International Journal on Future Revolution in Computer Science & Communication Engineering ISSN: 2454-4248
Volume: 3 Issue: 7 69 – 78

Efficient Method For Scratch Lines Noise Removal From Video

Vaibhav S. Manikpure
Final Year Student, M.Tech. ECE Department
Abha Gaikwad-Patil College of Engineering,
Nagpur Maharashtra, India

Prof. Deepak R. Bhoyar
Asst. Professor, Department of ECE
Abha Gaikwad-Patil College of Engineering,
Nagpur Maharashtra, India

Prof. Sagar Ghormade
HOD, Department of ECE
Abha Gaikwad-Patil College of Engineering,
Nagpur Maharashtra, India

Abstract— The digitalization and transfer of older films into high definition (HD) formats imply that high quality of restoration is necessary. Now a day's Digital film restoration is an area under discussion of increasing interest to researchers and film archives alike. Old films, including cultural heritage masterpieces, are being digitally premastered and transferred into novel, higher quality formats and distributed through various means such as DVD, Blu-ray or HD pictures. Detection of Line scratches in old movies is a particularly difficult problem due to the variable spatiotemporal characteristics of this deficiency. Some of the main problems consist of sensitivity to noise and texture, and false detections due to thin vertical structures belonging to the scene. Automatic finding of image damaged regions is the key to automatic video image in-painting. Vertical scratches are the common damages in the old film. As the film is a collection of number of frames arrayed together to produce a motion sequence hence it becomes a lengthy and tedious work to process any video format in any manner. Normally if any scratch or noise generated on films it remains as it is on many frames in sequence in film which can be benefitted by the removal process by initially checking noise area on earlier slide. Hence proposed system is aimed at designing and developing of line scratches detection from old films and remove it. A line scratches detection algorithm based on edge detection is proposed. Edge detection is nothing but an image processing technique for finding the boundaries of objects inside images. The proposed algorithm first uses the operator which has the largest response to the vertical edge in Sobel operator to detect edges, and then uses canny operator to detect edges further. Third, we detect vertical lines in the image through probabilistic Hough transform. Finally, we obtain the true locations of the vertical lines scratches through morphology and width constraints. We contribute for removal of scratches using a new nonlinear continued fraction method dealing with both spatial and temporal information around the scratch is investigated in the restoration stage.

Keywords— Scratch detection, old film restorations, line detection, image inpainting, false detection

*****

I. INTRODUCTION

In recent times, digital image inpainting has become a demanding field in digital image processing. Digital inpainting techniques have found broad applications in digital restoration of ancient paintings for preservation purposes, restoration of old films and photographs. However, most repair work require professional to determine manually repaired area, or processes all pixels indiscriminately, so that these inpainting techniques cannot be used in huge quantity image processing and video repair. The key to solve this problem is automatic detection of image damaged regions. Base on these, in proposed system, focus on the common line scratches in the old films and discuss the scratch detection technology before the digital inpainting, which is the origin for the automatic restoration.

In early age, video films are recorded in digital tapes. Manual handling, dust, dirt or abrasion by contacts of a tape with mechanical parts like film projector affects the storage media hardware and hence causes scratches in video. Here, we think about the last defect, the line scratch, regularly caused by a scratch to the physical video. These line scratches become visible as thin bright or dark lines which are roughly straight and vertical. These defects also present the singular characteristic of temporal persistence, meaning that they remain in the same or a similar spatial position for a number of frames. Therefore, line scratch detection algorithms must be specially adapted to this defect. On the other hand, these characteristics are very variable, making line scratch detection and restoration a particularly complicated challenge. The restoration of old films is a subject of primary interest of work due to the great quantities of old film material present in film archives. Video is a collection of successive frames. One film video contains huge number of frames. To remove scratches from videos manually is highly cumbersome and time consuming task. Hence some automatic or semi-automatic tools are highly required to detect such scratches. After scratches detection removal of those scratches is again an important and challenging task. Line scratch detection and restoration is challenging task. For instance, in some cases, the scratch is semi- transparent, so that some of the original image information is still accessible, whereas in others all the information of original image is removed. Also, scratches are not necessarily totally straight and vertical, and their shape may in fact vary from frame to frame. Finally, although scratches can often be static, they may also travel with any type of motion.

