

# DETECTION AND REMOVAL OF LONG SCRATCH LINES IN AGED FILMS

*Timothy K. Shih and Louis H. Lin*

Department of Comp. Sci. and Information  
Engineering Tamkang University, Taiwan  
tshih@cs.tku.edu.tw

*Wonjun Lee*

Department of Computer Science and  
Engineering Korea University, Seoul, Korea  
wlee@korea.ac.kr

## ABSTRACT

Historical films usually have defects. We study the type of defects, and propose a series of solutions to detect defects before they are repaired by our inpainting algorithms. This paper focuses on a difficult issue to locate long vertical line defects in aged films. A progressive detection algorithm is proposed. We are able to detect more than 86% (recall rate) of effective line defects. These line defects are then removed step by step. The experiments use real historical video collected from national museum and public channel, instead of using computer generated noise. The results are visually pleasant based on our subjective evaluation by volunteers.

## 1. INTRODUCTION

Inpainting on still image are not new. In the literature, damages of aged photos can be repaired manually or automatically by referencing spatial or frequency properties. Usually, the results are pleasant on photos with small or thin area of defects. On the other hand, large objects can be removed and replaced by using prioritized information on the same photo. We also proposed a solution [6] to take a further step to divide an image into several layers. Each layer is inpainted and a fusion function is able to reconstruct the image from the inpainted layers. These processes of image inpainting require the users to precisely identify the area of defects or the objects to be removed. However, to manually identify defects in video is not feasible due to the number of frames. In general, there are three types of defects. Spike in dark or bright intensity usually occurs in one or two frames at a random location. We have proposed a solution [7] to detect spikes using temporal and special properties. The detected spike defects are efficiently repaired by estimation of motions and pixel continuations. Another mechanism to restore spike defects is found in [5]. The second type of defects is lone vertical lines which usually cover a large percentage of frame height. Detection of vertical lines is a different problem compared to spike detection. The line defects are usually in a nearby location in several frames (up to continuous frames of several minutes). As long as the line defects are located, they can be repaired by methods in [4] and [6]. The third type of defects

is due to the shaking of camera or the effect of lights. Frame jitters due to shaking of camera. Frame flicker means the fluctuation of intensity. Jitters and flicker can be recovered by statistical review of frame properties.

We focus on the second type of defect detection, which was also discussed in [1, 2, 3]. Line defects are hard to detect due to several reasons. Firstly, the intensity of defect (either dark or bright) is unstable and hard to predict. Secondly, the occurrences of line defects are random but with some near-predictable characteristics (e.g., vertical length or frequency). Finally, it is possible that a detection mechanism fails to differentiate a real object, such as a vertical rope, from a line defect.

This paper proposes a series of algorithms to detect and remove lone vertical line defects. We propose our detection algorithm in section 2. We then discuss the strategy of inpainting procedure in section 3, before our comparison of results and conclusions.

## 2. THE DETECTION MECHANISM

Detection of line scratches (i.e., long vertical line defects) is different from detecting spike defects [5, 7], which use motion estimation and the characteristics of spikes to find defect candidates. Since line defects usually occurs in a fixed or near position, spatial property seems to be more important as compared to occurrences of defects in continuous frames. The work proposed in [2] and [3] uses a morphological detector to find the local minimum of gray levels. The intensities of pixels along vertical directions are projected to obtain the value of histogram. We implement the projection with an extra parameter to consider the range of intensity variations along the X-axis for the line defects (i.e., thick defect lines). The result is shown in figure 1, where line defects are highlighted by red boxes and the projection of intensity is shown in the window below the original frame. However, intensity projection does not work very well since the background in each frame varies, as we can see the surrounding information of the two line defects marked in red boxes are different (i.e., dark sky versus white clothes). Thus, our first strategy subdivides each frame into a number of horizontal bands. As illustrated in figure 2, the frame in figure 1 is divided into 6 bands. Figure 2(a) and 2(c) are the top band and the bottom band.

Figure 2(b) and 2(d) are intensity projections with respect to the two bands. With the help of our strategy of band fusion, local minimum works much better. The number of bands in our system varies from 2 to 10 (the sufficient maximum). The decision of band number,  $n$ , is due to a maximum starching strategy. For each subdivided band, defects are detected. Defects at the same or near position are combined. We then find the maximum length of a line defect at a position among different frame subdivisions. This strategy also finds the approximation of a realistic line length since some line defects may cover only a portion of frame height.

