visual data, thus further improving classification results as more data becomes available. A case study on multimodal acquisitions of the *Ghent Altarpiece*, taken during the currently ongoing conservation-restoration treatment, shows improvements over the state-of-the-art in crack detection methods and demonstrates the potential of our proposed method in assisting art conservators.

INDEX TERMS Digital painting analysis, crack detection, virtual restoration, machine learning, morphological filtering, convolutional neural networks, transfer learning, multimodal data, Ghent Altarpiece.

I. INTRODUCTION

Paint cracking (or craquelure) is the most common type of deterioration encountered in old master paintings. Cracks appear in paint layers as a consequence of stress caused by different factors, including the ageing of the materials used (age cracks), a defective technical execution at the painting stage (premature cracks), and adverse storing conditions [1]. The main cause for cracking in 15th century Flemish paintings lies in the fluctuation of relative humidity, causing the wooden support, usually made from Baltic oak, to shrink and expand. Cracking also often occurs in the top varnish layer due to oxidation. Visually, cracks can be divided into two main categories, i.e. cracks that are darker than their surrounding background, and vice-versa, cracks that are brighter than their background.

The automatic detection of cracks proved to be a considerable help in various art analysis tasks. For example, in [2], [3] crack detection was used in combination with inpainting methods, to digitally remove cracks in selected areas of the *Ghent Altarpiece*. The method was applied on the depiction of a book in the panel of the *Virgin Annunciate*

The associate editor coordinating the review of this manuscript and approving it for publication was Qiangqiang Yuan.

and made a better deciphering of its content possible. Additionally, crack patterns can offer insights on the structural condition or conservation history of a painting [1].

Crack detection methods reported in the literature can roughly be divided into three categories: filtering-based methods [4], machine-learning based methods [5], [6], and methods combining the two [2]. Filtering-based methods typically employ a variety of grayscale morphological filters followed by a thresholding step with either a manual or automatic selection of the threshold value, using e.g. Otsu's method [7]. Machine learning methods include methods based on vector classification [3], [5], [6], [8] and tensor classification [9]–[11]. Numerous feature learning and classification methods have been proposed recently for high-dimensional data processing in other domains, such as hyperspectral processing [12]–[14]. Although hyperspectral imaging is becoming interesting in art investigation, e.g. for non-invasive analysis of pigments and for analysis of underdrawings, it is less relevant for crack detection due to relatively small spatial resolution. Moreover, some paint cracks are best visible in radiography images. Hence, a multimodal imaging approach combining high-resolution macro photographs in the visible and infrared spectral ranges together with radiographs is preferred for crack detection. In [3], [8] a Bayesian approach termed, the Bayesian Conditional Tensor Factorization (BCTF) method [15] was used to detect cracks on a multimodal dataset of high-resolution images of the *Ghent Altarpiece*. Several typical limitations of the methods described above should be noted: Firstly, most of the available crack detection techniques have been developed for single modality images and cannot exploit well the rich information supplied by other imaging modalities that are now common and readily available in conservation science and practice; secondly, traditional approaches require hand-engineering of features to be used in the classification task. Next, when dealing with high-resolution images, computational complexity becomes a huge problem for most of the existing methods; finally, to the best of our knowledge, the existing approaches for crack detection in paintings were not able to deal well with situations where on-line learning is desired. This is very important because hand labelling data for each painting is impractical.

The work presented here is, as far as the authors are aware, the first attempt to detect cracks in paintings by using deep learning, and in particular convolutional neural networks (CNNs). The prior research closest to our method can be found in the works of [5] and [6] in which the authors use morphological filtering operations for the preliminary detection of cracks and subsequently refine the result using a fully connected neural network. However, our work presents significant differences: We use deep learning based on convolutional neural networks rather than a traditional neural network that was employed in [5], [6]. The overall learning architecture is thus different, including a different optimization method, activation function and loss function. Also, as opposed to these earlier methods, our method is designed to effectively process multimodal data.

In [9], [10], [16], [17], convolutional neural network are used to detect cracks in road surfaces (and similar surfaces). The problem that we address is much more challenging. Paintings feature a much more complex background structure compared to roads. Cracks in paintings are often difficult to distinguish from other background objects, such as eyelashes and other line-like details, which makes their automatic detection using only one modality much more challenging.

A common problem that arises when CNN-based methods are applied to the detection of line-like structures is imprecise delineation (widening) of the detected lines. Partially, the problem can be solved using an extended set of training data, which includes additional negative samples taken at the crack boundaries [16]. However, this does not completely avoid excessive thickening, and may also lead to a decrease in classification accuracy.

To solve this problem, we introduce a compensation method that penalizes false broadening of the crack boundaries. This is one of the important technical novelties of this work. The proposed approach yields clearer delineation of the actual crack boundaries, avoiding the common thickening phenomenon.

The main contributions of this work are the following:

(i) We explore the potential of deep learning for crack detection in paintings. We are not aware of any reported works that apply CNNs or other deep learning models to the problem of crack detection in paintings, except for our preliminary result in a conference abstract [18].

(ii) We propose a novel method for reducing excessive thickening of the crack boundaries detected by CNNs. While this is of crucial importance for our application in digital painting analysis, the proposed solution is applicable in general to CNN-based detection of cracks and other line-like and tubular structures in images.

(iii) To enable efficient processing of multimodal images of huge spatial resolution, we design an original two-step procedure with morphological preprocessing. The proposed morphological processing step efficiently and safely eliminates areas where it makes little sense to run the learning process.

(iv) We design our network such that it can continuously learn from new annotations when these become available. The results demonstrate clearly the efficiency of our re-training approach where the network trained on one painting detects successfully cracks in another painting with relatively few extra labels added.

(v) A thorough evaluation is conducted on multimodal acquisitions of the *Ghent Altarpiece* including an elaborate case study, which demonstrates the actual benefit from crack detection for diagnosing the state of specific panels and the importance of this tool in support of art painting conservators.

Preliminary results of this work were presented in a conference abstract only [18]. The present paper gives not only

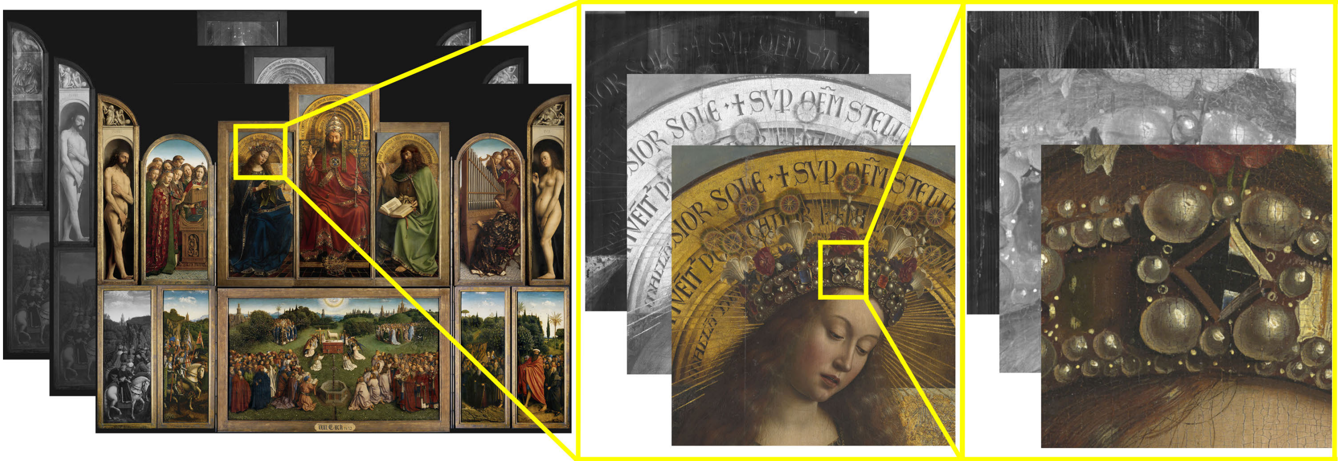


FIGURE 1. **An illustration of the multimodal data set *Ghent Altarpiece*, publicly available at the website of the *Closer to Van Eyck* project. Image copyright: Ghent, Kathedrale Kerkfabriek, Lukasweb.**

a much more elaborate description of the overall approach but it also contains important novel technical improvements including the proposed compensation approach for accurate detection of crack boundaries, efficient re-training of the model and a case study conducted to support an actual painting conservation problem. Figure 1 illustrates parts of the multimodal acquisitions of the *Ghent Altarpiece* used in our study, while additional data have been acquired during the ongoing conservation treatment of this masterpiece as detailed later on.

This paper is structured as follows: Section II introduces the problem of crack detection in paintings in general, and the particular dataset used in our study. Basic concepts behind convolutional neural networks and their application to multimodal images are reviewed briefly. In Section III. The proposed approach is presented in Section IV and the experimental evaluation on a portion of the *Ghent Altarpiece* and comparative analysis with the related state-of-the-art methods is in Section V. In Section VI, we conduct a case study where crack detection is employed to diagnose the state of a panel painting and to provide objective supporting material for decisions made in the actual conservation process of a master painting. Section VII concludes the work and gives possible directions for further research.

II. CRACK DETECTION CHALLENGE

This section discusses the main problems associated with the detection of cracks in panel paintings. It also provides details of the multimodal dataset used further on in our study.

A. CRACKS IN PANEL PAINTINGS

Cracks are the most common type of deterioration found in paintings and can provide invaluable indications in the assessment of their authenticity. This is due to the fact that the cracks represent a kind of record of the history of the painting's deterioration. Crack patterns, and possibly their evolution over time, provide crucial information to art conservators about the state of the painting and possible causes of its degradation. As such, crack detection constitutes a vital input to virtual restoration. The reasons for crack formation, in addition to environmental factors, are largely related to the materials used by the artist. Hence, craquelure appears in a large variability of shapes from simple linear, to complex web-shaped or even seemingly random.

Automatic crack detection in paintings is far more challenging than crack detection on visually more uniform surfaces, such as roadways. Various thin and line-like painted objects may appear similar to cracks and in some cases, cracks are not visible at all in digital photographs because their color appears the same as the background. Figure 2 illustrates some of these challenges.

B. DATASET

We report the results of our work on a multimodal dataset of the *Ghent Altarpiece*[1] composed of extremely high-resolution images taken in several modalities: macrophotography, infrared macrophotography (IRP) and X-radiography (X-Ray) images. Examples of visible and infrared digital macrophotography are shown in Figure 3. The painting was captured in sections of 15 × 20 centimeters with a Hasselblad H4D-200MS 50 megapixel camera (with a CCD sensor of 49.1 × 36.7 mm), equipped with a Hasselblad 120mm macro lens, resulting in images of 8176 × 6132 pixels. Lighting was produced using Broncolor Scoro generators of 3200 Watt/s.

Infrared modalities reveal underdrawings and often show cracks more clearly than the visual images, due to a reduced contrast of some painted objects. Figure 3 also includes an example of X-radiography from the same data set. X-Ray images reveal valuable information about the structural condition of a painting. The high penetration of X-Rays can reveal the wood grain and splits in the oak support panels,

[1] http://closertovaneyck.kikirpa.be/ghentaltarpiece/

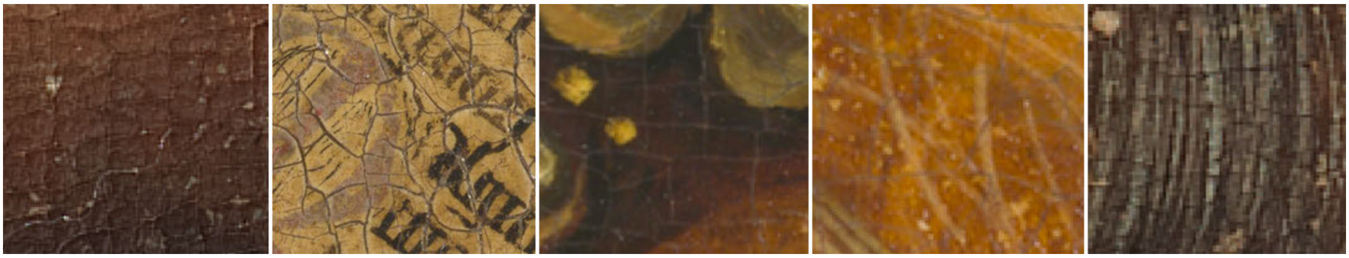


FIGURE 2. Examples of different crack appearances in paintings.

