Manual, Part IV

Modules

Thessaloniki 1997

Contents

1

Painting Crack estoration Module

1.1 Introduction

Many natural phenomena, like unfavorable weather conditions, cause frequently destructive effects on paintings. Wood is an anisotropic substance, exhibiting different degrees of hardness, toughness, which means that its physical characteristics in different directions are not similar. In dry environments, natural drying could lead to too rapid loss of water, resulting in non-uniform contraction and, eventually, in cracking. The cracks are frequent mostly in old paintings. The appearance of cracks on paintings deteriorates the perceived image quality. Thus, digital image processing techniques can be used in order to give the ability to the art historian to study the icon as it has been created by the painter. This technique consists of the following stages:

- Detection of the cracks.
- Separation of the brush strokes, which have been misidentified as cracks.
- Implementation of the crack filling procedure.

Cracks have usually low luminance and can be considered as being local minima with rather elongated structural characteristics. Therefore, crack detector will be applied on a luminance image, where local minima will be identified.

The detection of the cracks can be obtained with the implementation of a very useful morphological filter, called top-hat transformation. This high-pass filter can detect bright details in an image. The cracks have usually very small luminance. They look like dark details in a bright background. Thus, if we want to extract the cracks, we must negate the luminance image and then apply the top-hat transformation. The user can modify the size of the structuring element. The size of the element depends on the thickness of the crack to be detected. We must choose carefully the size of the structuring element, because, otherwise, it is possible to misidentify some thin brush strokes of small luminance.

The top-hat transformation produces a grayscale output image $t(k, l)$. If its value is large, the corresponding pixel belongs to a crack (or crack-like element). Otherwise, this pixel location corresponds to background. Therefore, a thresholding operation is required to separate cracks from the background. The threshold T can be chosen by the user. The thresholding is global, because T is chosen based on global information.

In some paintings there are thin dark brush strokes (e.g. in hair), which have almost the same features as cracks (e.g thickness, luminance). Therefore, it is possible that the top-hat transformation misclassifies these brush strokes as cracks. It would be better to separate these brush strokes, before the implementation of the crack filling procedure, in order to avoid any undesirable alterations to the original image.

A simple approach to the separation of cracks is to start from some pixels (seeds) in the thresholded output of the top-hat transformation, which represent distinct cracks. Then we *grow* them, until they cover all the cracks in the image. The pixel seeds are chosen by the user in a supervised mode. At least one seed per crack is chosen. The growth mechanism is quite simple. We must check if there are unclassified pixels with value 1 in the 8-neighborhood of each pixel of the crack. At the end of this process, the pixels in the binary image, which correspond to the brush strokes will be removed, since we have not determined seeds for them.

After identifying cracks, our aim is to restore them, possibly by filling (interpolating) image information content within cracks. The restoration method is implemented on each RGB channel independently and only on those pixels, which belong to cracks. We use the local information (neighboring pixels) for crack restoration and filling. Anisotropic diffusion has been used as an efficient nonlinear technique for crack filling. The classical anisotropic diffusion filter has been modified, by taking into account also the crack orientation.

1.2 User's guide

The following sections describe in detail the Crack Restoration module. We will use the image depicted in Figure 1.2.2 for demontration.

1.2.1 Installation

The installation of the Crack Restoration module is particularily simple. Just copy the `cracks.dll` file in the directory where EIKONA is installed. EIKONA will recognize and load the module the next time it is started, adding the option **Crack Restoration** under it's **Modules** menu.

1.2.2 Restoring a cracked image

Crack restoration can be performed by selecting **Modules, Crack Restoration, Crack Restoration** which brings up the *Crack Restoration* dialog box. Select the source and destination image buffers checking the *Color image* button if the source image is a color one. In the *Window Size* fields specify the dimensions of the window that will be used by the top-hat transformation to identify the cracks on the source image. After clicking *OK* the program displays the *Crack Propagation* window and the *Crack selection* dialog box depicted in figures 1.2.3 and 1.2.1 respectively.