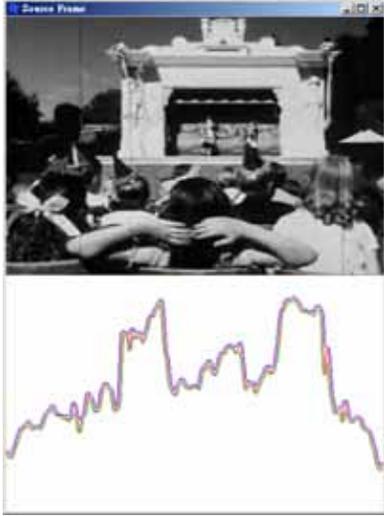


Figure 1: Projection of Intensity Values along the X-axis

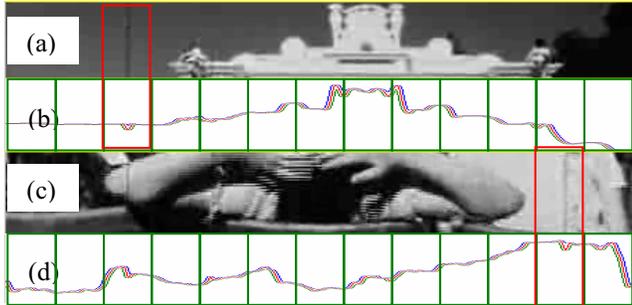


Figure 2: Subdivision of frame into bands ((a) and (c)) and their intensity projection ((b) and (d))

Our second strategy is to use watching windows where local minimum may occur. As show in figure 2(b) and 2(d), each band is divided into several windows. The exact width of window is based on experiment. We found a suitable window size of 40 pixels. This threshold was due to the resolutions of 12 video examples that we used. However, to avoid boundary conditions, the positions of windows are not fixed. Conclusively, our detection mechanism follows.

## 2.1. Defect Identification

Let  $n$  be the number of bands,  $w$  be the number of windows in our detection. We define a threshold,  $t$ , for intensity projection. This threshold will be iterated in our progressive

inpainting algorithm (discussed in section 3). We use  $W_{i,j}$  where  $1 \leq i \leq n$ ,  $1 \leq j \leq w$ , to denote the  $j$ -th watching window in band  $i$ . Note that, all bands have the same number of windows. We want to find one potential horizontal position of the defect within a watching window,  $W_{i,j}$ . Let  $x$  be any horizontal position (i.e., an x-axis coordinate) in  $W_{i,j}$ . We define

$$B_{np_{i,j}} = \max(h^+(x) - h^-(x)), \text{ where } h^+(x) - h^-(x) >$$

Thus,  $B_{np_{i,j}}$  is the horizontal position for a possible line defect in  $W_{i,j}$  where  $n$  bands were used. Note that, the differences between  $h^+(x)$  and  $h^-(x)$  may be less than the threshold  $t$ . As such, no defect is detected in the current band and window. The threshold will be decreased in the next iteration of the progressive inpainting algorithm to detect additional defects. The two functions,  $h^+(x)$  and  $h^-(x)$  are defined as

$$\begin{aligned} \forall x_i, \quad 0 \leq i \leq \text{frame\_width} - 1 \\ \forall \theta, \quad 0 \leq \theta \leq 359 \\ x'_i = x_i + \cos\left(\theta \times \frac{\pi}{180}\right) \times r \\ y'_i = IP(x_i) + \sin\left(\theta \times \frac{\pi}{180}\right) \times r \\ h^+(x'_i) = \max\left(h^+(x_{i-1}), y'_i\right) \\ h^-(x'_i) = \min\left(h^-(x_{i-1}), y'_i\right) \end{aligned}$$

The above equations iterate through each horizontal position till the end of frame width. The function  $IP(x_i)$  accumulates intensities of each pixel at position  $x_i$ . We use a range,  $r$ , to perform morphology on the intensity projections. Typically,  $r$  is set to 5 pixels. The function  $h^+(x)$  and  $h^-(x)$  result in two projection curves, as illustrates in figure 1. The red (central) line is the function  $IP(x_i)$ . The blue (above) line and the green (below) line are the functions  $h^+(x)$  and  $h^-(x)$ , respectively.

