RESEARCH ARTICLE OPEN ACCESS

Identification and Estimation of Cracks on Digitized Paintings Using Weiner filter and our method

Subramanian Karuppiah[1]* S.K. Srivatsa[2]

[1] *Jawaharal Nehru Technological University, Hyderabad, India*
[2] *Anna University, Chennai, India*

Retired Professor, Anna University, Chennai, TamilNadu, India.

Corresponding Author Asst.Professor, St.Joseph's College of Engineering, Chennai-119

ABSTRACT

An new technique for the detection and elimination of cracks in digitized paintings and images is proposed in this paper. Compare the paper with wiener filter and our method. Cracks degrade the quality of painting as well as authenticity of painting becomes questionable. In the proposed method the cracks are detected by thresholding the result of the morphological top-hat transform. Further, misidentified cracks are detected either by involving user intervention or by using a semi-automatic procedure based on region growing technique. Finally, crack interpolation also called crack filling is performed using order statistics filters so as to restore the cracked image. The true positive rate and false positive rate are used to evaluate the performance of the proposed technique. For experimental purpose we collected 2000 classified cracked and un-cracked digital paintings and images from online digital art database. The result shows achievement of true positive rate of about 98.3% at the rate of 0.1 false positive per image. This is because of providing user intervention during module called identifying miss-identified cracks.

Keywords- Digital image processing, digitized paintings, crack detection, crack filling, and Wiener filter.

INTRODUCTION

Image processing techniques have recently been applied to analysis, preservation and restoration of artwork. Ancient paintings are cultural heritage for one’s country which can be preserved by computer aided analysis and processing. These paintings get deteriorated mainly by an undesired pattern that causes breaks in the paint, or varnish. Such a pattern can be rectangular, circular, spider-web, unidirectional, tree branches and random [3] and are usually called cracks. Cracks are caused mainly by aging, drying and mechanical factors like vibration, and human handling. Computer aided tools can be used to implement image processing technique for the elimination and detection of cracks in ancient digital paintings. Though this methodology produces a digitally restored version of the artwork, it can still prove to be a useful guide for historians, and museum curators. A technique that is able to track and fill a crack is proposed in [4] but it requires the user to manually start with the initial point of the crack pattern to fill them. A similar problem of detection and filling of cracks has been treated by Giakoumis and Pitas [1]. Their process first detects the crack using a morphological top-hat operator and then fills them using a trimmed median filter [6], and an anisotropic diffusion filter.

Crack detection and removal bears certain similarities with methods proposed for the detection and removal of scratches and other artifacts from motion picture films [3]. However, such methods rely on information obtained over several adjacent frames for both artifact detection and filling and, thus, are not directly applicable in the case of painting cracks.

Where information has to be filled in are known. Different approaches for interpolating information in structured [6]–[10] and textured image areas [11] have been developed. The former are usually based on partial differential equations (PDEs) and on the calculus of variations whereas the latter rely on texture synthesis principles. A technique that decomposes the image to textured and structured areas and uses appropriate interpolation techniques depending on the area where the missing information lies has also been proposed [12]. The results obtained by these techniques are very good. A methodology for the restoration of cracks on digitized paintings, which adapts and integrates a number of image processing and analysis tools, is proposed in this paper. The methodology is an extension of the crack removal framework presented in [13]. The technique consists of the following stages:

- Crack detection
- Separation of the thin dark brush strokes, which have been misidentified as cracks;
- Crack filling (interpolation).

A certain degree of user interaction, most notably in the crack-detection stage, is required for optimal results. User interaction is rather unavoidable since the large variations observed in the typology of cracks would lead any fully automatic algorithm to failure. However, all processing steps can be executed in real time, and, thus, the user can instantly observe the effect of parameter tuning on the image under study and select in an intuitive way the values that achieve the optimal visual result. Needless to say, only subjective optimality criteria can be used in this case since no ground truth data are available. The opinion of restoration experts that inspected the virtually restored images was very positive. This paper is organized as follows. Section II describes the crack-detection procedure. Two methods for the separation of the brush strokes which have been falsely identified as cracks are presented in Section III. Methods for filling the cracks with image content from neighboring pixels are proposed in Section IV

RELATED WORKS

A Crack-detector system for detecting and removal of cracks in digitized paintings was presented in [1]. It can track and interpolate cracks. But this methodology is not widely adopted since the user should manually select a point on each crack to be restored.

A method for the detection of cracks using multi oriented Gabor filters is presented in [2]. Advantage of Gabor filter: It is used at different scales and spatial frequencies, edge tracking control of in detail. Disadvantage of Gabor filter: computational time is more.

Crack detection and removal bears certain similarities with methods proposed for the detection and removal of scratches and other artifacts from motion picture films [3]. However, such methods rely on information obtained over several adjacent frames for both artifact detection and filling and, thus, are not directly applicable in the case of painting cracks.

Further research areas [4], [5] that are closely connected to crack removal include image in painting which deals with the reconstruction of missing or damaged image areas by filling in information from the neighboring areas, and disoclusion (filling in action), i.e., recovery of object parts that are hidden behind other objects within an image. Methods developed in these areas assume that the regions where information has to be filled in are known.

Various approaches for interpolating information in structured image have been developed [6]–[10]. These methods are usually based on partial differential equations (PDEs) and on the calculus of variations.

