"FOXING" IN VINTAGE PHOTOGRAPHS: DETECTION AND REMOVAL

Filippo Stanco, Livio Tenze, Giovanni Ramponi

DEEI, University of Trieste
via A. Valerio, 10 - 34127 Trieste, Italy
{fstanco, ramponi}@units.it, tenze@ipl.units.it

ABSTRACT

In this paper a new algorithm to digitally restore vintage photographic prints affected by the "foxing" defect is proposed. The algorithm is composed of several automatic and consecutive steps where detection of damaged areas is coupled with removal methods. The restoration algorithms are based on an inpainting technique and on an additive/multiplicative model. The first one replaces the lost data; the second one fixes the defective area searching for optimal parameters around the damaged area. The obtained results show that the foxing spots are completely removed without producing significant artifacts.

1. INTRODUCTION

Old photographic prints should be preserved with care to avoid irreversible damages. Experts suggest to maintain as long as possible the good quality of a print by handling it with care and saving it under optimal humidity conditions. However, there are many damaged pictures that need to be restored. As expressed in [1] the classical physical restoration is extremely expensive and automated virtual restoration is hence required to obtain quick, simple and effective results. In this paper we propose a technique to restore photographic prints affected by foxing. The term "foxing" was used for the first time in the 18th Century to indicate the scattered reddish-brown (the color of a fox) spots on the surface of paper in old books [2, 3, 4]. The same technical word was introduced in photography to refer to a similar chemical damage on the prints. Foxing is characterized by a dark-brown center and an area where the color is smoothed. In the center all the original information is covered by the spot, and hence it is considered as lacking. The area around the center, on the contrary, can include residual original information that should be enhanced. Fig. 1 reports two examples of "foxed" images.

The causes of foxing are not completely understood; probably it depends on joined fungal activity and metal-induced degradation. The paper used in oldest prints has microorganisms that can remain latent for decades awaiting conditions appropriate for growth. Moreover, airborne spores can attach the paper, creating colonies of foxing. Another element that seems to accelerate foxing is the presence of iron in the paper. Despite the statistics showing that foxed areas have higher proportion of acid and iron than clean ones, what role iron has in creating or accelerating foxing has to be demonstrated yet. However, if the relative humidity is below 50% and we use modern paper without iron, the foxing is strongly reduced.

On photographic paper, the effects of foxing may be chemically reduced to a reasonable extent by use of a reducing agent such as sodium borohydride or calcium hypochlorite [2]. These physical restoration procedures are very expensive, complex, and are performed only by skilled personnel. Often this is not justified by prints value. In this case it can be helpful to perform a digital automated restoration to reduce the cost and the processing time.

In this paper we suggest an algorithm to restore foxing stains. It alternates the detection phases with the restoration ones. Both the detection and the restoration methods are automatic and do not need user assistance. The detection is based on the color properties of foxing. The restoration method uses inpainting if there is no residual information, while it exploits an additive/multiplicative model elsewhere.

Inpainting algorithms [5, 6, 7, 8], solve the problem of disocclusion (i.e. the recovery of hidden parts of objects in a digital image). They connect T-junctions at the occluding boundaries by propagating both the gradient direction and gray-values of the image in a band surrounding the hole to be filled-in. Isophote (region with the same level lines) directions are obtained by computing at each pixel along the inpainting contour a gradient vector and by rotating the resulting vector by 90 degrees. This method tries to propagate the information while preserving the edges. After few iterations of the inpainting process, the algorithm performs an anisotropic diffusion run to preserve boundaries across the inpainted region. On the other hand, an additive/multiplicative model [9] reconstructs the low-spatial frequency components of the image via an approximation of the uncorrupted image, and the high-spatial frequency components by means of a suitable model.

To our best knowledge, no reference can be found in the open literature to automated methods dealing with the problem of foxing.

The rest of the paper is organized as follows: Section II describes the proposed algorithm, and Section III shows some experimental results. A Conclusions section ends the paper.

2. THE PROPOSED ALGORITHM

First of all the photographic prints are acquired with a scanner or other acquisition systems to obtain a high-resolution digitized version of the image. This image I is the input of our algorithm. It starts with a detection phase that determines where the foxing is located. Then an inpainting algorithm restores the detected areas. A second detection, obtained extending the previous one by searching for similar points, is performed before the last step that fixes the defective areas, while preserving the original information where available. All these steps are automatic.