Next, we accumulate defect continuations at the same horizontal position through different bands. Let  $C_{np_j}$  be the horizontal position of the accumulated continuous defect line located at  $B_{np_{i,j}}$ . We define

$$D_j = \max(C_{ip_j}), \text{ where } 1 \leq k \leq n$$

which is the line defect with a maximum length in the  $j$ -th watching window.  $D_j$  is computed by searching on various number of band subdivisions of a video frame. In general, if the number of bands is too small, some short defect segments may not be detected. If the number of bands is too large, it will be hard to find a local minimum. Thus, the searching algorithm takes an optimal approach to search for the optimal solution among different number of bands. With a proper threshold of intensity projection (i.e.,  $t$ ), the detected defect lines may achieve a best approximation.

## 2.2. Defect Tracking

Our previous discussion only uses spatial properties. As indicated in [2] and [3], temporal properties can be used to track and predict line defects. Kalman filter was used as a tracking process to determine whether the detected defects fit the scenario of line defects. However, the approach is restrictive, as also indicated in [1]. We study the behavior of line defects. We found that, among the 12 video clips that we collected, line defects occur in a near position within a range of the window size that we defined (i.e., 40 pixels). However, the duration of occurrence is hard to predict. In most cases where the line defects occupy the entire frame height, the duration of defects is very long. However, some line defects with a shorter length (less than 30% of frame height) or discontinued defects (i.e., line defects divided into multiple segments) may occur in a relatively random position as compared to the line defects with a full length. Thus, our strategy of using temporal property in line defect detection is divided into two steps. We firstly eliminate long defect lines. Short defect lines are then removed. Since defect tracking and removal are in the same iterative procedure, we discuss them in the next section.

## 3. THE PROGRESSIVE INPAINTING ALGORITHM

The progressive inpainting algorithm takes two steps. The differentiation depends on the length of line defects. Recall that  $D_j$  is the line defect with a maximum length in the  $j$ -th watching window. Let  $F_x D_j$  be the  $D_j$  in frame  $x$ , where  $1 \leq x \leq fn$ ,  $1 \leq j \leq w$ ,  $fn$  is the number of frames in the video sequence. The first step of inpainting follows:

Set  $t = t_0$

Repeat until no  $F_x D_j$  is found

Find  $F_x D_j$  with  $length(F_x D_j) = \text{frame\_height}$   
and  $duration(F_x D_j) > d_1$

$inpaint(F_x D_j)$

update  $t$  by 10%

shift the  $j$ -th watching window by  $\delta_w$

The initial intensity projection threshold,  $t_0$ , is set to the minimum of intensity projections among all bands. And,  $duration(F_x D_j)$  is a temporal estimation function to track  $F_x D_j$ . This function finds  $F_x D_j$  within a nearby horizontal position varies by 5 pixels and within a continuation of  $d_1$  frames, where  $d_1$  is set to 30 (based on our experimental results). The defect  $F_x D_j$  is inpainted. We then update the intensity projection threshold by 10 percents. For detecting dark defects, the threshold is decreased. Detecting bright defects takes an opposite direction. Meanwhile, the position of watching window is shift by 5 pixels (i.e.,  $\delta_w$ ). In practical experiments, the first step of inpainting only takes one or two iterations to find defects. Figure 3 shows an example of full-length line defect.