Different approaches for interpolating information textured image areas [11] rely on texture synthesis principles. An algorithm for texture transfer between images that is up to several orders of magnitude faster than current state-of-the-art techniques the technique can leverage self-similarity of complex images to increase resolution of some types of images and to create novel, artistic looking images from photographs without any prior artistic source. Compared to other alternatives, methods based on texture transfer are global in the sense that the user need not deal with details such as defining and painting individual brush strokes. Texture transfer methods are also more general since they don't need to emulate any particular artistic style (line drawing, hatching, realistic oil painting, and so on). Not surprisingly, there is a price to pay for this generality - an algorithm designed for a specific artistic style will most likely produce superior results.

Mostly digitized paintings consist of both structured and textured images. Hence, A technique that decomposes the image to textured and structured areas and uses appropriate interpolation techniques depending on the area where the missing information lies has also been proposed [12]. The results obtained by these techniques are very good.

A.Vijayaraj & R.Saravanan[15] proposed , the path of the network is based on the routing table it is not fixed in Ad-hoc technique network path is also not fixed and create dynamically. It design is practical oriented But it is more cost. The information is sent by the source and there is repeatedly stored in the rear.

A method for the renovation of cracks on digitized paintings, which adapts and integrates a number of image processing and analysis tools are used in this paper. The methodology is an extension of the crack removal framework presented in [13]. The technique consists of the following stages:

1. Crack detection
2. Separation of the thin dark brush strokes, which have been misidentified as cracks
3. Crack filling (interpolation).

A definite degree of user interaction, most particularly in the crack-detection stage, is required for optimal results. User contact is rather inevitable since the large variations observed in the typology of cracks would lead any fully automatic algorithm to failure. However, all processing steps can be executed in real time, and, thus, the user can instantly observe the effect of parameter tuning on the image under study and select in an intuitive way the values that achieve the optimal visual result. Needless to say, only subjective optimality criteria can be used in this case since no ground truth data are available. The opinion of restoration experts that inspected the virtually restored images was very positive.

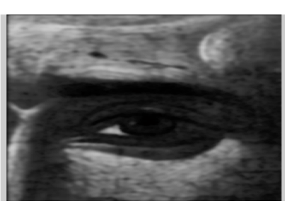
This paper proposes an integrated strategy for crack detection and filling in digitized paintings. Cracks are detected by using top-hat transform, whereas the thin dark brush strokes, which are misidentified as cracks, are separated either by an automatic technique (MRBF networks) or by a semi-automatic approach. Appropriately modified order statistics filters or controlled anisotropic diffusion performs crack interpolation. The goal of the paper is to restore the old cracked paintings image by finding the cracks in the gray scale image and filling cracks with neighbor pixel values using median filter.

IMPLEMENTATION

In first step, “CRACK DETECTION PROCEDURE” the user can control the result of the crack-detection procedure by choosing appropriate values for the following parameters:

The type of the structuring element;

The size of the structuring element and the number of dilations in (2).



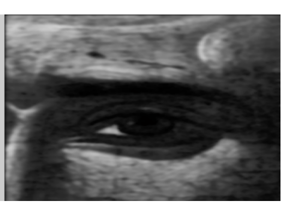


Figure1.Original image Figure2.Final image

Bottom-hat filtering

Syntax

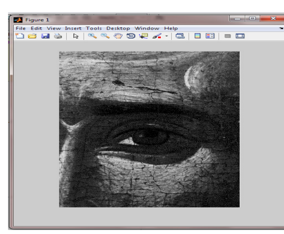
IM2 = imbothat (IM,SE)

IM2 = imbothat(IM,NHOOD)

Description

IM2 = imbothat (IM, SE) performs morphological bottom-hat filtering on the grayscale or binary input image, IM, returning the filtered image, IM2. The argumentSE is a structuring element returned by the strel function. SE must be a single structuring element object, not an array containing multiple structuring element objects.

IM2 = imbothat (IM, NHOOD) performs morphological bottom-hat filtering where NHOOD is an array of 0's and 1's that specifies the size and shape of the structuring element. This is equivalent to imbothat (IM, strel (NHOOD)).



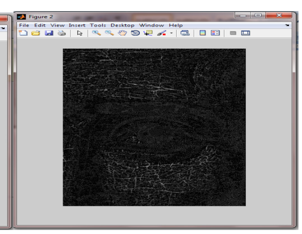


Figure3.Original image Figure4

Figure4.Image after bottom hat Transform

(Structuring element (Square (7X7))

These parameters affect the size of the “final” structuring element and must be chosen according to the thickness of the cracks to be detected. It should be noted, however, that these parameters are not very critical for the algorithm performance due to the thresholding operation that will be described in the next paragraph and also due to the existence of the brush-stroke/crack separation procedure, which is able to remove crack-like brush strokes that have been erroneously identified as cracks. The fact that all the results presented in this paper have been obtained with the same top-hat transform parameters comes as a clear indication that the above statement is indeed true. These parameters were the following:

- Structuring element type: square;
- Structuring element size: 7×7;
- Number of dilations in (2): 2.

The top-hat transform generates a grayscale output image where pixels with a large grey value are potential crack or crack-like elements. Therefore, a thresholding operation on is required to separate cracks from the rest of the image. Advantages: The top-hat transformation method will produce accurate crack pixels with threshold adjustments.