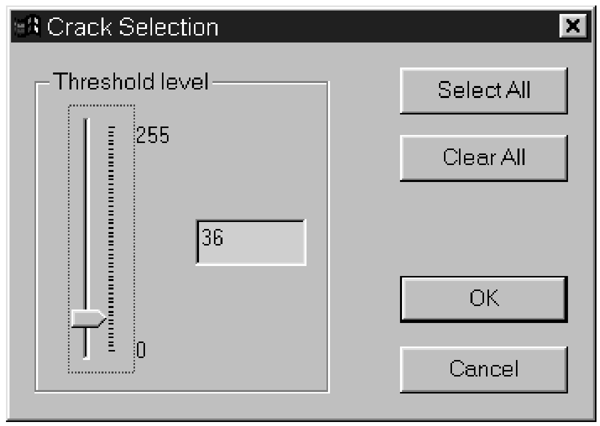

Figure 1.2.1 The Crack Selection dialog box.

You can now specify the cracks that the algorithm should fill using the mouse. Pressing the left mouse button on a crack in the *Crack Propagation* window toggles the crack for filling or not. The right mouse button opens a small window that shows a portion of the original image near the cursor so you can quickly decide whether a crack is actually a crack that should be filled or a painting element. You can quickly select or clear all cracks through the *Select All* and *Clear All* buttons of the *Crack Selection* dialog box. The slider in the *Threshold level* group can be used to adjust the crack width. The *Crack Propagation* window is continiously refreshed when you drag the slider so you can easliy determine the best threshold. After selecting some cracks and pressing *OK* the restored image appears. For the image of Figure 1.2.2 and a threshold of 20, selecting all cracks gives the restored image shown in Figure 1.2.4.