In our project we present a technique for scratches detection and removal of line scratches from video. Firstly for scratch detection we proposed vertical scratches detection algorithm based on edge detection. The proposed algorithm first uses the operator which has the largest response to the vertical edge in Sobel operator to detect edges, and then uses canny operator to detect edges further. Third, we detect vertical lines in the image through probabilistic Hough transform. Finally, we obtain the true locations of the vertical lines scratches through morphology and width constraints. We collect the detected scratch pixel and identifies the shape, using which we remove false detections of scratches in frame. To remove scratches from old video we proposed a novel non-linear interpolation method, both of information in and between the frames are used to create the interpolation polynomial. Since the non-linear scheme is more suitable for human visualization and interpolation method is always speedy, our method shows its usefulness in video inpainting. We make a scratch detection before the interpolation.

II. BACKGROUND OF PRESENT WORK

Since more than a hundred years techniques for recording and playing of motion images exist. The early movies were recorded on an easy flammable and destructible material. But also recent movies have damages through declining, chemical changes and contact with mechanical projection tools. From the cultural and historical point of view the restoration of the film material is an important task. Due to the huge data volume the manual restoration is time and cost intensive. Therefore unsupervised video processing methods for removal of frequently occurring defects like scratches and blotches have to be developed. We define a defect as a miss of the original information in a particular range of the video sequence. The disturbance is usually a short, abrupt alteration of only a few pixels in the frame/sequence or some specific structure, like vertical scratches, which result from mechanical failures. The defects can be separated accordingly to their shape, colour, size and length of time into:

- Scratches or streaks
- Blotches or dirt and sparkle,
- Loss of whole frames or sections within a frame and
- Global changes of the colour, brightness and contrast

A part of the film surface is missing resulting in a scratch. But also other causes can lead to such defects for example not careful handling with the film tape (reparation, archiving). Figure 2.1 shows some examples for scratches. In Figure 2.1(a) a weak distinct scratch near the right border can be seen. Much more obvious scratches are visible in Figure 2.1(c) (in the middle) and in Figure 2.1(d) (on the right hand side). In the last case many small vertical scratches are distributed over the whole image. An unusual kind of scratch is shown in Figure 2.1(b). The diagonal scratch begins in the left half of the image and runs over the image.

a)

b)

c)

d)

Figure 2.1: Examples for scratches: (a) weak distinct scratch near the right border; (b) diagonal scratch; (c) dark scratch in the middle of the image; (d) dark scratch on the right hand side

Line scratches in motion films are lines of noticeably darker or lighter intensity than the nearby pixels. These scratches are exactly or close to vertical. They can be visible on all or much of the height of the image, in which two cases first one is primary scratches define the scratches which are visible along more than 90% of the height of the image. Another one is Secondary scratches defines all scratches which do not cover at least 90% of the height of the image (non-primary scratches). Scratches are defined in the literature as persistent (meaning they last more than one frame), or non-impulsive. They are also described (empirically) in papers by Bruni [4,5] as having a width of 3-10 pixels. This is not a definite criterion, however, and Kokaram [1,2] considers the width as a variable to be estimated with a probability density function. Scratches are also supposed in several papers [1, 2, 4, 5, 6] to have "side-lobes" (lines on the left and right of the central part of the scratch which are less extreme).From an image processing point of view we can describe a vertical scratch as a few columns-wide portion of the image where the intensity information is completely lost.