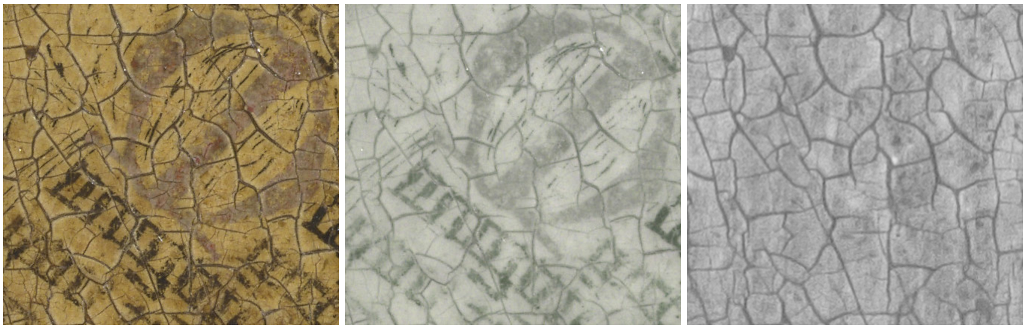


FIGURE 3. A detail of the panel *Virgin Annunciate* in the *Ghent Altarpiece* in different imaging modalities. Left to right: Visual and infrared macrophotography and X-radiography images.

as well as cracks and losses found in several paint layers. However, the interpretation of these images is often difficult due to spatial distortions and a severe periodic brightness distortion on panels, caused by hardwood lattices (called cradling) attached to the backside.

III. METHODOLOGY

This section reviews basic concepts of convolutional neural networks and discusses their application to multimodal data.

A. A DEEP CNN FOR MULTIMODAL DATA

Over the recent years, deep learning led to tremendous improvements in image classification [19]–[22]. Among various deep learning architectures, the convolutional neural network (CNN) is dominant in image processing applications because it cleverly exploits non-local self-similarities in images to greatly reduce the amount of parameters compared to a fully connected neural network. Figure 4 illustrates the basic architecture of a CNN, showing alternating layers of convolutions with kernels (that are shared across the image) and pooling layers. The convolutional layers produce feature maps while pooling layers progressively reduce the spatial size of the feature maps, increasing the receptive field in the subsequent layers [23], [24]. This basic architecture was proposed in the nineties [25], and an overview of recent developments can be found in [26].

The core of a CNN is the convolutional layer. The convolution operation is in actual implementations typically replaced by cross-correlation: a filter mask called kernel is moved over the image and the sum of products between the kernel coefficients and pixel values is computed at each location. Typically, multiple kernels are used in each layer. Assuming that kernels are sliding with the step of one pixel, i.e., with stride equal to one, the network output in the k-th feature map of the l-th layer, at spatial location (i, j) is:

$$x_{i,j}^{l,k} = \rho\Big(\sum_{d=1}^{D} \sum_{p=0}^{H^l-1} \sum_{q=0}^{W^l-1} w_{p,q}^{l,k,d} \cdot x_{i+p,j+q}^{l-1,d} + b^{l,k}\Big) \qquad (1)$$

where ρ denotes the activation function, d indexes the input feature map, H^l and W^l are spatial sizes of the kernels composed of the weights $w_{p,q}^{l,k,m}$, and $b^{l,k}$ is the bias term.

The existing approaches for applying CNNs to multimodal data can be divided into two main types. The first approach trains a separate convolutional neural network for each input modality [27] as illustrated in Figure 5(a). Training in such a structure is typically done sequentially, on separate streams of input data. After all neural networks have been trained, their fully connected layers are united into one compound vector [28], or alternatively, the most discriminating characteristics are extracted from each vector [29]. This type of architecture has been applied to RGB-D [30], [31] and audio-visual data [32], [33].

The second type of multimodal architecture (see Figure 5(b)) is trained on all the input modalities jointly [34]. This type of architecture was applied to hyperspectral image processing in [35]–[37]. We adhere to the second approach because it allows us to learn and exploit the correlation that exist among different imaging modalities.

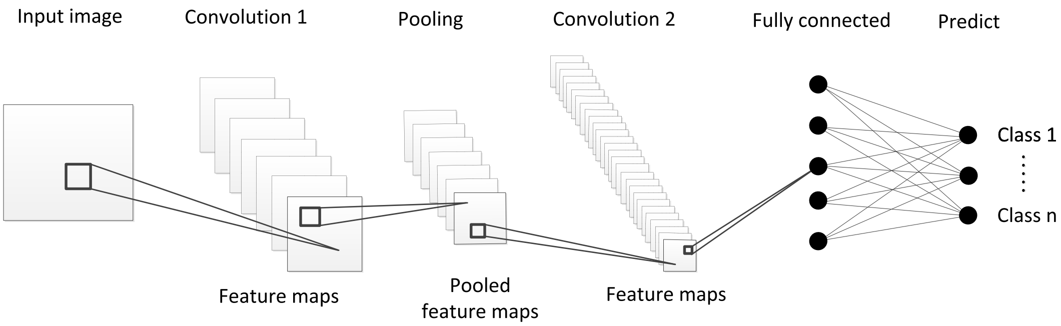


FIGURE 4. A generic concept of a classifier based on a convolutional neural network (CNN).

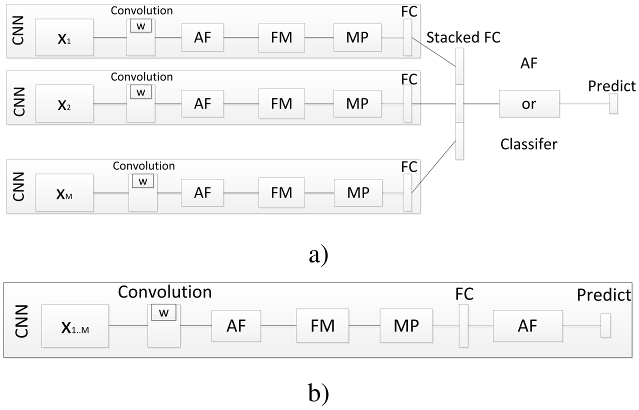


FIGURE 5. Different types of CNN architectures for the classification of multimodal data, where: x - source data, w - kernel, AF - activation function, FM - features map, MP - layer subsampling, FC - fully connected layer.

IV. PROPOSED METHOD

In the very high-resolution scans of paintings that we are dealing with, it is of interest to avoid running the deep learning model on image regions that can be confidently identified (by means of lightweight processing) as crack-free. This way, the whole process can be significantly accelerated, facilitating user interaction and avoiding unnecessary computational burden. Thus, the proposed method consists of two processing stages: (i) a morphological filtering stage and (ii) a classification stage by means of a convolutional neural network. The morphological filtering essentially ensures that the amount of pixels to be classified in the second stage is strongly reduced as only those image regions that cannot be safely rejected as crack-free are retained for further CNN-based processing. We train a convolutional neural network architecture with hand-labeled data. The network trained on a given set of paintings can be applied to a new painting, possibly with re-training when new annotations become available. We feed all the input modalities simultaneously to our network, like in the architecture depicted in Figure 5(b). As the experimental results will demonstrate, this choice allows for rapid training and re-training of the model. We extend the standard CNN architecture by introducing a compensation method that successfully eliminates excessive thickening of the crack boundaries, as detailed in Section IV-D.

A. GENERAL OVERVIEW OF THE PROPOSED METHOD

The general scheme of the proposed method is illustrated in Figure 6. The morphological filtering block, which makes an initial rough selection of crack candidate pixels, can be omitted when running the proposed method. However, its use can significantly reduce the overall computational cost, as well as slightly reduce the number of false positives. The proposed convolutional neural network architecture, trained on manually labelled data, is used to further classify selected pixels and to remove false positives. The proposed boundary correction approach based on the introduced shift coefficient improves the final classification result, as it will be demonstrated in the experimental section. Each processing block is explained in detail next.

B. PREPROCESSING WITH MORPHOLOGICAL FILTERING

Using multimodal input data at high resolutions can render the resulting computational complexity of machine learning infeasible for practical use. To alleviate this problem, we introduce a preprocessing stage based on morphological filtering. Morphological filtering has been widely used in crack detection, e.g., in [2], [4], [5]. In our approach, the role of this stage is to divide the input data into two classes, i.e. pixels possibly belonging to a crack (which will later be refined in a classification stage) and background (non-crack) pixels that will not be subjected to further classification. The particular choice of the morphological operators allows for false positives (as these can later be removed by the CNN model) while ensuring that false negatives are very unlikely (i.e., the pixels rejected as non-cracks should indeed be non-cracks with overwhelming probability). With such a mode of operation, the morphological pre-processing stage actually does not compromise the final crack detection accuracy.

We obtain a preliminary selection of crack pixels by feeding each modality to a morphological ''bottom-hat'' and/or ''top-hat'' operation, emphasizing bright objects and dark objects, respectively. To recall, ''top-hat'' filtering subtracts the result of the morphological ''opening'' from the original

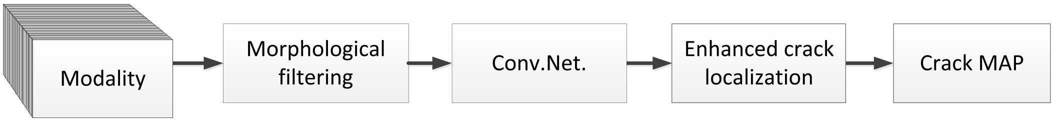


FIGURE 6. General schematic of the proposed method.

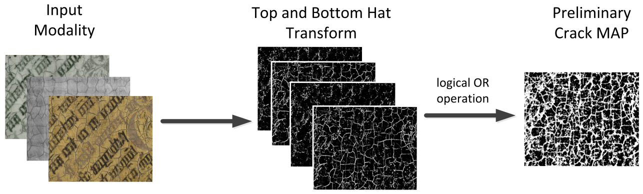


FIGURE 7. Workflow of the construction of a binary mask for the initial detection of crack pixels.

image and "bottom-hat" subtracts the result of the morphological "closing" from the original image.

With an appropriately chosen structuring element,[2] the cracks in the image will be highlighted, along with other small-scale objects. Since in the infrared and X-ray images cracks are always dark, only a "bottom-hat" operation is performed on those modalities. In visual photographs (i.e. RGB images) cracks can appear both dark and bright, so both the "bottom-hat" and "top-hat" operations are applied on those images. Each filtered result is converted to a binary image by thresholding, where the thresholds are determined for each modality separately using the Otsu method [7]. The resulting binary maps are combined into one so-called binary crack map using the logical "OR" operator, as shown in Figure 7.

C. PROPOSED CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE

The task of the classifier is to assign each of the candidate pixels, selected after the preliminary morphological filtering, to one of two possible classes: *crack* or *non-crack* (i.e., the background). We designed the architecture of our convolutional neural network with two goals in mind: (i) ensuring high classification accuracy, and (ii) enabling efficient re-training when new annotations become available. To this end, we maximize the depth of the network by removing the sub-sampling layers and reducing the size of the kernels in the convolution layers, as described in [21]. The overall architecture is depicted in Figure 8.