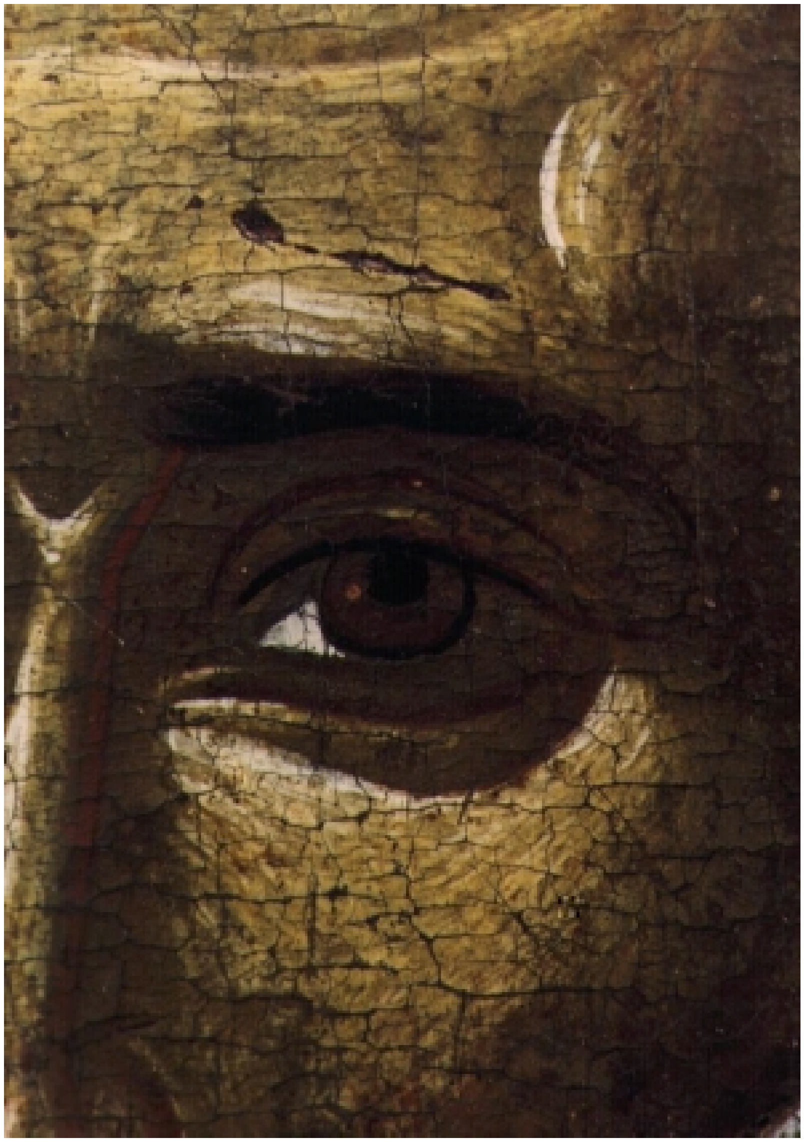

Figure 1.2.2 Original cracked image .