In Second step 'SEPARATION OF THE BRUSH STROKES FROM THE CRACKS' In some paintings, certain areas exist where brush strokes have almost the same thickness and luminance features as cracks. The hair of a person in a portrait could be such an area. Therefore, the top-hat transform might misclassify these dark brush strokes as cracks. Thus, in order to avoid any undesirable alterations to the original image, it is very important to separate these brush strokes from the actual cracks, before the implementation of the crack filling procedure. For this we use Semi-Automatic Crack Separation. A simple interactive approach for the separation of cracks from brush strokes is to apply a region growing algorithm on the threshold output of the top-hat transform, starting from pixels (seeds) on the actual cracks. The pixels are chosen by the user in an interactive mode. At least one seed per connected crack element should be chosen. Alternatively, the user can choose to apply the technique on the brush strokes, if this is more convenient. The growth mechanism that was used implements the well-known grassfire algorithm that checks recursively for unclassified pixels with value 1 in the 8-neighborhood of each crack pixel. At the end of this procedure, the pixels in the binary image, which correspond to brush, strokes that are not 8-connected to cracks will be removed. The above procedure can be used either in a stand-alone mode or applied on the output of the MRBF separation procedure described in the next section to eliminate any remaining brush strokes. Advantages: The grassfire algorithm efficiently identifies the misidentified crack pixel i.e. thin hair-like brush stroke information's.

In third step, 'CRACK-FILLING METHODS' After identifying cracks and separating misclassified brush strokes, the final task is to restore the image using local image information (i.e., information from neighboring pixels) to fill (interpolate) the cracks. Two classes of techniques, utilizing order statistics filtering and anisotropic diffusion are proposed for this purpose. Both are implemented on each RGB channel independently and affect only those pixels which belong to cracks. Therefore, provided that the identified crack pixels are indeed crack pixels, the filling procedure does not affect the "useful" content of the image. Image in painting techniques like the ones cited in Section I can also be used for crack filling. The performance of the crack filling methods presented below was judged by visual inspection of the results. Obviously, measuring the performance of these methods in an objective way is infeasible since ground truth data (e.g., images depicting the paintings in perfect condition, i.e., without cracks) are not available.

Crack Filling Based on Weiner Filter and our conventional Filter

An effective way to interpolate the cracks is to apply median or other order statistics filters [15] in their neighborhood. All filters are selectively applied on the cracks, i.e., the center of the filter window traverses only the crack pixels. If the filter window is sufficiently large, the crack pixels within the window will be outliers and will be rejected. Thus, the crack pixel will be assigned the value of one of the neighboring non-crack pixels.

The following filters can be used for this purpose.•

The Wiener filtering is optimal in terms of the mean square error. In other words, it minimizes the overall mean square error in the process of inverse filtering and noise smoothing. The Wiener filtering is a linear estimation of the original image. The approach is based on a stochastic framework. The orthogonality principle implies that the Wiener filter in Fourier domain can be expressed as follows:

$$W(f_1, f_2) = \frac{H^*(f_1, f_2) S_{xx}(f_1, f_2)}{|H(f_1, f_2)|^2 S_{xx}(f_1, f_2) + S_{\eta\eta}(f_1, f_2)},$$

Where $S_{xx}(f_1, f_2)$, $S_{\eta\eta}(f_1, f_2)$ are respectively power spectra of the original image and the additive noise, and $H(f_1, f_2)$ is the blurring filter. It is easy to see that the Wiener filter has two separate parts, an inverse filtering part and a noise smoothing part. It not only performs the deconvolution by inverse filtering (highpass filtering) but also removes the noise with a compression operation (lowpass filtering).

Our conventional method as follows

Step1: Read the painting image from database.

Step2: Convert the image into grayscale.

Step3: Define the structuring element with arguments.

Step4: Apply the bottom up transform with structuring element to identify the cracks.

Step5: image can be resized.

Step6: Apply my method with pass the arguments with structuring element.