The input of our network is a tensor $\mathbf{x}^0_{u,v} \in \mathbb{R}^{n\times n\times M}$ composed of M two-dimensional $n \times n$ patches centred at (u, v) from the M available modalities. In the first convolution layer we have:

$$x^{1,k}_{u,v} = \rho(\sum_{m=1}^{M}\sum_{p=0}^{H^l-1}\sum_{q=0}^{W^l-1} w^{0,k,m}_{p,q}\cdot x^{0,m}_{u+p,v+q} + b^{l,k}) \qquad (2)$$

For all subsequent layers (i.e. $l > 1$), feature maps are calculated as in (1).

Our network produces 100 feature maps in the first convolutional layer, 200 in the second and 300 in the third layer. The first fully connected layer has 300 neurons. The spatial size of the filters for convolution operations remains the same across all layers. Exponential linear units (ELU) were used for the activation functions ρ [38]:

$$\rho(x) = \begin{cases} x & \text{if } x > 0 \\ a(e^x - 1) & \text{if } x \leq 0, \end{cases} \qquad (3)$$

where $a > 0$ is a hyperparameter that controls the value at which the ELU saturates for negative inputs. This activation function retains all the advantages of ReLU, but is

[2] A disk-shaped structuring element with a diameter of 3 pixels was used for all experiments.

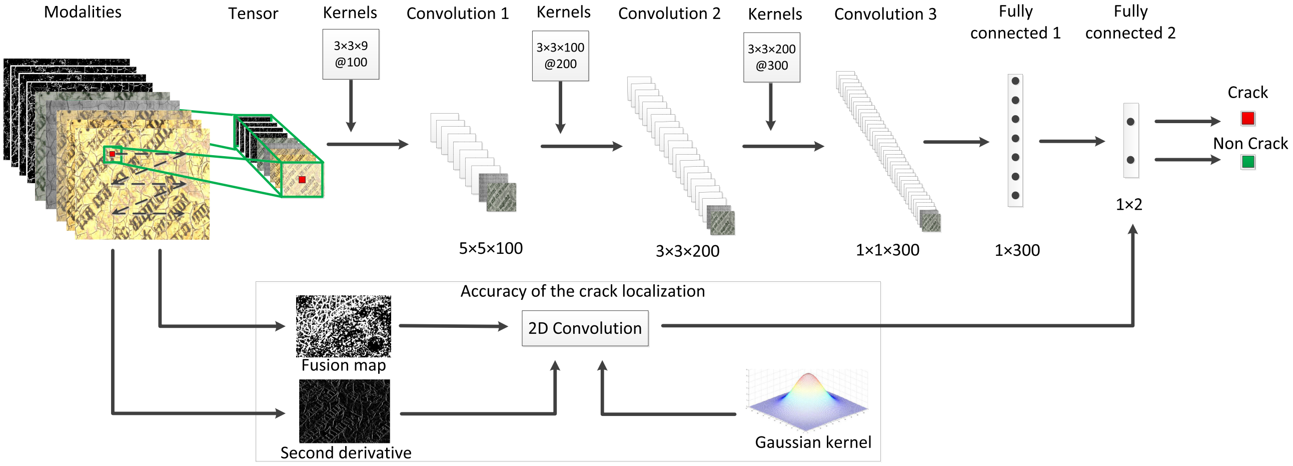


FIGURE 8. The proposed convolutional neural network architecture for crack detection in paintings with multimodal data.

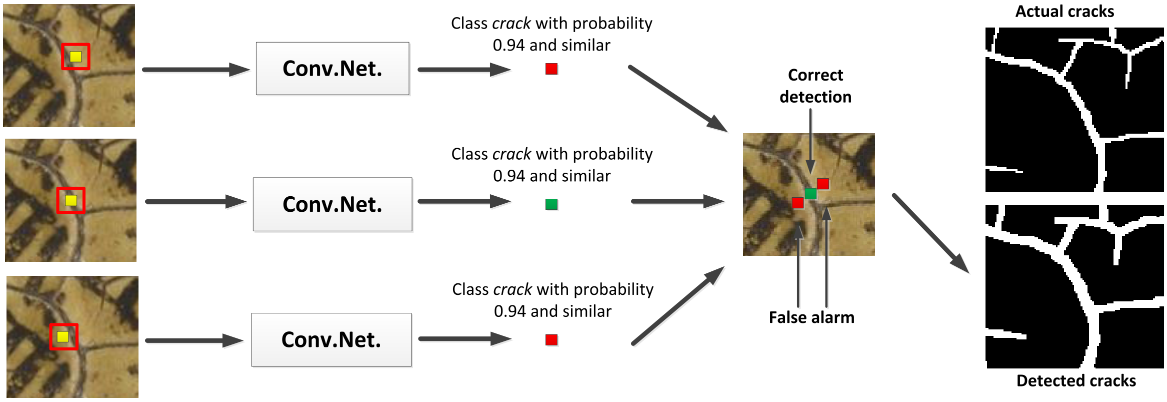

Conv.Net.
Conv.Net.
Conv.Net.
Class *crack* with probability 0.94 and similar
Class *crack* with probability 0.94 and similar
Class *crack* with probability 0.94 and similar
Correct detection
False alarm
Actual cracks
Detected cracks

FIGURE 9. An illustration explaining the problem of excessive thickening of crack boundaries using the traditional CNN approach. Yellow marks the central pixel of the window from which a tensor is formed for classification. After classification, this central pixel is assigned a response from the CNN, which corresponds to the response for the tensor formed around it. In the detection result, red denotes a false alarm, and green correctly detected crack pixel.

more resistant to noisy data, and results in a higher classification accuracy compared to activation functions such as ReLU [39], leaky ReLU [40] and shifted ReLU [38]. Furthermore, the authors of [38] have shown that when using the ELU activation function, the use of normalization in mini-batches [41] does not give a significant advantage. All kernels are initialized randomly at the beginning of the training procedure. The last layer of the architecture consists of a *softmax* function. Let z_ι, $\iota \in \{crack, non\text{-}crack\}$, denote the outputs of the second fully connected layer. These are the output scores that the network gives to each of the classes for a given input tensor. The *softmax* translates these scores to:

$$y(z_\iota) = \frac{e^{z_\iota}}{\sum_\kappa e^{z_\kappa}}, \tag{4}$$

When training the neural network, we use the Adam optimizer [42], which is more effective than standard backpropagation. The main difference is the use of gradient change history from previous iterations. The weights are updated as follows:

$$\Delta w(\tau) = \eta\big(g(\tau) + \epsilon w(\tau - 1)\big) + \mu \Delta w(\tau - 1), \tag{5}$$

where Δw is the weight change ϵ is a regularization coefficient, μ is the momentum, η is the learning rate, and τ is the current iteration. The parameter ϵ allows to avoid overfitting by imposing a "penalty" for excessive growth of the weights. The μ parameter gives inertia to the weight change, which avoids getting stuck in local minima. The variable $g(\tau)$ maintains the history of the gradient changes as follows:

$$g(\tau) = \frac{\mathcal{D}(\tau)}{1-\beta}\sqrt{\frac{1-\alpha}{\mathcal{S}(\tau)}}, \tag{6}$$

where $S(\tau)$ and $D(\tau)$ are defined as:

$$\mathcal{S}(\tau) = \alpha\mathcal{S}(\tau-1) + (1-\alpha)\nabla E(\tau)^2; \quad \mathcal{S}(0) = 0, \tag{7}$$

$$\mathcal{D}(\tau) = \beta\mathcal{D}(\tau-1) + (1-\beta)\nabla E(\tau); \quad \mathcal{D}(0) = 0, \tag{8}$$

The parameters α and β are fixed to 0.999 and 0.9, respectively, and $\nabla E(\tau)$ is the error gradient from the previous iteration. Using this optimizer allows to quickly achieve a global minimum error, due to the adaptive change of the weight update step.

To evaluate and minimize the losses of our network, we use the cross-entropy function, defined as:

$$Loss(y_\kappa, y'_\kappa) = -\frac{1}{\mathcal{K}}\sum_{\kappa=1}^{\mathcal{K}}\left[y_\kappa \cdot log(y'_\kappa) + (1-y_k)\cdot log(1-y'_\kappa)\right] \tag{9}$$

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

D. ENHANCING THE ACCURACY OF THE CRACK LOCALIZATION

A common limitation of standard CNN architectures, including those that were employed in crack detection on road surfaces (and similar surfaces) [9], [10], [16], [43], is their inability to accurately delineate the crack boundary. The essence of the problem lies in the fact that the tensor containing a crack pixel gives a high response, regardless of the position of the crack pixel inside the tensor. As a result, crack detection with CNNs suffers from excessive thickening at the boundaries of the detected cracks. Figure 9 depicts an example of a sliding window along a crack. Notice that the probability that the tensor contains a crack remains approximately unchanged during the window movement, which causes neighboring pixels to be marked as part of a crack. The actual crack width cannot be restored by some simple morphological thinning operation. This is a serious limitation since the width of cracks in paintings varies a lot from one place to another and thus providing only the crack centerline or a skeleton means loosing important information, which is unacceptable for most tasks where crack detection is required.

Applying morphological filtering prior to CNN-based classification can alleviate this problem to some extent. This is simply because the preprocessing step already rejects some pixels that do not belong to cracks. However, some actual cracks may be lost as well. In the literature, multiscale convolutional networks [44]–[47] are often used to solve this problem. However, the computational complexity of these models is still prohibitive for processing very high-resolution data in multiple modalities as we need to do, especially when user interaction and re-training processes are a requirement.

We propose a low-complexity solution for accurate localization of small objects, that can be combined with any CNN architecture. The main idea is to account for the position of crack pixels relative to the center of the tensor through a *shift coefficient*. We define the shift coefficient at location (i, j) as follows:

$$s_{i,j} = \arg\max_{\sigma \in S} \sum_{p=0}^{H-1} \sum_{q=0}^{W-1} F_{i,j} G^{\sigma}_{p,q} \big(X_{i-2+p,j+q} + Y_{i+p,j-2+q} \big) \tag{10}$$

where $F_{i,j} \in \{-1, 1\}$ is a binary label denoting the type of crack, taking the value 1 for dark and -1 for bright cracks, $G^{\sigma}_{p,q} = \frac{1}{2\pi\sigma^2} e^{-\frac{p^2+q^2}{2\sigma^2}}$ is a symmetric lowpass Gaussian filter, X and Y are finite-difference approximations of the second derivatives of an input image channel[3] in horizontal and vertical directions, respectively.[4] S denotes the set of possible σ values, with a range depending on the size of the cracks.

We determine the labels $F_{i,j}$ by comparing the values of the "bottom-hat" and "top hat" filtering results. Both morphological filters are applied to the digital visual macrophotograph. Let $\xi^t_{i,j}$ denote the value of the "top-hat" filtered image at location (i, j) and $\xi^b_{i,j}$ the value of the "bottom-hat" filtered image at the same location. For robustness, we take the maximum value within a small window centred at (i, j). Knowing that bright cracks give a larger response in the "top-hat" filtered image and the dark ones in the "bottom-hat" image, we assign the labels as follows: $F_{i,j} = 1, \quad \text{if} \quad \xi^b_{i,j} > \xi^t_{i,j}; \quad \text{else} \quad F_{i,j} = -1.$

We refer to the binary map $F_{i,j}$ at all image locations as the fusion map. Figure 10 depicts an example of a fusion map. The obtained shift coefficient imposes a penalty on the falsely widened regions of the detected cracks, as shown in Figure 11.