Line scratches can be originated by a number of causes, and typically they are vertical lines across the frame. It is important to know what the physical reasons of line scratches are. Such knowledge helps to understand why these scratches have certain characteristics and is the basis for any line scratch model. Line scratches are caused by abrasions against the film. This can happen in several cases: A particle or a mechanical part rubbing against the film during projection or transportation .A particle caught between the films roll itself, if the film is rewound carelessly. The scratches start at the top of the image and run vertically over the entire image. They are typically darker than the rest of the image and can be found on at least several subsequent frames. Thus the scratches are viewable over a certain time interval, sometimes over a couple of seconds. During this time they can remain on the same position or move to the left or right side. The analysis of original and noisy TV material has shown that the typical width of a scratch is approximately 1% of the horizontal image resolution. Regions with different intensity than the local neighborhood can be observed on both sides of the destroyed pixel lines: dark scratches are surrounded by bright regions and white scratches by dark regions. A viewer notices scratches quickly, because some knowledge and context information about the structure and the objects on the scene is available. Moreover the attention of the observer is paying attention by local discontinuities on the temporal axis.

ISSN: 2454-4248
69 – 78

Although such structures are clear emphasizing, the color change is not typical for an edge. The scratch area is wide and no abrupt alterations of the color run exist. These smooth changes are usually not recognized by edge detectors, for example gradient operators with standard thresholds. A further class consists of scratches which are caused by short contacts among some objects and the film role. These scratches are viewable over a very short period (less than a couple of frames) and occur like a disturbing flash of lightning. They have different shapes and their position is limited to a particular image area, thus it is more complicated to distinguish between scratches and real image elements than in the case of the vertical scratches.

III. RELATED WORK

The period of research on detection of vertical scratch line is more than fifteen years. It is come out many detection methods during this period. The last century in the late nineties, Kokaram [1,2] first started to study systematically line scratches, and presented a mathematical model of vertical scratches and distorted image. This method can detect vertical scratches single frame image. After the preprocessing through median filtering, Hough transform, Bayesian algorithm, it can decide really scratch location in image frame. However, the algorithm requirements to adjust the threshold value for different images. It cannot differentiate between true vertical scratches and vertical objects in image. On the other hand, finishing time of this algorithm is very long, which is not applied to real-time processing and cannot handle a large number of consecutive video frames. In view of this , 1. Joyeux [3] proposed a Space- Time Decision Method to solve this complexity, but the algorithm was not totally linear scratches services and was not largely adopted. V. Bruni [41 believed that the mathematical model of vertical scratches is not purely additive, but the scratch region has been lost. In 2004, he enhanced the scratch mathematical model of Kokaram, and proposed a new model of vertical scratches. The model works well to a single image, but it is time consuming, also it cannot differentiate between true vertical scratches and vertical objects in image. In 2008,V.Bruni [5] tried to detect scratches in the color image. Before detection algorithm, they used Shannon sampling theorem to sample vertically image in order to extract red component, and they applied the scratch detection method in gray scale images to detect vertical scratches. Although this method sometimes presented fake alarm, it provided a direction to vertical scratches detection and restoration work in color images.

There are two well known digital video inpainting methods, texture synthesis and nonlinear PDE method. PDE usually used to restore small scratches in a global way but it always introduces blurred images and time-consuming. Bertalmio is the first person to bring in PDE in video inpainting. In his paper[10], a image repair model BSCB was proposed which repaired the damaged part along the direction of isophotes based on an isotropic diffusion method Chan put forward a new TV(total variation) model[11] to process the small damaged region automatically. But this method can't keep connectivity principle prone to ladder effect, at the same time; it is also time-consuming for the sake of iteration. Chan gave a new CCD [12] method to improve the earlier algorithm. CCD model based on the Curvature-Driven Diffusion, the repairing process carried out along the lines perpendicular to the direction of isophotes, the intensity of the diffusion determined by the curvature and gradient magnitude, however, this method is local and also time consuming. Shao [13] proposed an adaptive way to inpaint the video based on TV model, this algorithm shortened time for picking parameters in interation process and enhanced the TV algorithm. Zhao [14] put forward a fast inpainting algorithm based on TV model, magnitude and orientation of gradient were used to synthesize the pixels information for damaged part. This algorithm also took a approach of sort. It determined sequence of the repairing process according to the gradient information of neighbor pixels. Vijaykumar V.R [15] proposed an self adaption median filter, this filter can regulate its size based on the region of spots during the entire process. All above algorithms can be classified to Local method and global method. Local ones cannot reflect the whole feature of the images; global ones are time-consuming and always result in vague images. Moreover these methods are spatial ones which did not consider the time coherency of the video.