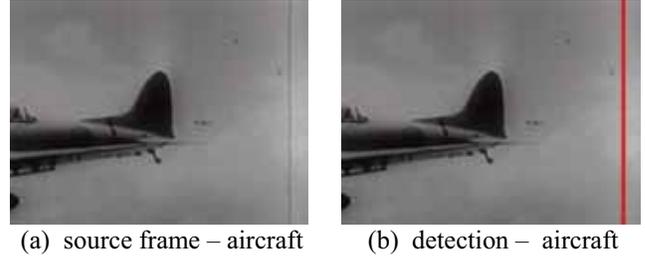


Figure 3: Detected Line Defect with a Full Length

The differences between the first step of detection and the second step depend on the *length* and the *duration* (and the threshold  $d_1$ ) functions. However, the second step of detection incorporates another factor, which is the actual length of the line defect. Since the actual length of defect is not necessary equal to band width, a searching strategy is implemented. We use a simple block matching strategy to extend the length. For each defect  $F_x D_j$ , we consider the scope of bands w. r. t. the  $j$ -th watching window (i.e., a continuation of one or more bands), where the defect is located. The bands before the scope and after the scope are searched. Since each  $D_j$  locates at a horizontal position, we search for similar 3 by 3 blocks at the same horizontal position. A simple method, SAD (Sum of Absolute Differences), is used. In addition, in the second step of detection, we avoid line defects that were detected and inpainted in previous iterations. The  $extract(F_x D_j)$  function remembers positions of defects, as well as its neighborhood positions within a range of  $2/ID$ , where  $ID$  is the *inpainting distance* (to be discussed). Pixels in these preserved positions are extracted from being used as a candidate defects. The number of continuous frames  $d_2$  is set to 10. In summary, the second step of inpainting follows:

Set  $t = t_0$

Repeat until no  $F_x D_j$  is found

Find  $F_x D_j$  with  $duration(F_x D_j) > d_2$

$extend(F_x D_j)$

$extract(F_x D_j)$

$inpaint(F_x D_j)$

update  $t$  by 10%

shift the  $j$ -th watching window by  $\delta_w$

One important function in the above two steps of algorithms is the  $inpaint(F_x D_j)$  procedure. We define an inpainting distance,  $ID$ , which represents an estimation of the width of line defects. The use of inpainting distance depends on practical experiments. In most videos, we use  $ID = 2$ . The position of  $F_x D_j$  alone with inpainting distance decides the vertical strip to be inpainted. In case that the line defects are wider than the distance, extra portion of the defect will be detected in the next iteration. We use an inpainting algorithm that we have proposed in [6]. Examples of detection and inpainting results are illustrated in figure 4. The paper title (on the first page) should begin 1.38 inches (35 mm) from the top edge of the page, centered, completely capitalized, and in Times 14-point, boldface type. The authors' name(s) and affiliation(s) appear below

the title in capital and lower case letters. Papers with multiple authors and affiliations may require two or more lines for this information.