In particular, at the final classification stage, the shift coefficient value is multiplied with the estimated crack probability in the second fully connected layer of the convolutional neural network.

E. TRANSFER LEARNING

Convolutional neural networks significantly simplify the task of re-training when new training data becomes available, in comparison to more traditional machine learning methods [48], [49]. Transfer learning and fine-tuning are two main re-training strategies have been followed in the literature. In the first case, the trained network generates an N-dimensional feature vector, which is then used as a descriptor of the classes of interest also for other data sets that need to be processed and for which no labeled data are availble. A CNN is then used as a feature extractor for a new classifier, which can be another CNN, but also a different type of a classifier such as SVM, AdaBoost, or any other. In the case of fine-tuning all layers or some of the layers, usually the last ones as they are the most discriminative, are re-tuned using the labeled data from a new data set. In this strategy, the trained network is used as a weight initializer. The choice of re-training strategy depends on the number and type of additional data available for training. If the new data differs significantly in type from the source, then in practice the first strategy is typically used, or the second one by training only the last layers. If the new data is similar to the source data, then the second strategy is followed by re-tuning of all or only the last layers of the neural network.

In our work, we use both re-training strategies with a complete re-traning of the weights in all layers. This decision is motivated by two main factors: the original training data and the additional data are of similar type and the number

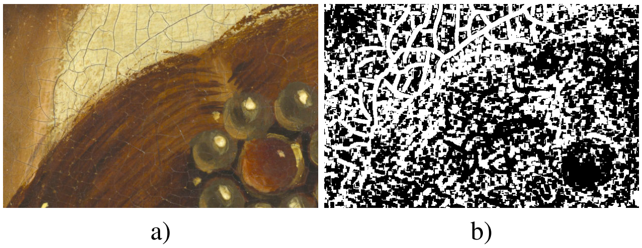


FIGURE 10. An example illustrating the construction of a fusion map. a) Original image (a detail from the panel *Singing Angels*). b) Fusion map. Dark cracks appear as white color and light cracks appear as black color on fusion map.

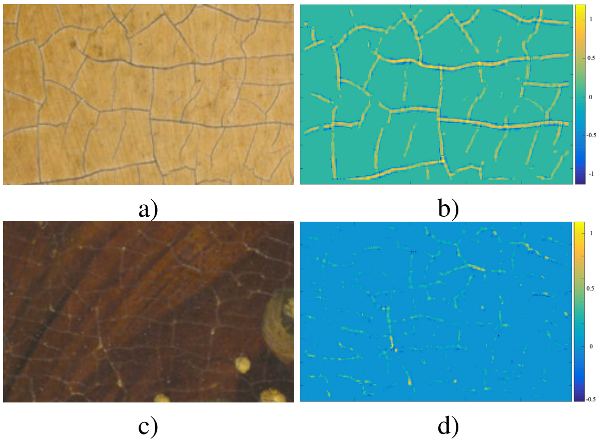


FIGURE 11. Visualization of penalty maps for the pixels which are assumed to be part of a crack. a) An original image with dark cracks and b) the corresponding penalty map. c) An image with bright cracks and d) the corresponding penalty map. As the distance from the center of the crack increases, the confidence decreases (the penalty increases).

[3] Dark cracks are better visible in the "Blue" and the bright ones in the "Red" channel of visual macrophotography, so it is best to select one of these adaptively, depending on $F_{i,j}$, although a grayscale version of the original image can also be used regardless of the crack type.

[4] In order to make sure that the finite difference matrices have the same size, their rows and columns are padded with zeros.

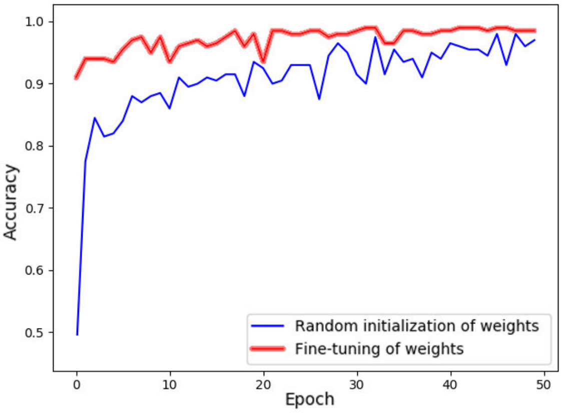


FIGURE 12. The evolution of the learning rate of the convolutional neural network with different types of initialization of weights.

of classes remains unchanged (hence no need to change the number of outputs in the last fully connected layer). For training, we combine the original and new training data, and then use them to re-train the neural network. This approach has important advantages in practice. Firstly, it improves the quality of classification continuously as new training data are obtained; secondly, combining data allows to keep the previously achieved result. We recommended to keep the full amount of the source training data, because otherwise the accuracy of crack detection, achieved during the initial training, might decrease. Figure 12 compares the evolution of the learning rate in cases when the weights of the neural network are initialized randomly, and using a previously trained neural network. The results show that the use of a pre-trained network to initialize weights allows to significantly speed up the training process.

F. PREPARATION OF TRAINING DATA

Preparing data to create a training database is carried out by labelling some parts of the image as *crack* or *non-crack* (i.e., *undamaged*) classes. Figure 13 shows an example of such a marked image. Green labelled pixels are crack pixels, and blue marks the undamaged image parts. A rectangular area of $n \times n$ pixels is extracted around each labelled pixel and the corresponding parts are extracted from all the available modalities (in our experiments: visual photographs, infrared photography and X-Ray images). This data preparation approach together with the ability to fine-tune the convolutional neural network with new annotations allows to improve the classification result continuously.

V. EXPERIMENTAL RESULTS

We report crack detection results on parts of the multimodal dataset of the *Ghent Altarpiece* described in Section II-B. Because of the large dimensions of the images, we report results only on relatively small parts of the following panels: *Virgin Annunciate*, *John the Evangelist* and *Singing Angels*. For evaluating the quality of the detected cracks numerically, we used ground truth data from [8].

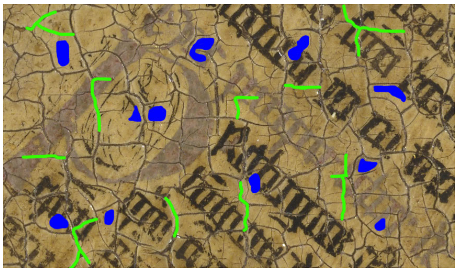


FIGURE 13. An example of a manually marked image as preparation for training. The image is a detail from the panel ***Annunciation of the Virgin Mary***. Green marks representatives of crack pixels and blue marks non-crack examples.

The full proposed method for crack detection, including the morphological prefiltering and the refinement of the improved crack boundary detection, will be denoted as MCNC.

We compared our MCNC method with its reduced version without the improved crack boundary localization – MCN, and against approaches using fully connected neural networks (NN) [50], Boosting methods (ADA) [51], support vector machines (SVM) [52], and the Bayesian Conditional Tensor Factorization method (BCTF) [8]. We also include comparisons with two other deep learning methods: a CNN-based method that was proposed for crack detection in roads [9] and a deep feature fusion network (DFFN) classifer from [53].

We explain next the experimental setup, with the exact configuration of our network, which is used in three scenarios: default training (Section V-A), training with subsequent fine-tuning (Section V-B) and transfer learning (Section V-C). We implement the CNN architecture described above with two hidden layers, which produce twelve and eighteen feature maps respectively. The logistic sigmoid is used as an activation function. Both the first and the second layer use kernels of size 4×4. Figure 14 shows an input image together with the first-layer features, and Figure 15 shows the corresponding feature maps in the second hidden layer. It can be seen that in some of the feature maps cracks are already rather well separated from the background. Based on this, the following parameters were used for training in all the experiments: the mini batch size equals 100, the kernel spatial size is 3×3, the first fully connected layer has 300 neurons, and the learning rate is 0.0001. The minimum classification error was achieved on average after 50-70 epochs.

The tensor that is used for classification contains the three color channels of the visual macrophotographs, the X-Ray image and single-channel infrared macrophotograph, as well as a fourth ''modality'' obtained by morphological filtering, adding up to $M = 9$ modalities. The grayscale ''top/bottom-hat'' transform is designed to emphasize elements smaller than the size of a chosen structuring element.[5] Hence, the filtered images, containing emphasized cracks (along with other

[5] As motivated earlier, the top- and bottom-hat transform are used for the visual image, and the ''bottom-hat'' transform for the X-Ray and IRP images. We used a disc-shaped structuring element with a diameter of 3 pixels.

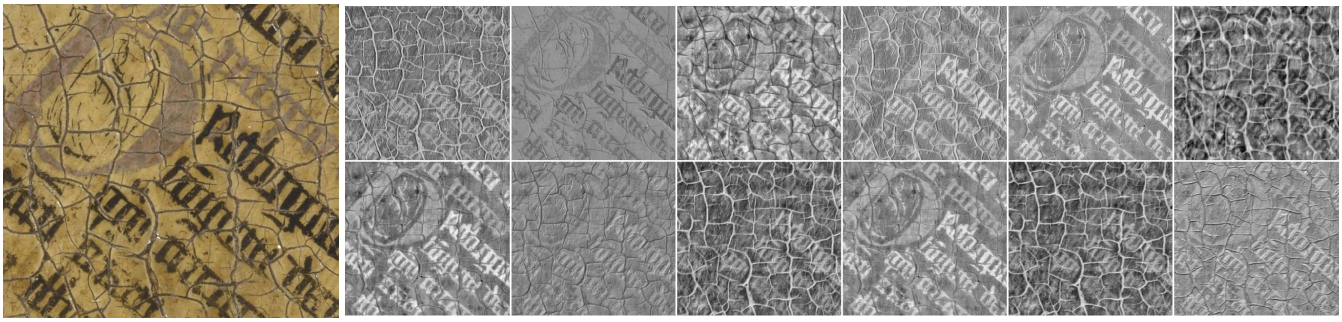


FIGURE 14. An original image (left) and twelve features (right) in the first convolutional layer.

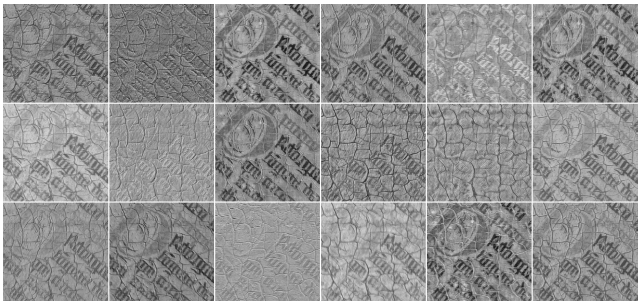


FIGURE 15. Features in the second convolutional layer corresponding to the input image from Fig. 14.

small-scale objects), are good candidates to serve as additional channels/modalities for our method. The resulting tensors are of dimensions 7 × 7 × 9. We compute and apply the shift coefficient only with tensors that contain a crack with a probability of 0.9 and above.

The optimal spatial tensor size for training the convolutional neural network was obtained experimentally. Figure 16 shows results of a comparative experiment where the experimental CNN architecture was applied with different tensor sizes. The classification was performed only in regions that were not eliminated as crack-free by the morphological filtering step. We observe that the spatial window should ideally cover the entire crack width and a small portion of the background. Excessively large window sizes give rise to two problems: a longer training time of the neural network and more likely false detections.

The approaches using the neural network, boosting method and support vector machine are built using Matlab 2016b, with three variants of input data: a raw data tensor, a raw data vector and a vector of LBP descriptors [54]. In the tensor and vector variants, the feature tensor/vector is formed by from all the N modalities, without additional calculations (i.e. using the raw data). The resulting vector input has 9 elements and the tensor input contains 441 elements. The descriptor version uses as input a vector of 20 elements, obtained by combining the results of LBP texture descriptors (producing a vector with 18 elements), mean value and standard deviation. These descriptors are applied in the local window of 7 × 7 pixels, for each of the N modalities. Subsequently, all the obtained values are combined into one vector which has 180 features for a tensor with dimensions 7 × 7 × 9.