Detection

The method tries to find where the foxing spots are located. In the Introduction we described this damage as a set of reddish-brown spots. Vintage photographic prints are usually gray or sepia. In the first case the C_r chrominance matrix related to red is null, while in the second one it presents a histogram whose peak is in the center. Chemical alterations however change this situation and, in particular, in presence of foxing artifacts they yield C_r matrices with the typical histogram shown in Fig. 2. The histogram presents a tail on the right formed by a set of (usually small) bins having almost uniform amplitude, and the peak is in the left

(a)

(b)

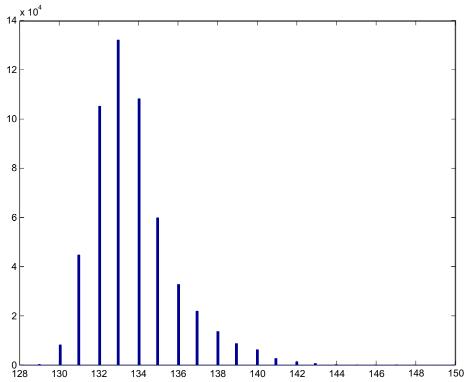
Figure 1: (a) and (b) two examples of foxing

portion of the histogram. The bins on the right tail represent all the points damaged by foxing.

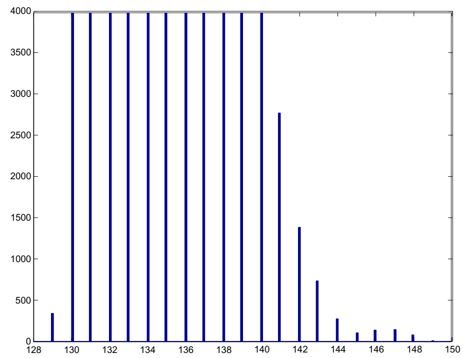
The detection procedure searches for all the connected image parts represented by this tail. If I is an RGB image, to perform the detection we change the RGB color space into YC_bC_r. Then the normalized histogram with 256 bins of C_r is computed. To find all the bins representing the damaged pixels we start from the right of the histogram, and we mark as foxing all the bins whose difference between their height and the height of the previous bin is less than a fixed value $Th1$. The *center* of the last bin marked as foxing gives the value a that is used to perform a thresholding over the matrix C_r to find Fox. Fox is a matrix where the coordinate of foxed pixels are represented as a 0 value. More precisely:

$$Fox(i,j) = \begin{cases} 0 & C_r(i,j) > a \\ 1 & elsewhere \end{cases}$$

where a is the above determined value. This detection step could extract isolated points; they do not represent relevant damaged areas, and hence are expunged using a simple median filter. The black regions in Fig. 3(a) are the foxed areas as labeled by the detection phase.



(a)



(b)

Figure 2: (a) Histogram of C_r; (b) A detail of the histogram in (a)

Restoration: Inpainting

All the pixels whose value in Fox is 0 are modified with an inpainting-like procedure. This is applied to the darker brown pixels in the foxing spots, where there is no residual information. In this step we propagate inside the foxed areas the boundary colors. Each pixel in the border of the detected foxing is assigned the average of the colors close to the pixel but outside the foxing. More precisely we set Ω as the area to be inpainted:

$$\Omega = \{I(i,j) \quad with \quad Fox(i,j) = 0\}$$

If we call N_{ij} the 3×3 neighborhood of $I(i,j)$, then the boundary pixels of $I(i,j) \in \Omega$ are:

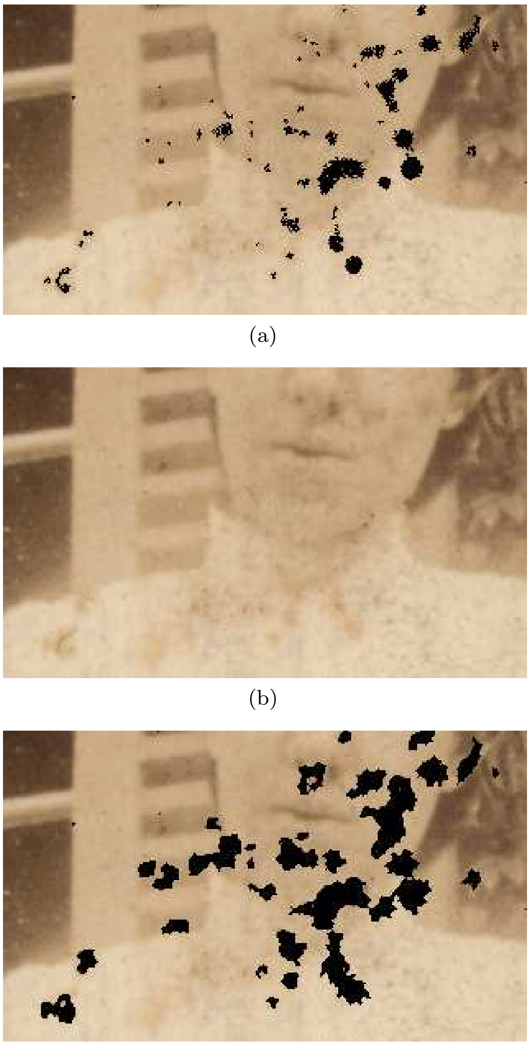
$$\partial\Omega_{ij} = \{I(p,q)| \quad (I(p,q) \notin \Omega) \wedge (I(p,q) \in N_{ij})\}.$$

We remark that $\partial\Omega_{ij}$ contains pixels that do not belong to Ω but are near $I(i,j)$ that is in Ω.