IV. PROPOSED METHODOLOGY

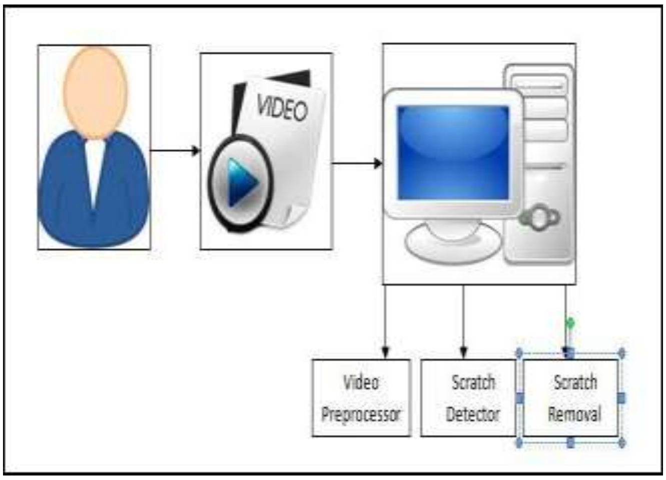

Figure 4.1: System Architecure

In our proposed system, we have collected one raw black and white video with scratches from user. We filter that is raw video frame and produce refined frame with removal of maximum scratches. For video processing we have separated video in k number of frames and each frame is processed individually to spot scratches. For scratch identification we have used vertical scratch detection algorithm base on edge detection technique. To remove false detected scratches we compare scratches with adjacent frames using temporal technique. After scratch detection we removed those scratches using a novel non-linear interpolation method.

The proposed methodology contains two main stages where first is to detect the noise or the scratches in the frame and second is to remove the scratches.

ISSN: 2454-4248
69 – 78

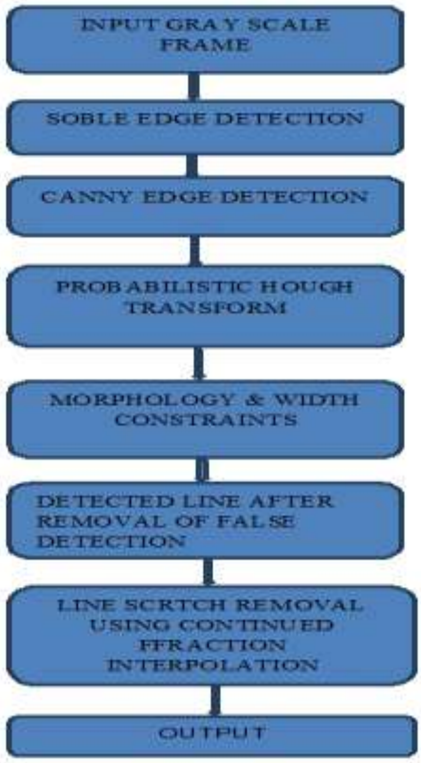

Figure 4.2: Proposed process flow of detection and removal of line scratches from video frame.