The configuration of the neural network adheres to the suggestions from [55]. In particular, it has 3 layers with sigmoidal activation functions with 250 neurons in each layer for the tensor version, 10 neurons for the vector version and 100 neurons for the descriptor version. 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. In the method using the support vector machine classifier, a "linear function" is used as the kernel. The parameter of the Karush-Kuhn-Tucker (KKT) complementarity condition is 0.05 for all versions.

The method proposed in [9] has 4 convolutional layers and two fully connected layer. All convolutional layers produce 48 feature maps. The first fully connected layer has 200 neurons. The linear rectification unit (ReLU) is used as the activation function. The stochastic gradient descent (SGD) method was used for training.

The method proposed in [53] has 9 convolutional layers and one fully connected layer. The convolutional layers are divided into three groups: a low-level residual block, a mid-level residual block, and a high-level residual block. The layers of the first residual block produce 16 feature maps, the layers of the second residual block produce 32 feature maps, and the layers of the third residual block produce 64 maps. The linear rectification unit (ReLU) was used as the activation function and the stochastic gradient descent (SGD) with batch normalization was employed for training.

For the BCTF method, a pixel is considered to be part of a crack when it has a crack probability of 0.5 or higher. BCTF uses a vector with 208 vector elements composed of the RAW multimodal data, as well as their pre-processed versions obtained with different spatial image filters. Bayes method is used for classification.

To evaluate the effectiveness of the methods, we use the following metrics:

$$FA = \frac{FP}{AlPx - DfPx}, \quad FM = \frac{FN}{AlPx - UdPx} \tag{11}$$

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F_1 = \frac{2 \cdot P \cdot R}{P + R} \tag{12}$$

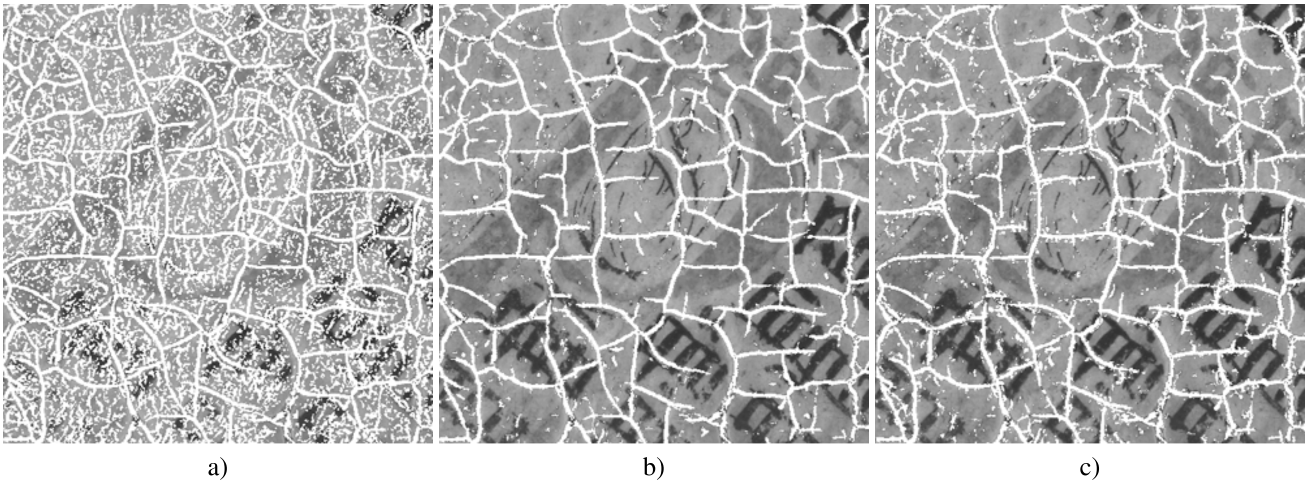


FIGURE 16. **The influence of the input tensor size. a) An initial crack map after the morphological filtering step and the final classification results with tensor spatial sizes of b) 7 × 7 and c) 15 × 15.**

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.

The results presented below are divided into three parts. The first part presents the results obtained by classical training, where learning occurs on training samples taken from the image that needs to be processed. The second part presents the results obtained through classical training, followed by an additional fine-tuning using new training data (i.e., re-training with newly available annotations). The results described in the third part are obtained by training the convolutional neural network on other panels (different from the one that is being processed), which demonstrates a transfer learning ability. Additionally, we show how the results improve after adding relatively few annotations from the panel being processed. This corresponds to the most realistic scenario for practical use. All experiments are performed on a laptop with an Intel Core i7 @ 2.8GHz CPU and 16GB RAM, GPU Nvidia 1050ti for accelerated training.

A. DEFAULT TRAINING OF THE NETWORK

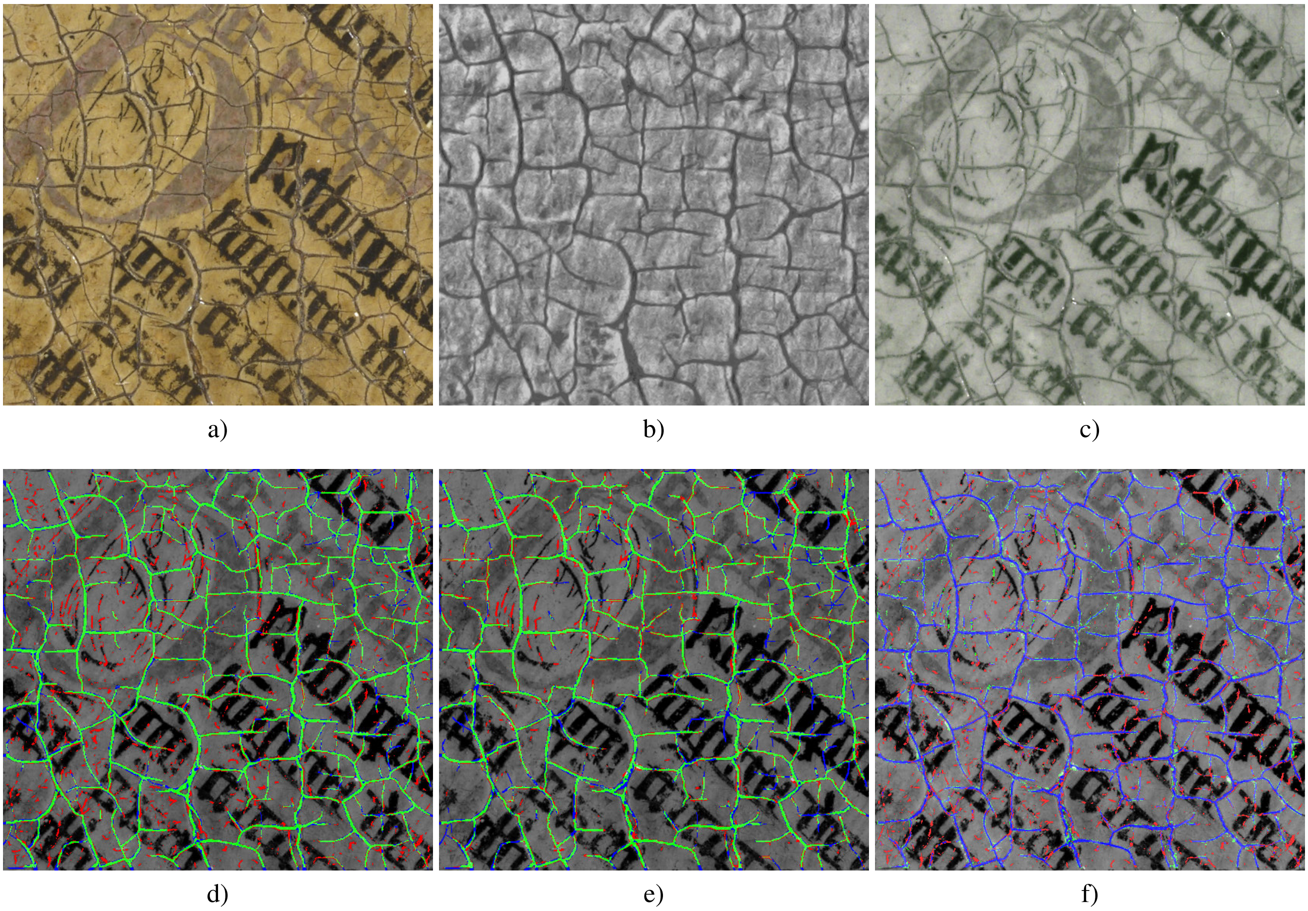
The first image we consider is part of a book in the *Virgin Annunciate* panel. From Figure 17, the variability in crack types becomes clear: there are fine, white specks at the edge of cracks which are only visible in the color image, thick cracks with irregular contrast in the color image, and cracks crossing letters. This selection includes painted objects with very similar properties to cracks, which can only be discriminated from cracks by using the different modalities at our disposal (see Figure 17(a-c)).

For training, 20,000 samples of crack pixels and 19,000 samples of undamaged areas were used. According to this result we can conclude that the method based on convolutional neural networks is capable of locating cracks with an accuracy similar to the BCTF method, with slightly better precision P and F_1 scores compared to BCTF. Additionally, it is important to note that the vast majority of false alarms in the BCTF method are falsely detected cracks, while in the MCN method and its improved version MCNC false alarms consist of excessive thickening of crack boundaries.

TABLE 1. **A comparison of different methods for the *Annunciation virgin Mary* panel. Corresponding index: *d*-descriptors, *t*-raw data tensor, *v*-vector through modalities.**

Method	Recall	False alar.	False miss.	Precision	F_1-m.
ADA$_d$	**0.8695**	0.1183	**0.1305**	0.5013	0.6360
ADA$_t$	0.8693	0.0920	0.1307	0.5636	0.6839
ADA$_v$	0.8357	0.0734	0.1643	0.6091	0.7046
SVM$_d$	0.8471	0.0832	0.1529	0.5822	0.6901
SVM$_t$	0.8530	0.0912	0.1470	0.5612	0.6770
SVM$_v$	0.7990	0.0579	0.2010	0.6538	0.7192
NN$_d$	0.8468	0.0840	0.1532	0.5796	0.6882
NN$_t$	0.8655	0.0877	0.1345	0.5745	0.6906
NN$_v$	0.8333	0.0703	0.1667	0.6183	0.7099
CNN [9]	0.8481	0.0777	0.1519	0.5989	0.7020
DFFN [53]	0.7488	0.0422	0.2512	0.7081	0.7279
BCTF [8]	0.7896	0.0535	0.2104	0.6686	0.7241
MCN	0.8161	0.0540	0.1839	0.6741	0.7383
MCNC	0.7673	**0.0375**	0.2327	**0.7365**	**0.7516**

Several conclusions can be drawn from the results in Table 1: The classification methods based on features calculated from multiple imaging modalities yield better F_1 score than methods that are classifying stacked hand-crafted features calculated for each patch. The proposed MCN approach (with both MCN and MCNC variants) outperforms all the other methods in terms of precision and the F_1 metric. The improved algorithm with the more precise boundary detection (MCNC) reduces significantly false broadening of the detected cracks, which is numerically reflected in reduced false alarms, and in improved precision and F_1 metrics.



a) b) c)

d) e) f)

FIGURE 17. Comparative results of a Bayesian multimodal approach BCTF and the proposed MCNC on part of the panel *Annunciation virgin Mary*. a) RGB image, b) X-Ray image, c) IRP image, d) The BCTF crack map, and e) The MCNC crack map. Green – true positives, Red – false positives, Blue – false negatives. f) BCTF–MCNC comparison: Blue – detected by both methods, Red – BCTF only, Green – MCNC only.