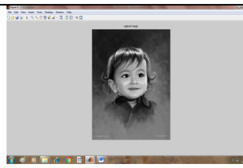
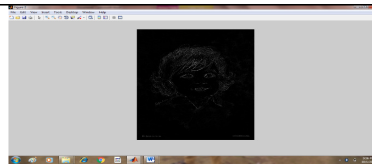
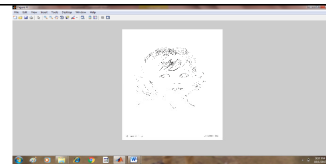
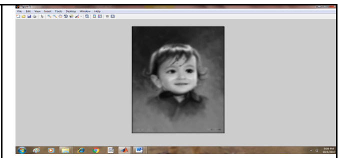
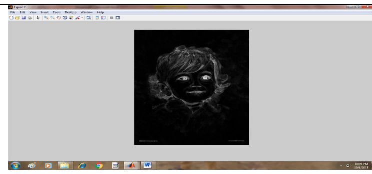
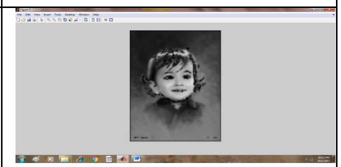
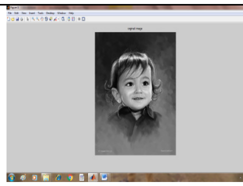
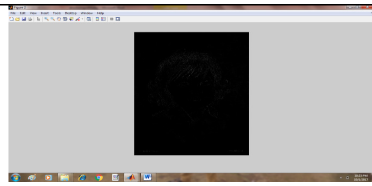
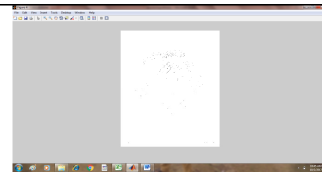
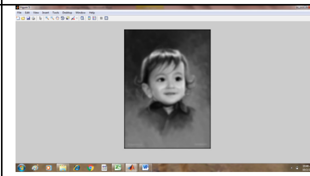
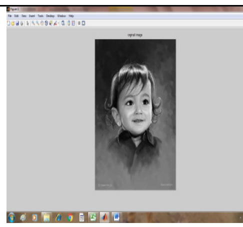
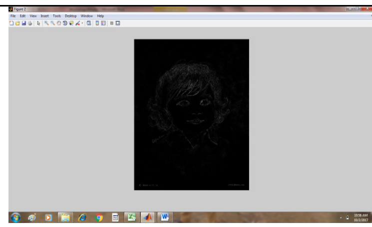
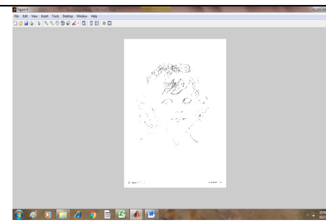
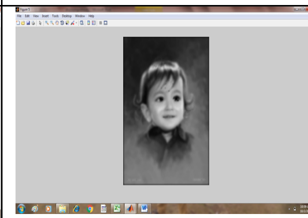
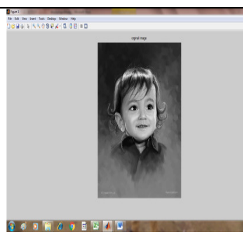
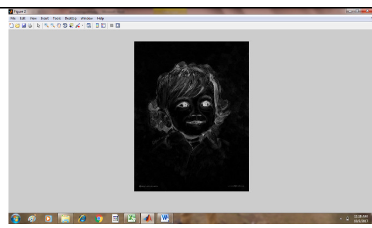
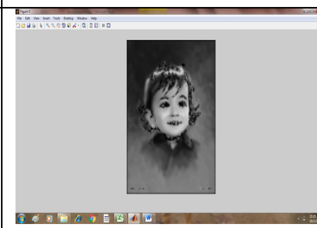
Step7: calculate the Psnr value between all the images.

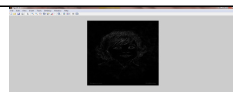
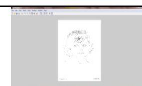
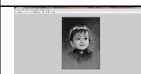
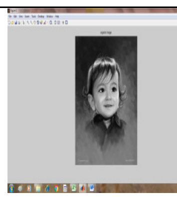
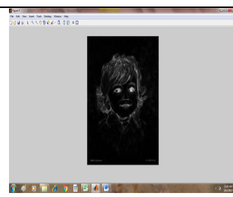
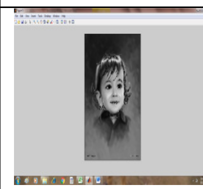
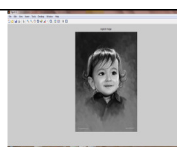
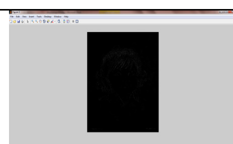
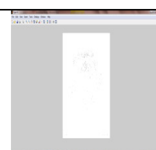
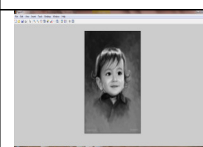
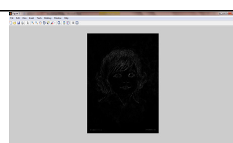
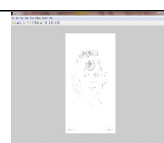
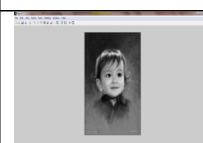
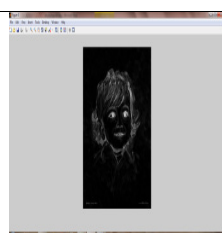
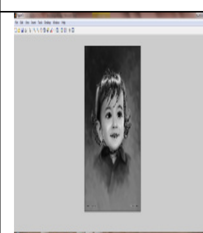
Step 8: calculate the tot and avg.

Step 9: calculate the mean and standard deviation for compare the quality of the rectified image.