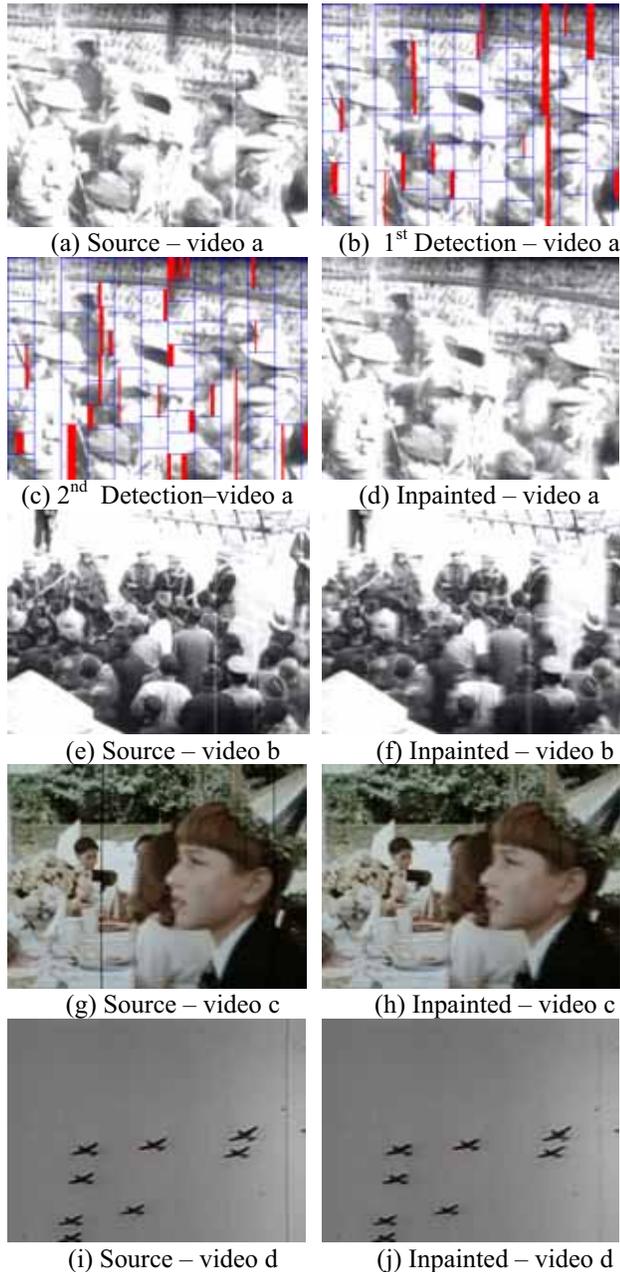


Figure 4: Detection and Inpainting Results

#### 4. EXPERIMENTAL RESULTS

We collected 12 video clips. Most of our videos are extracted from B/W historical films. These clips are stored in an uncompressed format to avoid block effects, which could affect our detection. We define “effective defects” as if the defect can be visually seen by a person. As shown in figure 4, the original frames are placed on the left columns (i.e., (a), (e), (g), and (i)) and inpainted frames are show on

the right. Human observation is used on 4 video clips since it is very time consuming to check defect on each frame. Let  $T$  be the total number of defects (ground truth),  $D$  be the number of detected defects,  $N$  be the number of incorrectly detected defects, and  $M$  be the number of missing detected defects. We proof  $T = D - N + M$  to show our careful experiment. Table 1 summarizes these numbers.

Table 1: Detection Results by Human Observation

	$T$	$D$	$N$	$M$	Precision	Recall
Video a	275	832	559	2	0.598	0.997
Video b	205	705	522	22	0.574	0.969
Video c	66	372	306	0	0.560	1.000
Video d	1794	1339	974	1429	0.578	0.483

Precision and recall ratios with respect to our definitions are used: Precision =  $D/(D+N) = 0.5775$  and Recall =  $D/(D+M) = 0.8623$  (in average). In addition, we use 10 volunteer students to look at the inpainted video clips. It was proved that the results are visually pleasant, with most defects removed. However, our inpainting algorithm [6] should be redesigned for the particular requirement. Also, video d was taken in World War II. A lot of unpredictable issues remain, which affect our recall ratio.

#### 5. CONCLUSIONS

This paper presents a contribution to a challenge issue which finds long vertical line defects in aged films. We propose a new approach based on two general strategies: subdivision of video bands and progressive detection/inpainting. The solution was proved to be feasible and more than 86% of effective defects are detected and removed. We will combine the newly proposed techniques with our inpainting system [7] which removes spikes. Our next step is to solve the frame flicker and frame jitters problems.

#### REFERENCES

- [1] Vittoria Bruni and Domenico Vitulano, “A Generalized Model for Scratch Detection,” *IEEE Transaction son Image Processing*, Vol. 13. No. 1, January 2004.
- [2] L. Joyeux, O. Buisson, B. Besserer, S. Boukir, “Detection and removal of line scratches in motion picture films,” *International Conference on Computer Vision and Pattern Recognition*, 1999.
- [3] L. Joyeux, S. Boukir, and B. Besserer, “Film Line Scratch Removal Using Kalman Filtering and Bayesian Restoration,” *Proceedings of WACV2000*, Palm Springs, CA, Dec. 2000.
- [4] Emmanuelle Lecan, Louis Laborelli, Jean-Hugues Chenot, “Missing Data Correction in Still Images and Image Sequences,” *Proceedings of the 10th International ACM Multimedia conference 2002*.
- [5] A. Machi, F. Collura, “Accurate spatio-temporal restoration of compact single frame defects in aged motion picture,” *Proceedings of 12th International Conference on Image Analysis and Processing*, 2003.
- [6] Timothy K. Shih., Rong-Chi Chang, Liang-Chen Lu and Huan-Chi Huang, “Multi-Layer Inpainting on Chinese Artwork,” *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo*, Taipei, Taiwan, June 27 - June 30, 2004.
- [7] Timothy K. Shih, Rong-Chi Chang and Yu-Ping Chen, “Motion Picture Inpainting on Aged Films,” *Proceedings of the 2005 ACM Multimedia Conference*, Singapore, November 2005.