A. DETECTION OF THE SCRATCHES

1) Image Types

Digital image is represented as a set of picture element called pixel. They are organized as two dimensional arrays. Digital images can be classified according to the number of bits per pixel since the number of distinct colors of a digital image depends on number bits per pixel (bpp). There are three common types of images

a) Binary image: In this type, one bit is allocated for each pixel. The value of a bit is represented as either 1 or 0. Each pixels of a binary image should be represented as any one of two colors (black and white). Binary image is also called as bi-level image

b) Grayscale image: It is also known as an intensity, gray scale, or gray level image. Array of class uint8, uint16, int16, single, or double whose pixel values specify intensity values. For single or double arrays, values range from [0, 1]. For uint8, values range from [0,255]. For uint16, values range from [0, 65535]. For int16, values range from [-32768, 32767].

c) RGB or True color image: It is also known as an RGB image. A true color image is an image in which each pixel is specified by three values one each for the red, blue, and green components of the pixel scalar. M by-n-by-3 array of class uint8, uint16, single, or double whose pixel values specify intensity values. For single or double arrays, values range from [0, 1]. For uint8, values range from [0, 255]. For uint16, values range from [0, 65535].

1.1 Conversion of RGB image to Gray image

We know that the Image is made by number of pixels and dissimilar major parameters like colour and monochrome (sometimes also known as black & white image or property). Image is processed and executed by an image processing techniques. So image processing is the most important part of signal processing. Gray scale conversion is also a vital part of image processing. RGB or colour information has a 3 dimensional property which makes signal processing so much bulky and heavy to remove this drawbacks gray scale translation is necessary. Gray scale images are the images where colour information is missing and all colour information is converted into gray scale format.

Grayscale images are distinct from one-bit bi-tonal black-and-white images, which in the context of computer imaging are images with only the two colors, black, and white .Grayscale images have many shades of gray in between. Matlab supports large amount of image formats i.e. jpg, tif, bitmap, png, gif etc.

i) Transforming input image f(i,j) into gray scale image I.

We use the formula as follows:

$$I = \max(R, G, B)$$

R, G, B is the red, green, and blue color channels of input image f (i ,j), respectively.

a) Original Image

b) Gray Scale Image

Figure 4.3: Conversion of RGB to Gray scale image

2) Introduction to fundamentals of edge detection

Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are sudden changes in pixel intensity which characterize boundaries of objects in a scene. Classical methods of edge

ISSN: 2454-4248
69 – 78

detection involve convolving the image with an operator (a 2-D filter), which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. This is an extremely large number of edge detection operators existing; each designed to be sensitive to certain types of edges.

In the ideal case, the result of applying an edge detector to an image may lead to a set of connected curves that show the boundaries of objects, the boundaries of surface markings as fine curves that correspond to discontinuities in surface orientation. Thus, applying an edge detector to an image may considerably reduce the amount of data to be processed and may therefore filter out information that may be regarded as less relevant, while preserving the important structural properties of an image. If the edge detection step is doing well, the subsequent task of interpreting the information contents in the original image may therefore be substantially simplified. Regrettably, however, it is not for all time possible to obtain such ideal edges from real life images of moderate complexity. Edges extracted from non-trivial images are often hampered by fragmentation, meaning that the edge curves are not connected, missing edge segments as well as false edges not corresponding to interesting phenomena in the image thus complicating the subsequent task of interpreting the image data.

2.1 Different type edge detector

Edge detection makes use of different operators to detect changes in the gradients of the gray levels. It is divided into two main classes