The results is as follows

SE	Original	After bottom-hop transform	Cracked image	After applying Wiener filtering

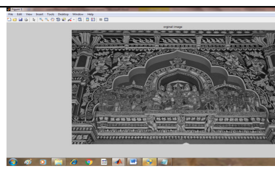
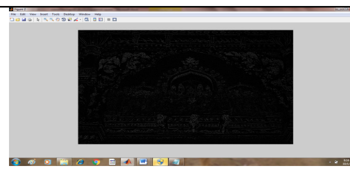
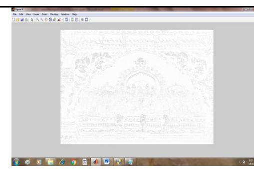
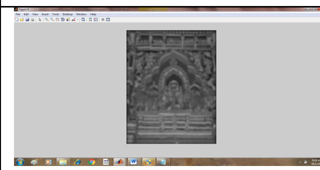
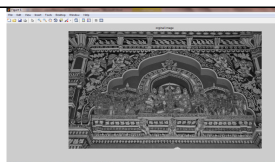
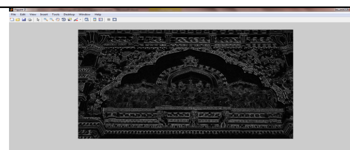
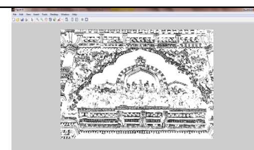
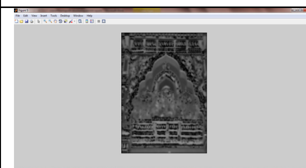
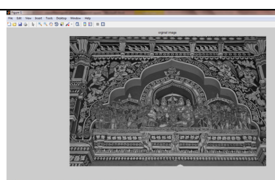
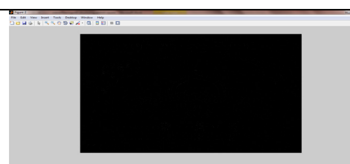
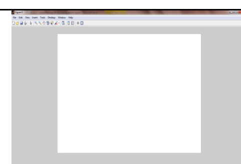
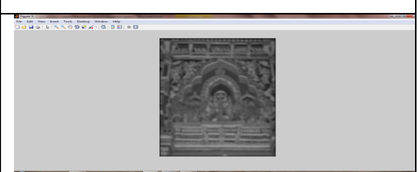
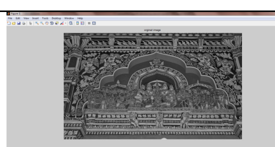
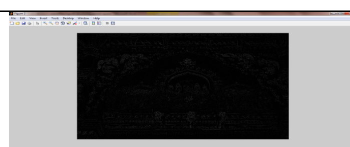
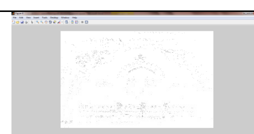
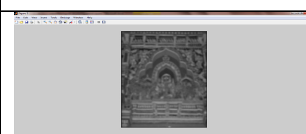
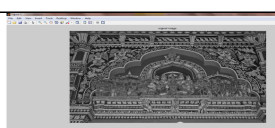
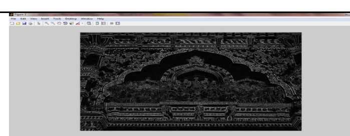
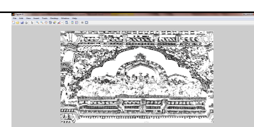
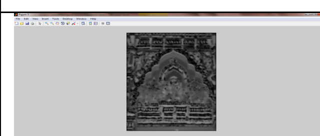
Square(6)				
Disk(16)				
Line(3,3)				
Rectangle(5,5)				
Diamond(20)				

SE	Original	After bottom-hop transform	Cracked image	After applying My method
Square(6)				
Disk(15)				
Line(3,3)				
Rectangle(5,5)				
Diamond(20)				

SE	Img1	Img2	Img3	Img4	Tot	Avg	Sd	Sd/M
Square(6)	17.7575	10.7138	0.2655	3.4417	31.7825	6.3565	12.7130	2
Disk(16)	17.7575	10.7866	0.7264	3.5000	32.7706	6.5541	13.1082	2
Line(3,3)	17.7575	10.0636	0.0795	3.3539	31.2545	6.2509	12.5058	2
Rectangle(5,5)	17.7575	10.2768	0.2326	3.4326	31.6995	6.3399	12.6798	2
Diamond(20)	17.7575	10.7980	0.7552	3.5163	32.6770	6.5654	13.1308	2