The inpainting procedure yielding I' can be described as follows:

$While\ \Omega \neq \emptyset$
  $For\ each\ pixel\ I(i,j) \in \Omega\ with\ \partial\Omega_{ij} \neq \emptyset$
    $I'(i,j) = average\ of\ pixels \in \partial\Omega_{ij}$
    $\Omega = \Omega \setminus \{I(i,j)\}$
    $\partial\Omega_{ij} = \partial\Omega_{ij} \cup \{I(i,j)\}$
  $end\ For$
$end\ While$

The new image I' presents more homogeneous foxing spots. Fig. 3(b) shows the image in Fig. 1(a) after inpainting. It can be seen that the foxing stains are still visible even if their saturation is reduced. This happens because in this step we have replaced the flawed regions using their outlines which are partially affected by foxing.



(a)

(b)

(c)

Figure 3: (a) Position of foxing defects; (b) Image in Fig. 1(a) after Step 1. (c) New detection of the foxed points

Enlarged Detection

In this stage we extend the previous detection step by finding all the pixels where the original information is only partially affected by foxing. They are characterized by a lighter coloring than the center of the foxing and their position is near the reddish-brown spot. Therefore we search them starting from the previous detection. If the color of a pixel near the foxing spot is close to the color of the pixels in the inpainted region, then that pixel is labeled as foxed. This procedure can be described in the following pseudo-code:

Set $\Omega = \{I'(i,j) \quad with \quad Fox(i,j) = 0\}$
Set $I'' = I'$
Repeat
  $\Omega' = \Omega$
  For $I''(p,q) \in \partial\Omega_{ij}$ *with* $I''(i,j) \in \Omega$
    If $|I''(p,q) - I''(i,j)| < Th2$
      $Fox(p,q) = 0$
      $\Omega = \Omega \cup \{I''(p,q)\}$
      $I''(p,q) = I''(i,j)$
    end If
  end For
Until $\Omega' \neq \Omega$

Where $Th2$ is a suitable threshold. After this step Fox is the updated detection map. Fig 3(c) shows the new detection result in our example.

Final restoration: additive/multiplicative model

The image I', produced by the previous step, has been treated with the inpainting technique, but some foxing spots are still visible; the Fox map contains the positions of the foxed pixels. Now we eliminate the foxing, taking into account the residual information in the detected regions. To this purpose we propose an additive-multiplicative model applied to each foxed area. Similarly to [9], we automatically extract the model parameters examining a suitable region around the foxing spot. If we denote with J the uncorrupted portion of the image I', and with Ω^* a single foxing spot, we can write our model as follows:

$$I'(\Omega) = \alpha * J(\Omega^*) + \beta \tag{1}$$

where α and β are the parameters to be estimated. We should be able to obtain an estimaton of the error-free image by inverting Eq. 1. Therefore, the problem is reduced to determine suitable values for the α and β parameters. To this purpose, denoting with $var[.]$ and $E[.]$ respectively the variance and the mean in [.], and using Eq. 1, we can state that

$$\begin{aligned} var[I'(\Omega^*)] &= \alpha^2 * var[J(\Omega^*)] \\ E[I'(\Omega^*)] &= \alpha * E[J(\Omega^*)] + \beta \end{aligned} \tag{2}$$

However, the variance and the average of the uncorrupted image J are unknown. We denote with $\widetilde{\Omega}^*$ an area around the foxing Ω^* with width W. To avoid using pixels that are too close to the spot, and hence are unreliable, $\widetilde{\Omega}^*$ is automatically determined as a strip around the blotch shifted by S pixels away from the contour. This shift ensures that an erroneous detection of the border will not compromise the accuracy of the final result. In order to solve Eq. 2, we approximate $J(\Omega^*)$ with $I'(\widetilde{\Omega}^*)$. This assumption permits to estimate the values of α and β:

$$\begin{aligned} \widetilde{\alpha} &= \sqrt{\frac{var[I'(\Omega^*)]}{var[I'(\widetilde{\Omega}^*)]}} \\ \widetilde{\beta} &= E[I'(\Omega^*)] - \alpha * E[I'(\widetilde{\Omega}^*)]; \end{aligned} \tag{3}$$