B. DEFAULT TRAINING OF THE NETWORK WITH SUBSEQUENT FINE-TUNING

Now we analyse the case where the network is first trained with a certain, relatively small, amount of annotations and then subsequently re-trained adding new annotations. This scenario is of practical importance to the users (art conservators and restorers). The image that we consider is part of the *Singing angels* panel. This image is quite challenging for the detection of cracks due to several reasons: the low contrast of cracks in the visual image (see Figure 18(a)), the absence of some cracks in the IRP image (see Figure 18(c)) and in the X-Ray image and the noisy nature of the X-Ray image (see Figure 18(b)). An additional complicating factor for the detection of cracks is the slight spatial shift between all available modalities due to the registration procedure. A total of 17,000 samples of cracks and 22,000 samples of undamaged areas were used for training.

The results in Table 2 show excellent perfomance of tensor-based methods indicating hereby also their robustness to the present distortions in the input data including noise, low contrast and imperfect alignment of the imaging modalities.

TABLE 2. The comparison of different methods for the *Singing angels* panel. Corresponding index: *d*-descriptors, *t*-tensor with raw data, *v*-vector through modalities.

Method	Recall	False alar.	False miss.	Precision	F_1-m.
ADA_d	**0.6475**	0.1391	**0.3525**	0.4008	0.4951
ADA_t	0.5681	0.0739	0.4319	0.5246	0.5455
ADA_v	0.5734	0.0886	0.4266	0.4817	0.5235
SVM_d	0.5111	0.0756	0.4889	0.4927	0.5017
SVM_t	0.5922	0.0884	0.4078	0.4903	0.5365
SVM_v	0.5056	0.0805	0.4944	0.4742	0.4894
NN_d	0.5655	0.0877	0.4345	0.4809	0.5198
NN_t	0.6002	0.0865	0.3998	0.4993	0.5451
NN_v	0.5733	0.0845	0.4267	0.4936	0.5305
CNN [9]	0.6119	0.0999	0.3881	0.4680	0.5304
DFFN [53]	0.6242	0.0966	0.3758	0.4814	0.5436
BCTF [8]	0.6150	0.0905	0.3850	0.4941	0.5479
MCN	0.6340	0.0894	0.3660	0.5048	0.5621
MCNC	0.6083	**0.0681**	0.3917	**0.5622**	**0.5843**

The proposed MCN approach shows superior precision in the crack detection, and the benefit from the improved boundary localization is also evident here when comparing the numerical results of MCNC with MCN.

An important advantage of using a learning-based classifier with convolutional neural networks is that it enables

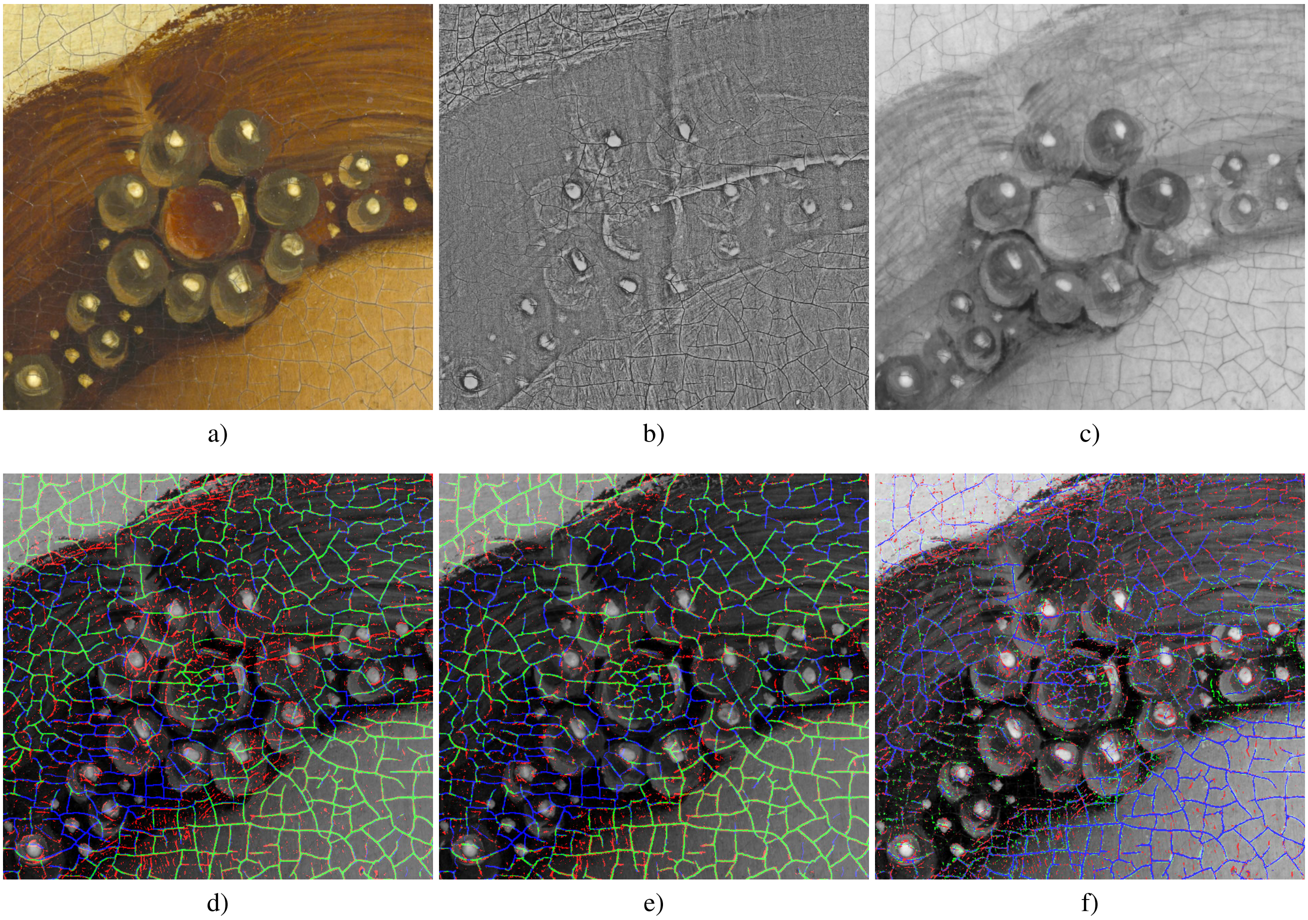


FIGURE 18. Comparative results of BCTF and the proposed MCNC on part of the panel *Singing Angels*. a) RGB image, b) X-Ray image, c) IRP image, d) The BCTF crack map, and e) The MCNC crack map. Green – true positives, Red – false positives, Blue – false negatives. f) BCTF–MCNC comparison: Blue – detected by both methods, Red – BCTF only, Green – MCNC only.

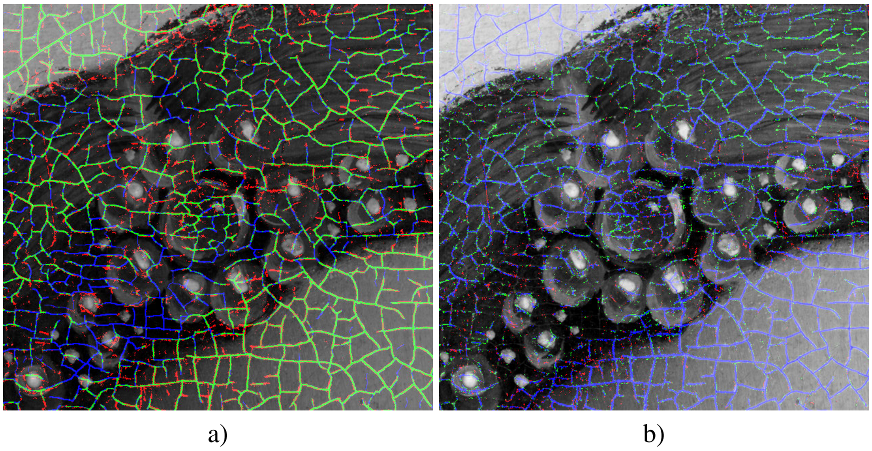


FIGURE 19. The effect of re-training MCNC with new training samples. a) Crack map of the re-trained method MCNC$_r$. Green – true positives, red – false positives, blue – false negatives. b) Comparison MCNC – MCNC$_r$. Blue – detected by both methods, red – MCNC only, green – MCNC$_r$ only.

continuously improving the results when new annotated data become available. To evaluate this re-training ability of our network, we augment the previous training set with some new training data. The new training data set consists of 13,000 samples of crack pixels and 8,000 samples of undamaged areas.

The detection result of this retrained method – MCNC$_r$ is shown in Figure 19 and numerical results are reported

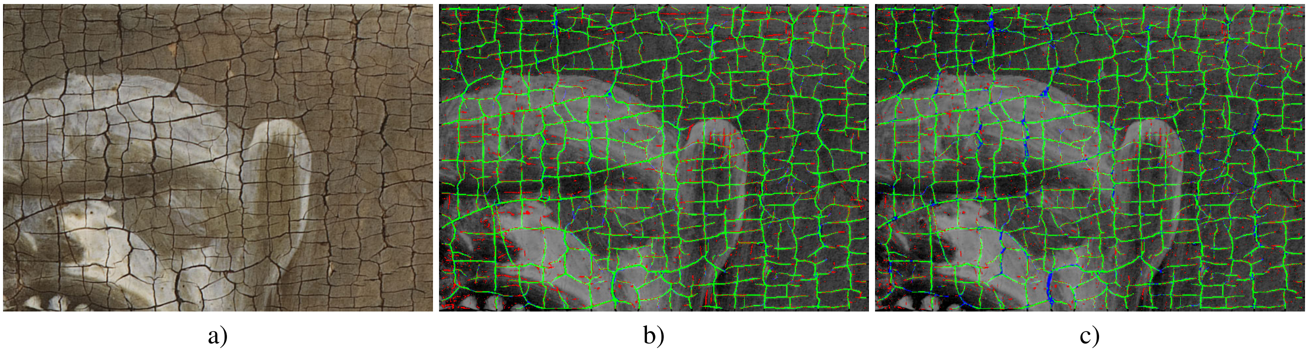


FIGURE 20. An example of using MCNC trained on different panels. a) Part of *John Evangelist* panel, b) The BCTF crack map, c) The result of MCNC trained on parts of other panels. Green – true positives, Red – false positives, Blue – false negatives.

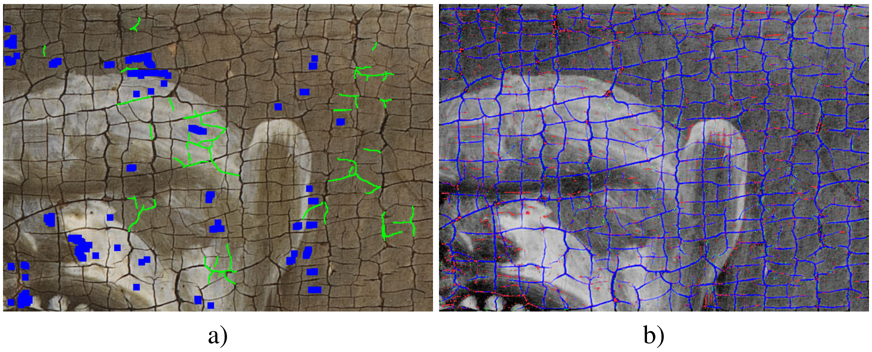


FIGURE 21. Comparative results of BCTF and MCNC$_r$. a) Part *John Evangelist* panel, with labelled cracks (green) and background (blue), b) BCTF – MCNC$_r$ comparison. Blue – detected by both methods, Red – BCTF only, Green – MCNC$_r$ only.