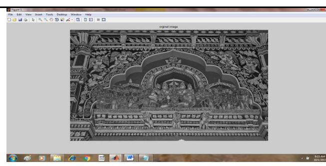
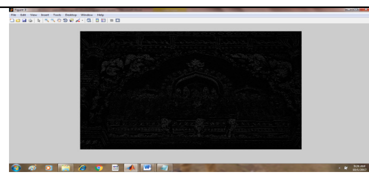
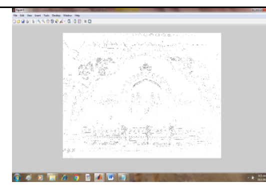
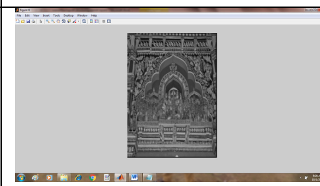
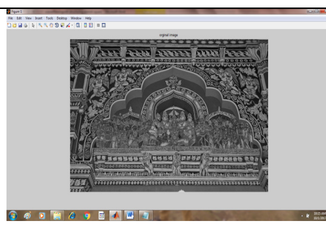
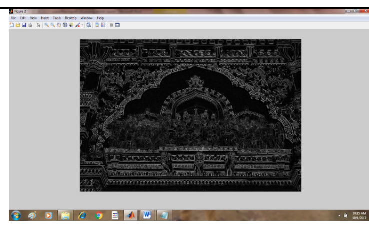
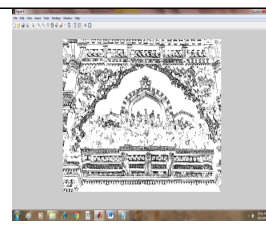
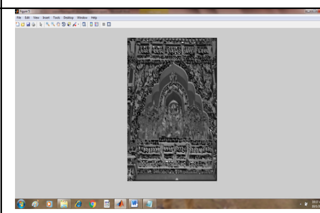
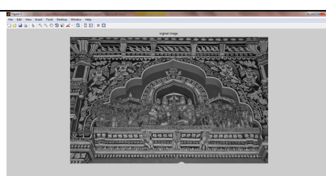
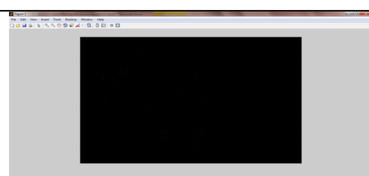
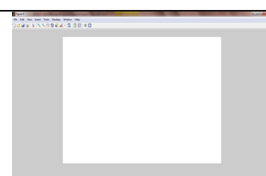
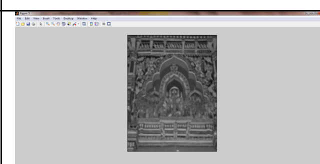
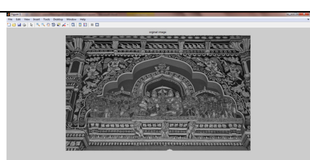
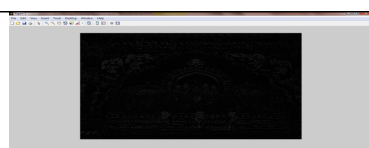
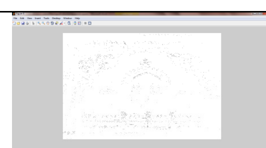
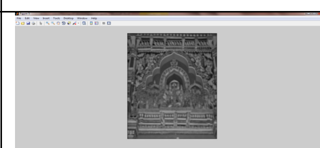
My method

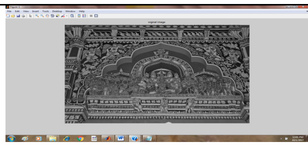
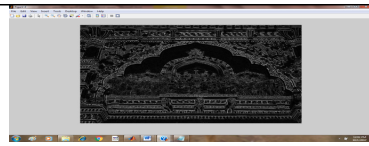
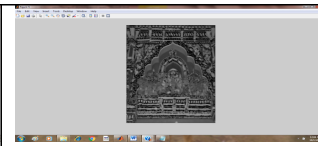
Square(6)	17.7575	10.3178	0.2655	3.4630	31.8038	6.3608	12.7215	2
Disk(15)	17.7575	10.7574	0.6921	3.5831	32.7901	5.5580	12.1160	2
Line(3,3)	17.7575	10.0636	0.0795	3.3723	31.2729	6.2546	12.5092	2
Rectangle(5,5)	17.7575	10.2768	0.2236	3.4521	31.7190	6.3438	12.6768	2
Diamond(20)	17.7575	10.7980	0.7552	3.6086	32.9193	6.5839	13.1677	2

SE	Original	After bottom-hop transform	Cracked image	After applying Wiener filtering
Square(6)				
Disk(16)				
Line(3,3)				
Rectangle (5,5)				
Diamond(20)				

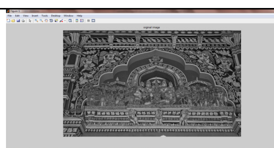
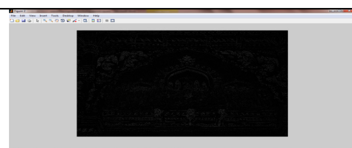
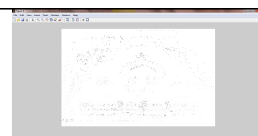
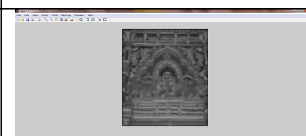
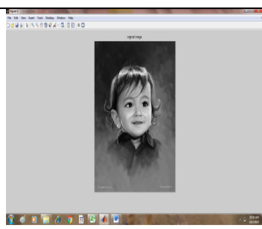
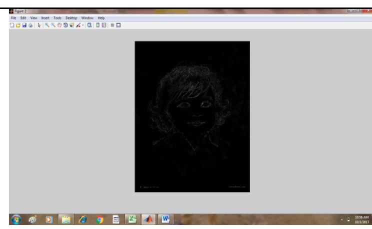
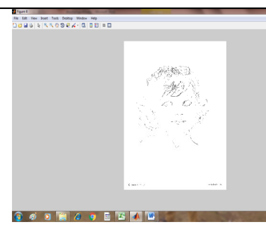
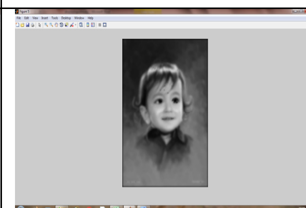
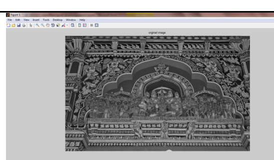
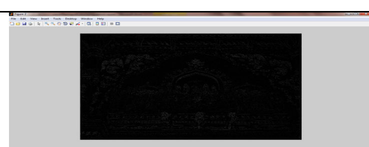
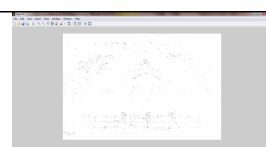
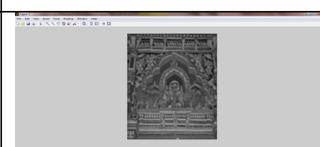
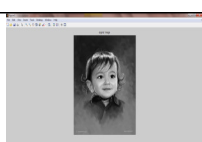
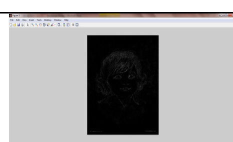
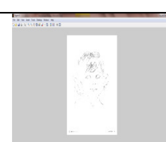
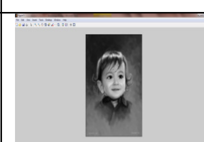
SE	Img1	Img2	Img3	Img4	Tot	Avg	Sd	Sd/ M