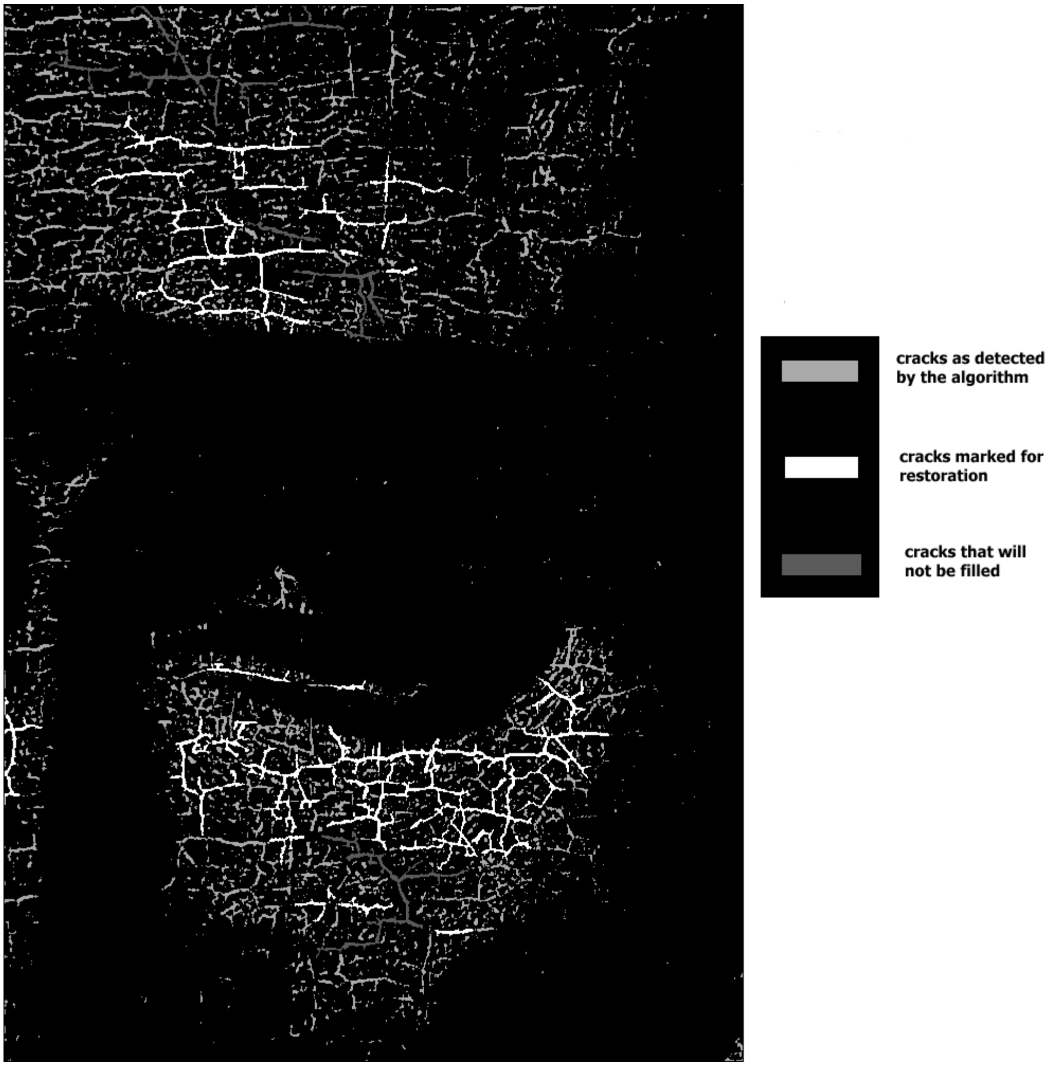

Figure 1.2.3 The Crack Propagation window.

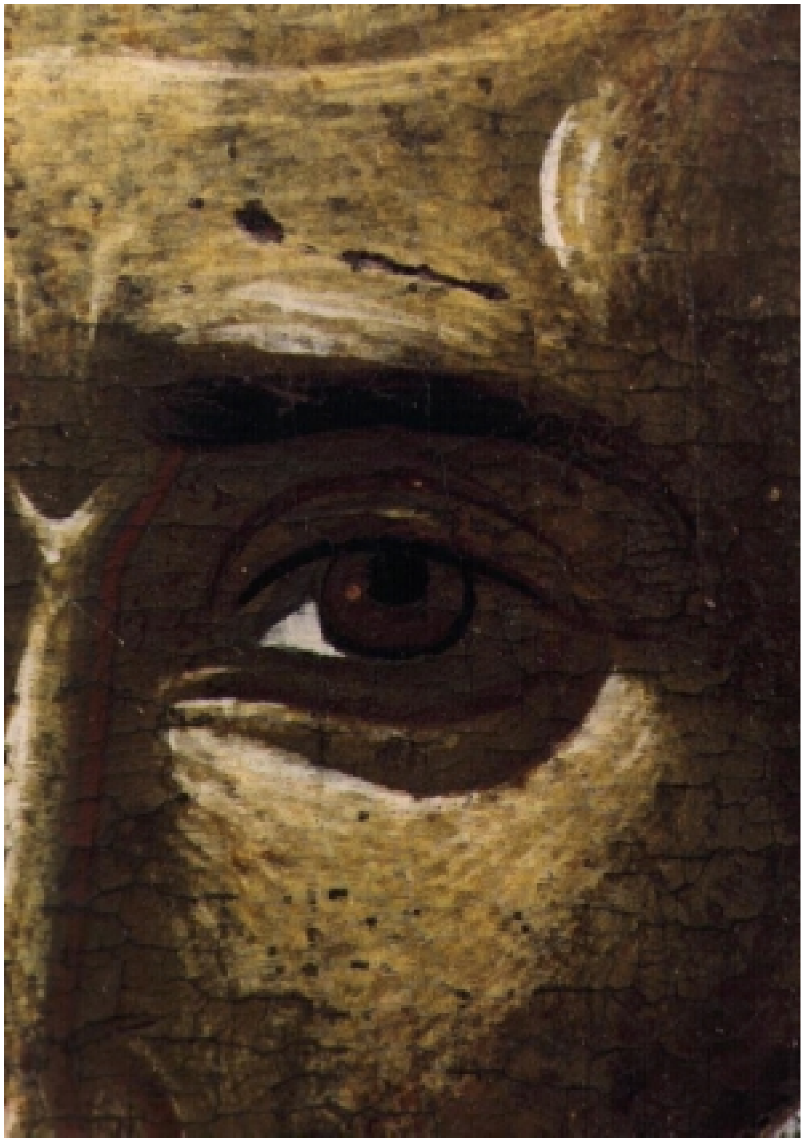

Figure 1.2.4 The restored image.