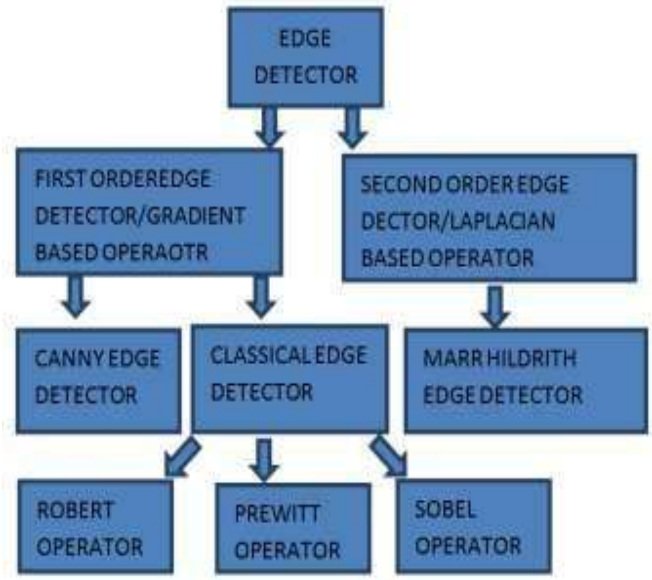

Figure 4.4: Types of Edge Detector

Edge detection is a terminology in image processing and computer vision, particularly in the areas of feature detection and feature extraction, to refer to algorithms which aim at identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities. They are many types but I am using only two types of edge detector that is

- Sobel Edge Detector - 3×3 gradient edge detector
- Canny Edge Detector - non-maximal suppression of local gradient magnitude

2.2 Soble edge detection

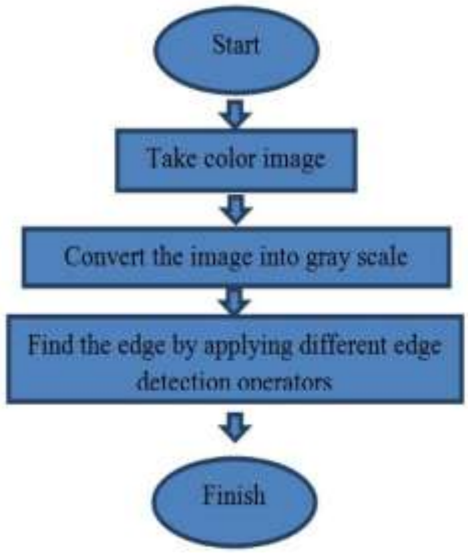

Figure 4.5: Flow chart for edge detection

The Sobel operator do a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial frequency that correspond to edges. Typically it is apply to find the approximate absolute gradient magnitude at each point in an input grayscale image.

2.2.1 How It Works

In theory at least, the operator consists of a pair of 3×3 convolution kernels as shown in Figure shown one kernel is simply the other rotated b y 90°.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Figure 4.6: Sobel convolution kernels

These kernels are designed to take action maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these *Gx* and *Gy*). These can then be joint together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given by:

 ISSN: 2454-4248
69 – 78

$$|G| = \sqrt{Gx^2 + Gy^2}$$

Typically, an approximate magnitude is computed using

$$|G| = |Gx| + |Gy|$$

which is much faster to compute.The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by:

$$\theta = \tan^{-1}(Gy/Gx)$$

In this case, orientation 0 is taken to mean that the direction of maximum contrast from black to white runs from left to right on the image, and other angles are calculated anti-clockwise from this. Often, this absolute magnitude is the only output the user in the two components of the gradient are conveniently computed and added in a single pass over the input image by the pseudo-convolution operator shown in Figure shown below.

P_1	P_2	P_3
P_4	P_5	P_6
P_7	P_8	P_9

Figure 4.7: Pseudo-convolution kernels used to quickly compute approximate gradient magnitude

Using this kernel the approximate magnitude is given by:

$$|G| = |(P_1+2\times P_2+P_3)-(P_7+2\times P_8+P_9)| + |(P_3+2\times P_6+P_9)-(P_1+2\times P_4+P_7)|$$

2.2.2 Sobel Explanation

The mask is slid over an area of the input image, changes that pixel's value and then shifts one pixel to the right and continues to the right until it reaches the end of a row. It then starts at the beginning of the next row. The example below shows the mask being slid over the top left portion of the input image represented by the green outline. The formula shows how a particular pixel in the output image would be calculated. The center of the mask is placed over the pixel you are manipulating in the image. And the I & J values are used to move the file pointer so you can multiply, for example, pixel (a22) by the corresponding mask value (m22). It is important to notice that pixels in the first and last rows, as well as the first and last columns cannot be manipulated by a 3x3 mask. This is because when placing the center of the mask over a pixel in the first row (for example), the mask will be outside the image boundaries.