TABLE 3. Improving the MCN classification result by extending the training dataset. Corresponding index: *r*-re-training.

Method	Recall	False alar.	False miss.	Precision	F_1-m.
BCTF [8]	0.6150	0.0905	0.3850	0.4941	0.5479
MCN	0.6340	0.0894	0.3660	0.5048	0.5621
MCN$_r$	**0.6849**	0.0992	**0.3151**	0.4979	0.5766
MCNC$_r$	0.6570	**0.0734**	0.3430	**0.5624**	**0.6060**

in Table 3. Evidently, adding some amount of new training data led to a noticeably improved result, without the need for a complete re-training of the neural network. As it will be demonstrated next, using this procedure the convolutional neural network also adapts quickly to searching for cracks in new paintings that were not previously used for training.

C. EXAMPLE OF TRANSFER LEARNING

Here we apply the previously trained network to another panel (i.e., to a different image), first directly and then after re-training with relatively few annotations from the new panel. Figure 20(a) portrays the result of processing a small part of the *Johh Evangelist* panel, consisting of only the visual modality. This result is obtained with our MCNC method that was trained on different images, the results of which are presented earlier (*Annunciation virgin Mary* and *Singing Angels*, from Section V-A and V-B). It is important to note that the BCTF method in this experiment has been specially trained to detect cracks for this particular image. Figure 20(b) shows the result by superimposing the crack map obtained with BCTF and manual labeling of crack pixels. Figure 20(c) shows the result by superimposing the crack map obtained with the improved MCNC method and manual labeling of crack pixels.

As expected, the MCNC model that was trained on other paintings, different from the one that is being processed, was not able to detect all the cracks and yielded also a significant number of false positives. To improve on the result, the user can manually add training samples from the areas of the image where the cracks were missed, as well as from those areas where false positives appeared. Figure 21 shows the crack detection result after adding approximately 3,000 positive and 3,000 negative additional tensors. This is about 10% of the initially used training samples.

As indicated by the visual result in Figure 21 and the numerical results in Table 4, adding this relatively small amount of new training data improves the result significantly. Furthermore, this indicates that a pre-trained MCN can quickly be adapted for the detection of cracks in other panels.

The obtained result is now similar to the result of BCTF, which was especially trained and tuned for this panel. This clearly shows the ability of the proposed method to adapt to crack detection on new panels.

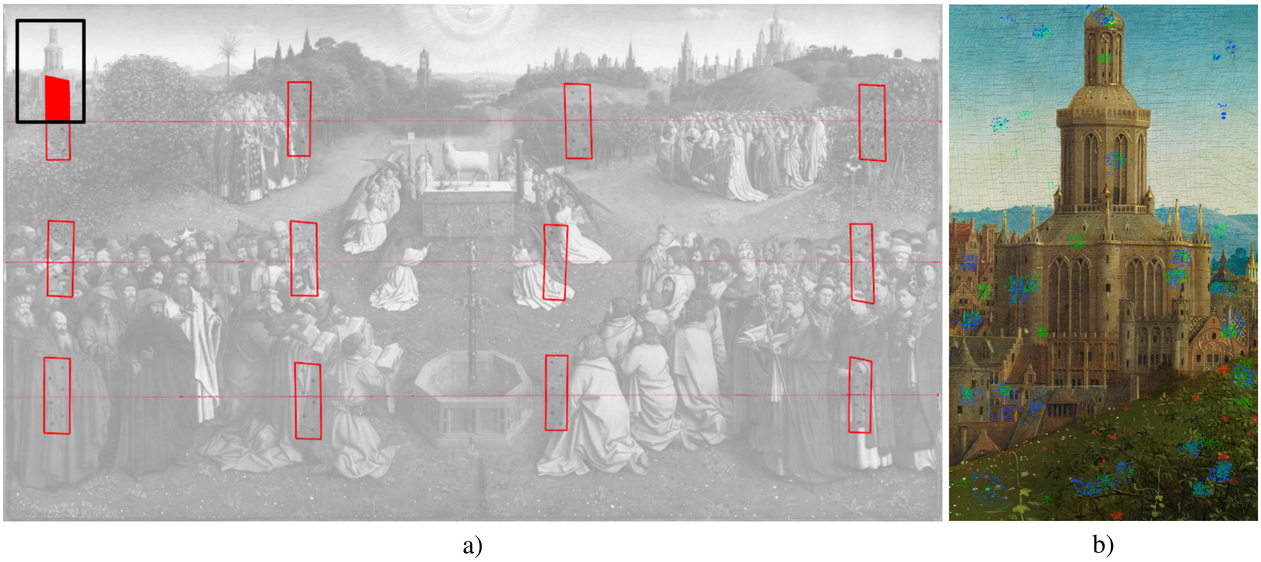


FIGURE 22. a) Positions of wooden reinforcements (marked in red) on the back side of the central panel *Adoration of the Mystic Lamb*. The black rectangle marks an area of interest for inspecting cracks. b) The labelled color version of the marked image part, where green denotes examples of cracks and blue examples of non-crack areas.

TABLE 4. MCN classification result before and after extending the training dataset. Corresponding index: *b*-before and after *a* transfer learning.

Method	Recall	False alar.	False miss.	Precision	F_1-m.
BCTF [8]	**0.9001**	0.0522	**0.0999**	0.7258	0.8036
MCN_b	0.8413	0.0487	0.1587	0.7263	0.7796
MCNC_b	0.8318	0.0436	0.1682	0.7454	0.7862
MCN_a	0.8532	0.0386	0.1468	0.7723	0.8107
MCNC_a	0.8419	**0.0330**	0.1581	**0.7964**	**0.8185**

D. DISCUSSION

The results show clearly that classical classification methods such as SVM, AdaBoost and standard fully connected networks are inferior to deep learning methods in the task of crack detection in paintings. The BCTF method shows comparable performance to deep learning approaches, and can even outperform the CNN-based method [9] and the deep feature fusion method [53] in this task, because it makes efficient use of mutimodal data. However, the parameters of BCTF have been carefully optimized for each new image at hand and its computational complexity makes it less practical for processing large images. The proposed approach yielded better results compared to the CNN-based method that was earlier proposed for crack detection in roads [9] and also better performance compared to the deep learning method [53]. This can mainly be attributed to the fact that our approach efficiently deals with large-scale multimodal data, is more carefully adapted to the problem of crack detection in paintings, and that it includes a novel compensation technique to improve the localization of the exact crack borders.

Compared to BCTF, the proposed approach yields comparable or slightly better results but without the need to be specifically trained for each new image. The results showed clearly that the proposed MCNC trained on other panels and retrained with only relatively few annotations from the new painting performs even better than BCTF that was trained specifically for the particular image. This fact and the ability of the proposed MCNC to process larger images make it especially interesting for practical use. A limitation of the proposed approach is that some of the detected cracks still show up slightly wider than they actually are. Although the proposed compensation approach reduces significantly false thickening of the crack boundaries compared to the traditional CNN-based classification methods, this effect is not entirely suppressed, which can be a point of interest for further research.

VI. CASE STUDY: MYSTIC LAMB PANEL

This section shows a practical example of the application of the proposed method in assisting the conservation carried out by the Belgian Royal Institute of Cultural Heritage (KIK-IRPA). The conservators were interested in investigating how much the vertical wooden reinforcements applied during the 19th century on the reverse side of the panel of the *Adoration of the Mystic Lamb* had affected the formation of cracks on the front side, over time. Especially as the grain of the wood of these blocks (Figure 22) runs vertically, i.e. in the opposite direction of the original panel, such damage may have been expected.

The main difficulty in detecting cracks in the image shown in Figure 23(a) is its high resolution ($6,000 \times 11,500$ pixels), as well as the absence of additional modalities, such as infrared macrophotography and X-Ray images. Moreover, to obtain an acceptable result, it is necessary to mark a large amount of data for training. We used approximately 150,000 tensors of two types (Figure 22(b)) to

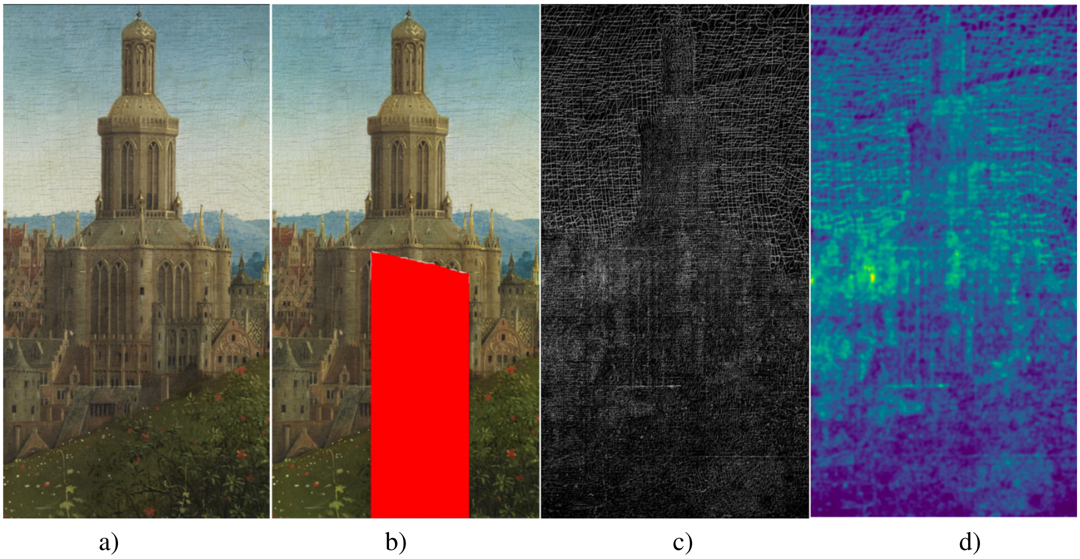


FIGURE 23. Crack detection results. a) Original image, b) Original image with the reinforcement area superimposed, c) Crack map detected by MCNC, d) Blured crack map.

train the network. The training process took approximately 15-20 minutes. The process of classifying the image shown in Figure 22(b)) took around 0.5 hours, using an Nvidia 1050ti graphics card. Manually detecting all the cracks on such a large image would take many hours or days of tedious work.

In Figure 23(c) the result of the crack detection, the location of the reinforcements (Figure 23(b)), and a blurred version of the crack map (Figure 23(d)) are depicted. A standard smoothing filter with a sliding window size of 15 × 15 pixels was used for blurring. The blurred version presents a kind of heat map, indicating the density of the detected cracks. This way we identify areas of higher crack concentrations and visually assess whether they correspond to the areas where the reinforcements are attached to the back of the painting. The analysis of the results indicated a slight increase in crack density in the areas around one of the sharp corners of the reinforcement blocks. However, this could also be coincidental since other areas of increased density did not correspond with the shape of the blocks, but are likely related to the different mechanical behavior of the paint layers according to their composition, or to accidental damages. Therefore, it could not be concluded on the base of these measurements that the reinforcements caused local tensions within the paint layers. No additional consolidation treatments were needed in these areas. However, to prevent possible tensions along the joints of the support, the blocks were replaced with fastenings with the grain running parallel to the direction of the wood in the panel.[6]

VII. CONCLUSIONS

In this paper we explored the potential of deep learning for crack detection in paintings and developed an efficient crack detection method based on convolutional neural networks.