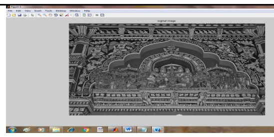
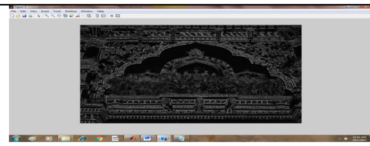
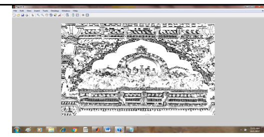
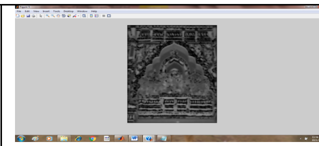
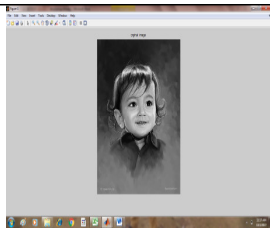
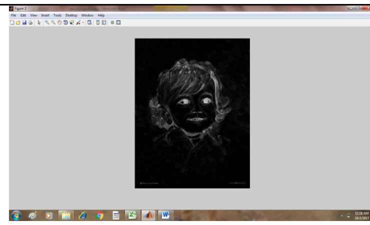
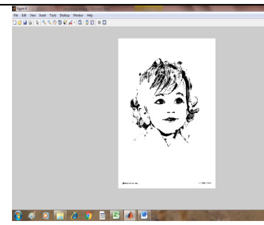
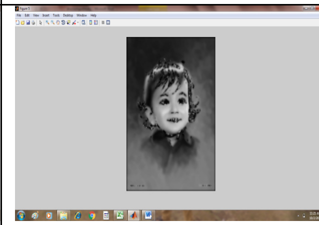
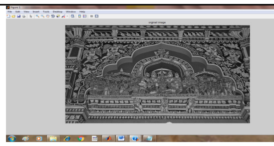
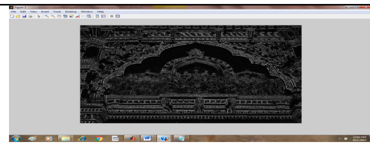
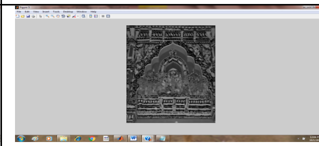
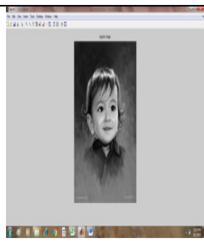
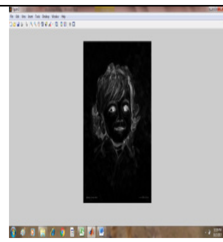
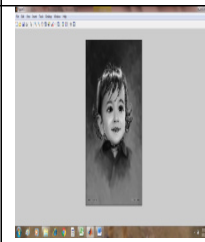
Square(6)	25.1476	9.3516	0.4520	3.9018	38.8529	7.7706	15.5412	2
Disk(16)	25.4176	10.8912	2.2142	4.0506	42.3731	8.8474	16.9848	2
Line(3,3)	25.4176	8.9013	0.0506	3.6048	37.4073	7.5049	15.0817	2
Rectangle(5 ,5)	25.4176	9.2215	0.3272	3.8089	32.5052	7.7010	15.4021	2
Diamond(2 0)	25.4176	11.0105	2.5278	4.0320	42.7179	8.5436	17.0872	2

SE	Original	After bottom-hop transform	Cracked image	After applying Wiener filtering
Square(6)				
Disk(15)				
Line(3,3)				
Rectangle(5,5)				

Diamond(20)				

SE	Img1	Img2	Img3	Img4	Tot	Avg	Sd	Sd/M
Square(6)	25.4176	9.3516	0.4520	3.8710	38.222	7.7644	15.5289	2
Disk(15)	25.4176	10.8912	2.2142	4.3377	42.5188	8.5038	17.0055	2
Line(3,3)	25.4176	8.0913	0.0506	3.6102	37.7096	7.5419	15.0838	2
Rectangle(5,5)	25.4176	9.2215	0.3272	3.8138	38.5100	7.7020	15.4040	2
Diamond(20)	25.4176	11.0105	2.5278	4.3725	43.0584	8.6117	17.2233	2