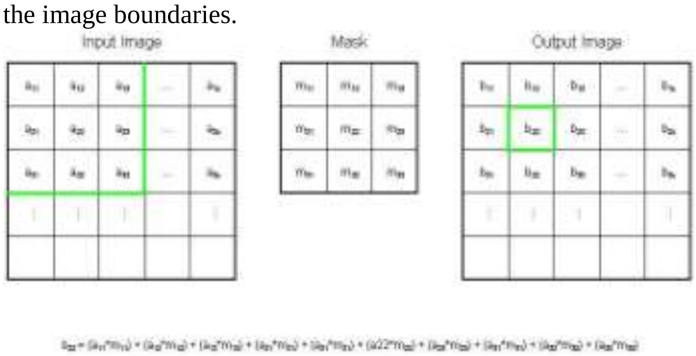

Figure 4.8: Sobel Explanation how to get output b_{22}

The GX mask highlights the edges in the horizontal direction while the GY mask highlights the edges in the vertical direction. After taking the magnitude of both, the resulting output detects edges in both directions.

2.3 Canny Edge Detector

The Canny edge detection algorithm is known to many as the optimal edge detector. Canny's purpose were to enhance the many edge detectors already out at the time he started his work. He was very successful in achieving his goal and his ideas and methods can be found in his paper, "A Computational Approach to Edge Detection". He followed a list of criteria to improve current methods of edge detection. The first and most obvious is low error rate. It is significant that edges occurring in images should not be missed and that there be no responses to non-edges. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a least. A third criterion is to have only one response to a single edge. This was implemented because the first two were not substantial enough to completely remove the possibility of multiple responses to an edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (non maximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a non edge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2.

3) Hough Transform

The Hough Transform is an algorithm presented by Paul Hough in 1962 for the finding of features of a particular shape like lines or circles in digitalized images. In its classical form it was limited to features that can be described in a parametric form. However, a generalized version of the algorithm exists which can also be applied to features with no simple analytic form, but it is very complex in terms of computation time.

Hough transform can be applied to numerous computer vision problems as most images contain feature boundaries which can be described by regular curves. The main benefit of the Hough transform technique is that it is tolerant of gaps in feature boundary descriptions and is relatively unaffected by image noise, dissimilar edge detectors. One important difference between the Hough transform and other approaches is resistance of the former to noise in the image and its tolerance towards holes in the boundary line.

In Hough transform, the points are linked by determining first if they lie on the curve of specified shape. Unlike the local analysis method, where given n points of an image are taken into consideration. Suppose we want to find the subset of these points that lie on the straight lines. One possible solution is to find all the lines determine by every pair of points and then find all subsets of points that are close to particular lines. The problem with this procedure is that it involves finding n (n-1)/2 ~ n^2 lines and then performing (n) (n (n-1)) ~ n^3 comparisons of every point to all lines. This approach is computationally prohibitive in all but in most the trivial applications. The Hough transform is a method that, in theory, can be used to find features of any shape in an image. In practice it is only generally used for finding straight lines or circles. The computational complexity of the method grows rapidly with more complex shapes.

4) Morphological Operations

Morphology is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size. In a morphological operation, the value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbors. By choosing the size and shape of the neighborhood, we can construct a morphological operation that is sensitive to specific shapes in the input image.

4.1Theoretical considerations

The two principal morphological operations are *dilation* and *erosion*. Dilation allows objects to increase, thus potentially filling in small holes and connecting disjoint objects. Erosion shrinks objects by etching away (eroding) their boundaries. These operations can be customized for an application by the proper selection of the structuring element, which determines exactly how the objects will be dilated or eroded.