Inverting Eq. 1 using the estimated parameters $\widetilde{\alpha}$ and $\widetilde{\beta}$, it is possible to compute the restored value for each pixel of the foxing regions. Formally:

$$\widetilde{J}(\Omega^*) = (I'(\Omega^*) - \widetilde{\beta})/\widetilde{\alpha}. \tag{4}$$

where $\widetilde{J}$ is the restored image. After this process the area inside the foxing spot is perfectly restored and it is similar to the rest of the image. However, the contour of the stain could still be apparent. This behaviour is caused by the fact that, during the detection step, some damaged pixels are not labeled as belonging to a foxing region. To avoid this artifact we use a linear interpolation across the border of the foxing spot. To compute this interpolation, the luminance gradient is evaluated for each pixel p in the contour. Then, an array of $2L+1$ pixels centred on p is considered. This array contains L pixels belonging to the foxing region and L outside the spot chosen along the gradient direction indicated in p. If we denote with P_{start} the first pixel of the array and with P_{end} the last one, a linear interpolation is performed between these two values according to the distance between the pixels. If $d(P_i) = \frac{P_i - Pend}{P_{start} - p_{end}}$ is the normalized distance of each pixel in the array from the P_{start} position, the new intensity values are:

$$\widetilde{J}(P_i) = d(P_i) * \widetilde{J}(P_{start}) + (1 - d(P_i)) * \widetilde{J}(P_{end}) \quad (5)$$

It has to be noticed that the new values in the vector depend only on the intensities in P_{start} and P_{end}, and the original values are not considered. Due to the fact that the gradient orientation can be very different even for neighboring pixels, it could happen that there are some un-processed pixels. Therefore, we search all the unchanged pixels in a border strip of width $2L+1$, and we assign a gray level value to each pixel corresponding to the average of its already interpolated neighbors in a 3×3 mask. All the operations described in this subsection are repeated for each foxing spot Ω^*. The output of this iteration is the final restored image.

3. EXPERIMENTAL RESULTS

This section shows some results obtained applying the algorithm to old photographic prints. We remark that the method does not need any selection by the user and does not process the entire image if it is not necessary. The algorithm parameters have been set experimentally as follows: the threshold $Th1$ for the detection in the normalized histogram is set equal to 0.1; $Th2$ is 2; the distance S between the foxing and the uncorrupted area is chosen equal to 7; the width W of the $\widetilde{\Omega}^*$ is 10; and L=3 in Step3.

Fig. 4 reports some results obtained applying the proposed algorithm. It should be noted that the proposed algorithm has been tested with uncompressed and compressed images, in Tiff and Jpeg format respectively. In all cases the algorithm provided good results. The goodness of the results cannot be estimated quantitatively due to the fact that the images are originally affected by foxing: it is not possible to compare the performances via MSE or PSNR.

(a)

(b)

Figure 4: (a) Image in Fig. 1(a) after restoration; (b) Image in Fig. 1(b) after restoration

4. CONCLUSIONS

In this paper an algorithm to remove foxing defects from vintage photographic prints has been proposed. The method works preserving the residual information where available and replacing irreversible damaged area with data obtained considering the area around the defect.

The algorithm works without any user intervention for both the detection and the restoration. Our experiments show the efficacy of the method.

5. ACKNOWLEDGEMENTS

We thank F.lli Alinari s.p.a. for providing all the pictures used in our experiments.

This work has been partially supported by a grant of the Regione Friuli Venezia Giulia.

REFERENCES

[1] F. Stanco, G. Ramponi, A. de Polo, "Towards the Automated Restoration of Old Photographic Prints: a Survey", in *Proc. IEEE-EUROCON 2003*, Ljubljana, Slovenia, September 2003;

[2] http://palimpsest.stanford.edu/don/dt/dt1434.html

[3] http://www.octavo.com/collections/essays/foxing.html

[4] T.D. Beckwith, W.H. Swanson, and T.M. Iiams, *Deterioration of Paper: the Cause and Effect of Foxing*, University of California Publications in Biological Science 1(13):299-356, 1940.

[5] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, "Filling-in by Joint Interpolation of Vector Fields and Gray Levels", *IEEE Transactions on Image Processing*, vol. 10 (8), pp. 1200- 1211, 2001.

[6] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, "Image Inpainting", in *Proc. SIGGRAPH 2000*, pp. 417-424, 2000.

[7] M.M. Oliveira, B. Bowen, R. McKenna, Y.S. Chang, "Fast Digital Image Inpainting", in *Proc. International Conference on Visualization, Imaging and Image Processing VIIP 2001*, Marbella, Spain, pp. 261-266, 2001.

[8] S. Masnou, J.M. Morel, "Level lines based disocclusion", in *Proc. ICIP 98*, pp. 259-263, 1998.

[9] F. Stanco, G. Ramponi, L. Tenze, "Removal of Semi-Transparent Blotches in Old Photographic prints", in *Proc. 5th COST 276 Workshop*, Praha, Czech Republic, October 2003;