SE	Original	After bottom-hop transform	Cracked image	After applying Wiener filtering
Rectangle(5,5) Weiner Filter				
				
My method				
				

Diamond(20) Weiner Filter				
				
My method				
				

SE	Img1	Img2	Img3	Img4	Tot	Avg	Sd	Sd/ M
Rectangle(5,5) Weiner Filter	25.4176	9.2215	0.3272	3.8089	32.5052	7.6010	15.4021	2
My Method	**25.4176**	**9.2215**	**0.3272**	**3.8138**	**38.5100**	**7.7020**	**15.4040**	**2**
Diamond(20) Weiner Fliter	17.7575	10.7980	0.7552	3.6086	32.9193	6.5839	13.1677	2
My Method	**25.4176**	**11.0105**	**2.5278**	**4.3725**	**43.0584**	**8.6117**	**17.2233**	**2**

Conclusion

In this paper, we have presented a strategy for crack detection and filling in digitized paintings and medical images. Cracks are detected by using top-hat transform, whereas the thin dark brush strokes, which are misidentified as cracks, are separated either by an automatic technique (MRBF networks) or by a semi-automatic approach. Crack interpolation is performed by appropriately modified order statistics filters or controlled anisotropic diffusion. The methodology has been applied for the virtual restoration of images and was found very effective by restoration experts. However, there are certain aspects of the proposed methodology that can be further improved. For example, the crack-detection stage is not very efficient in detecting cracks located on very dark image areas, since in these areas the intensity of crack pixels is very close to the intensity of the surrounding region. A possible solution to this shortcoming would be to apply the crack-detection algorithm locally on this area and select a low threshold value. Another situation where the system (more particularly, the crack filling stage) does not perform as efficiently as expected is in the case of cracks that cross the border between regions of different color. In such situations, it might be the case that part of the crack in one area is filled with color from the other area, resulting in small spurs of color in the border between the two regions. Such a situation is depicted in Figure. 5. However, this phenomenon is rather seldom and, furthermore, the extent of these erroneously filled areas is very small (2–3 pixels maximum). A possible solution would be to perform edge detection or segmentation on the image and confine the filling of cracks that cross edges or region borders to pixels from the corresponding region. Use of image in painting techniques [6]–[10] could also improve results in that aspect. Another improvement of the crack filling stage could aim at using properly adapted versions of nonlinear multi-channel filters e.g., variants of the vector median filter) instead of processing each color channel independently. These improvements will be the topic of future work on this subject.

REFERENCES

[1] Giakoumis, N. Nikolaidis, and Pitas, A, "Digital image Processing Techniques for the Detection and Removal of Cracks in Digitized Paintings", IEEE Transactions on Image Processing, 15, 178-188, (2006).

[2] Barni, M. Bartolini, F. and Cappellini, V. "Image processing for virtual restoration of artworks," IEEE Multimedia, 7, 34–37, (2000).

[3] Abas, F. and Martinez, F. "Craquelure analysis for content-based retrieval," in Proc. 14th Int. Conf. Digital Signal Processing, 1, 111–114, (2002).

[4] Joyeux, F. Buisson, L.Besserer, B. and Boukir, S "Detection and removal of line scratches in motion picture films," in Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition, 548–553, (1999).

[5] Kokaram, A. Morris, R. Fitzgerald, W. and Rayner, P. "Detection of missing data in image sequences," IEEE Trans. Image Process., 4, 1496–1508, (1995).

[6] Kokaram, A. Morris, R. Fitzgerald, W. and Rayner, P. "Interpolation of missing data in image sequences," IEEE Trans. Image Process., 4, 1509–1519, (1995).

[7] Bertalmio, M. Sapiro, G. Caselles, V.and Ballester, C. "Image inpainting," in Proc. SIGGRAPH, 417–424, (2000).

[8] Ballester, C. Bertalmio, M. Caselles, V. Sapiro, G.and Verdera, J. "Filling-in by joint interpolation of vector fields and gray levels," IEEE Trans. Image Process., 10, 1200–1211, (2001).

[9] Masnou S. and Morel, J. M. "Level lines based disocclusion," in *Proc. IEEE Int. Conf. Image Process.*, 3, 259 – 263, (1998).

[10] Chan, T. and Shen, J. "Non-texture inpaintings by curvature-driven diffusions," *J. Vis. Commun. Image Represent* 12, 436–449, (2001).

[11] Esedoglu, S.and Shen, J. "Digital inpainting based on the Mumford- Shah-Euler image model," Eur. J. Appl. Math., 13, 353–370, (2002).

[12] Efros, A. and Leung, T. "Texture synthesis by nonparametric sampling," in *Proc. IEEE Int. Conf. Computer Vision*, 1033–1038, (1999).

[13] Bertalmio, M. Vese, L. Sapiro, G. and Osher, S. "Simultaneous structure and texture image inpainting," *IEEE Trans. Image Process.*, 12, 882–889, (2003).

[14] Giakoumis, I. and Pitas, I. "Digital restoration of painting cracks," in *Proc. IEEE Int. Symp. Circuits and Systems*, 4, 269–272, (1998).

[15] A.Vijayaraj and R.Saravanan "Automated EB Billing System Using Gsm And Ad-Hoc Wireless Routing", International Journal of Engineering and Technology Vol.2 (5), 2010, 343-347.