The proposed method is designed to efficiently process high-resolution multimodal images with an arbitrary number of image channels. Our two-step procedure with morphological preprocessing efficiently and safely eliminates the areas where it makes little sense to run the learning process, hereby greatly reducing the total computation burden. Owning to the inherent continuous-learning property of CNNs, our MCN/MCNC method improves its performance when new annotations become available, without the need to fully re-train the network. The results demonstrate clearly the efficiency of our re-training approach where the network trained on one painting successfully detects cracks in another painting with relatively few extra labels added.

Another important contribution of this paper is an efficient approach to alleviating an inherent limitation of CNN-based methods, which is manifested by false thickening of the detected line-like structures such as cracks. The proposed compensation method significantly improves the localization of actual crack boundaries. The proposed solution is applicable in general to CNN-based detection of cracks and other line-like or tubular structures in images.

A thorough evaluation on multimodal acquisitions of the *Ghent Altarpiece* demonstrates the benefit of crack detection in diagnosing the state of panels and the importance of this supporting tool for art conservation practice.

ACKNOWLEDGMENT

The authors would like to thank Voronin Viacheslav thanks Ghent University under the Russia Platform for the visiting fellowship.

[6] This work was carried-out by Jean-Albert Glatigny during the conservation project by the Royal Institute for Cultural Heritage (KIK-IRPA).

REFERENCES

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

[2] B. Cornelis, T. Ružić, E. Gezels, A. Dooms, A. Pižurica, L. Platiša, J. Cornelis, M. Martens, M. De Mey, and I. Daubechies, "Crack detection and inpainting for virtual restoration of paintings: The case of the ghent altarpiece," *Signal Process.*, vol. 93, no. 3, pp. 605–619, Mar. 2013.

[3] A. Pizurica, L. Platisa, T. Ruzic, B. Cornelis, A. Dooms, M. Martens, H. Dubois, B. Devolder, M. De Mey, and I. Daubechies, "Digital image processing of the ghent altarpiece: Supporting the painting's study and conservation treatment," *IEEE Signal Process. Mag.*, vol. 32, no. 4, pp. 112–122, Jul. 2015.

[4] A. Gupta, V. Khandelwal, A. Gupta, and M. C. S. Thammasat, "Image processing methods for the restoration of digitized paintings," *Int. J. Sci. Technol.*, vol. 13, no. 3, pp. 66–72, 2008.

[5] I. Giakoumis, N. Nikolaidis, and I. Pitas, "Digital image processing techniques for the detection and removal of cracks in digitized paintings," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 178–188, Jan. 2006.

[6] G. S. Spagnolo and F. Somma, "Virtual restoration of cracks in digitized image of paintings," *J. Phys., Conf. Ser.*, vol. 249, Nov. 2010, Art. no. 012059.

[7] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[8] B. Cornelis, Y. Yang, J. T. Vogelstein, A. Dooms, I. Daubechies, and D. Dunson, "Bayesian crack detection in ultra high resolution multimodal images of paintings," in *Proc. 18th Int. Conf. Digit. Signal Process. (DSP)*, Jul. 2013, pp. 1–8.

[9] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3708–3712.

[10] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 5, pp. 361–378, May 2017.

[11] V. Voronin, V. Marchuk, R. Sizyakin, N. Gapon, M. Pismenskova, and S. Tokareva, "Automatic image cracks detection and removal on mobile devices," *Proc. SPIE*, vol. 9869, May 2016, Art. no. 98690R.

[12] L. Zhang, Q. Zhang, L. Zhang, D. Tao, X. Huang, and B. Du, "Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding," *Pattern Recognit.*, vol. 48, no. 10, pp. 3102–3112, Oct. 2015.

[13] F. Luo, L. Zhang, B. Du, and L. Zhang, "Dimensionality reduction with enhanced hybrid-graph discriminant learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, early access, Jan. 27, 2020, doi: 10.1109/TGRS.2020.2963848.

[14] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2406–2419, Jul. 2019.

[15] Y. Yang and D. B. Dunson, "Bayesian conditional tensor factorizations for high-dimensional classification," *J. Amer. Stat. Assoc.*, vol. 111, no. 514, pp. 656–669, Apr. 2016.

[16] Y. Li, H. Li, and H. Wang, "Pixel-wise crack detection using deep local pattern predictor for robot application," *Sensors*, vol. 18, no. 9, p. 3042, 2018.

[17] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li, "DeepCrack: A deep hierarchical feature learning architecture for crack segmentation," *Neurocomputing*, vol. 338, pp. 139–153, Apr. 2019.

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

[19] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[23] S. Mallat, "Understanding deep convolutional networks," *Philos. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 374, Apr. 2016, Art. no. 20150203.

[24] J.-H. Jacobsen, E. Oyallon, S. Mallat, and A. W. M. Smeulders, "Multiscale hierarchical convolutional networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1–10.

[25] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[26] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," 2019, *arXiv:1901.06032*. [Online]. Available: http://arxiv.org/abs/1901.06032

[27] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[28] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust RGB-D object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 681–687.

[29] A. Wang, J. Lu, J. Cai, T.-J. Cham, and G. Wang, "Large-margin multimodal deep learning for RGB-D object recognition," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1887–1898, Nov. 2015.

[30] S. Gupta, R. B. Girshick, P. Arbelaez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, Zurich, Switzerland, Sep. 2014, pp. 345–360.

[31] K. Zhou, A. Paiement, and M. Mirmehdi, "Detecting humans in RGB-D data with CNNs," in *Proc. 15th IAPR Int. Conf. Mach. Vis. Appl. (MVA)*, May 2017, pp. 280–283.

[32] N. Takahashi, M. Gygli, and L. Van Gool, "AENet: Learning deep audio features for video analysis," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 513–524, Mar. 2018.

[33] A. Torfi, S. M. Iranmanesh, N. Nasrabadi, and J. Dawson, "3D convolutional neural networks for cross audio-visual matching recognition," *IEEE Access*, vol. 5, pp. 22081–22091, 2017.

[34] C. Couprie, C. Farabet, L. Najman, and Y. LeCun, "Indoor semantic segmentation using depth information," in *Proc. 1st Int. Conf. Learn. Representations, ICLR*, 2013, pp. 1–8.

[35] L. Windrim, R. Ramakrishnan, A. Melkumyan, and R. Murphy, "Hyperspectral CNN classification with limited training samples," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2017, pp. 4.1–4.12.

[36] Y. Luo, J. Zou, C. Yao, X. Zhao, T. Li, and G. Bai, "HSI-CNN: A novel convolution neural network for hyperspectral image," in *Proc. Int. Conf. Audio, Lang. Image Process. (ICALIP)*, Jul. 2018, pp. 464–469.

[37] X. Li, M. Ding, and A. Pizurica, "Group convolutional neural networks for hyperspectral image classification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 639–643.

[38] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *Proc. Int. Conf. Learn. Representations ICLR*, 2016, pp. 1–14.

[39] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist., PMLR*, vol. 15, Jun. 2011, pp. 315–323.

[40] A. L. Maas, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2013, p. 3.

[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations ICLR*, 2015, pp. 1–13.

[43] B. Kim and S. Cho, "Automated vision-based detection of cracks on concrete surfaces using a deep learning technique," *Sensors*, vol. 18, no. 10, p. 3452, 2018.

[44] P. Buyssens, A. Elmoataz, and O. Lezoray, "Multiscale convolutional neural networks for classification vision based of cells," *Computer Vision—ACCV*. Berlin, Germany: Springer, 2013, pp. 342–352.

[45] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 354–370.

[46] W. Lotter, G. Sorensen, and D. Cox, "A multi-scale CNN and curriculum learning strategy for mammogram classification," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support* (Lecture Notes in Computer Science). Cham, Switzerland: Springer, 2017, pp. 169–177.

[47] R. Sizyakin, V. Voronin, N. Gapon, A. Zelensky, and A. Pižurica, "Automatic detection of welding defects using the convolutional neural network," *Proc. SPIE*, vol. 11061, Jun. 2019, Art. no. 110610E.

[48] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in *Proc. Workshop Unsupervised Transf. Learn. ICML*, 2012, pp. 17–37.

[49] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. 31st Int. Conf. Mach. Learn., PMLR*, Jan. 2014, pp. 647–655.

[50] D. Rumelhart, G. Hinton, and R. Williams, "Learning internal representations by error propagation," *Parallel Data Process.*, vol. 1, pp. 318–362, Sep. 1986.

[51] Y. Freund, R. E. Schapire, and N. Abe, "A short introduction to boosting," *J. Japn. Soc. Artif. Intell.*, vol. 14, no. 5, pp. 771–780, 1999.

[52] B. Scholkopf and A. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization and Beyond, Adaptive Computation and Machine Learning*. Cambridge, MA, USA: MIT Press, 2002.

[53] W. Song, S. Li, L. Fang, and T. Lu, "Hyperspectral image classification with deep feature fusion network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3173–3184, Jun. 2018.

[54] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[55] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

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.

BRUNO CORNELIS (Member, IEEE) received the master's degree of Industrial Engineer in electronics from Erasmus Hogeschool Brussel (EHB), Belgium, in 2005, the master's degree in electrical engineering and the Ph.D. degree (Hons.) in applied sciences from Vrije Universiteit Brussel (VUB), Belgium, in 2007 and 2014, respectively. During his Ph.D. degree with the Department of Electronics and Informatics, he investigated the use of various image processing tools in support of art scholarship. His research interests include statistical data analysis and sparse representations in computational imaging applications. He is currently a data scientist at the company Macq, Belgium. Since 2016, he has been a Guest Professorship with the Department of Electronics and Informatics, VUB.

LAURENS MEEUS (Member, IEEE) received the Bachelor of Science degree in engineering physics and the Master of Science degree in photonics engineering from Ghent University, in 2014 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Group for Artificial Intelligence and Sparse Modeling, Department of Telecommunication and Interpretation, Ghent University. His research interests include multimodal image processing and deep learning for assisting art investigation.

HÉLÈNE DUBOIS is currently a Painting Conservator and Art Historian. She trained at the Université Libre de Bruxelles and the Hamilton Kerr Institute, University of Cambridge (UK). She worked at the Doerner-Institut, Munich, the J. Paul Getty Museum, Malibu, the Royal Museums of Fine Arts in Brussel, and the Limburg Conservation Institute, Maastricht, where she taught conservation of Old Masters Paintings. Attached to the Royal Institute for Cultural Heritage, Brussel (KIK-IRPA) and to Ghent University (UGent), she leads the conservation project of the brothers van Eyck's Adoration of the Mystic Lamb, since October 2016. She is researching the material history of the altarpiece for a Ph.D. dissertation at Ghent University.

MAXIMILIAAN MARTENS received the Ph.D. degree in art history from the University of California at Santa Barbara, in 1992. He is currently a Full Professor of art history with Ghent University and a member of the Royal Flemish Academy of Belgium for Science and Arts (KVAB). As a specialist in Flemish art of the fifteenth and early sixteenth century, he has authored several books and articles on Jan van Eyck, Hans Memling, Pieter Bruegel the Elder and others and supervised numerous Ph.D.-dissertations. His main research interests include artistic production in an urban environment, technical art history and the history and theory of art conservation.

VIACHESLAV VORONIN received the B.S. and M.S. degree in radio engineering from the South-Russian State University of Economics and Service, in 2006 and 2008, respectively, and the Ph.D. degree in technics from Southern Federal University in 2009. He is the Head of the Center for Cognitive Technology and Machine Vision Moscow State University of Technology STANKIN, Moscow, Russia. He was born in Rostov, Russia, in 1985. He is a member of Program Committee of conference SPIE. His research interests include image processing, painting, and computer vision.

ALEKSANDRA PIŽURICA (Senior Member, IEEE) 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.