Notations:
black pixel: in grayscale values for a 8 bits/pixel indexed image its value will be 0
white pixel: in grayscale values for a 8 bits/pixel indexed image its value will be 255

5) Temporal Filtering

In a film, motion of an object is totally independent from the background scene. Any finding from moving object in scene leads to false detection. Temporal technique identifies false finding of scratches in a frame by comparing the frame objects with adjacent frames. For this we have used motion coherence technique. In this we find out whether scene has been moved during a trajectory's time span. For this, we have *horizontal* motion of the scene. We identify original positions of object as a trajectory set. This helps to identify motion in successive frames. Two segments *Q* and *R* belonging to this set which verify the following inequality:

$$|x(Q) - x(R)| \geq \tau m,$$

where τm is a motion threshold.

This corresponds to the absolute distance that the scene has moved between the frames **t**(Q) and **t**(R).

B. REMOVAL OF SCRATCHES

After the scratch detection, we find the position of scratch. Then proper pixels should be found to create interpolation formulation. For digital image restoration, a neighbor pixel around the scratch contributes more than pixels far away. In the case of digital video inpainting, neighbor frames also contain significant information. The temporal correlation of the frames and randomness of the scratch make it possible to locate the missing data in the neighbor frames. In this paper, we not only consider the spatial continuity but also the time coherency. We first use the information of frames nearby. We choose four frames before and after the current frame, so there are eight correlated frames altogether. For every missing pixel within the scratch, we can find a pixel in the same location of the correlated frame. Then Linear interpolaton can be constructed using these pixels of eight frames.

Let F_c denote the current frame at t moment, then the four frames before it can be denoted as $F_{C-1}, F_{C-2}, F_{C-3}, F_{C-4}$ and frames after are $F_{C+1}, F_{C+2}, F_{C+3}, F_{C+4}$

Suppose F_c (x, y) is a pixel within scratch, then it can be estimated by Linear interpolation using

$F_{c-4}(x, y)$, $F_{c-3}(x, y)$ $F_{c+3}(x, y)$, $F_{c+4}(x, y)$

$$F_c(x,y) = F_{c-4}(x,y) + F_{c-3}(x,y)(t-t_0) + F_{c-2}(x,y)(t-t_1) + \cdots + F_{c-4}(x,y)(t-t_7)$$

Where t_i, i =0...7 are moments of interpolated frames.

Sometimes we will encounter the situation that a scratch appears a long time at the similar location. To this kind of circumstance, spatial information is taken into considered. Unlike some local interpolation method, global strategy is taken in this paper. Some algorithm may choose the neighbor pixels around the missing data to make the interpolation polynomial whereas we use all of the pixels in horizontal direction thus more global information can be used. However Recursive form makes the reconstruction process further efficient, procession will not be slow.

Suppose I is a m×n image, Let S (x , y) denote the pixel within scratch, P (i , y), i ≠x,i = 1…m are pixels in the same row. Inverse difference table can be constructed like below:

$$\begin{array}{llll}
x_0 & p(1,y) & & \\
x_1 & p(2,y) & p_1^1 & \\
x_2 & p(3,y) & p_2^1 & \\
. & . & . & \\
. & . & . & \\
x_m & p(m,y) & p_m^1 \;\ldots\ldots & p_m^m
\end{array}$$

We can construct the continued fraction through the following steps:

Suppose $D = \{(x_0, f_0), (x_1, f_1), \ , (x_n, f_n)\}$

(1) Calculate the inverse difference:

$$f_i^0 = f_i, i = 1,2,3\ldots.n$$

ISSN: 2454-4248
69 – 78

$$f_i^l = \frac{x_i - x_{l-1}}{f_i^{l-1} - f_{l-1}^{l-1}}$$
$$l = 1,2,3 \cdots n, i = l, l+1 \cdots n$$

We can construct an inverse difference table from the recursive formulation above:

$$\begin{array}{lllll} x_1 & f_1 & f_1^1 & & \\ x_2 & f_2 & f_2^1 & & \\ . & . & . & & \\ . & . & . & & \\ x_n & f_n & f_n^1 & \ldots\ldots & f_n^n \end{array}$$

(2) If $f_j^j = f_{j+1}^j, j \le n-2$ then we use difference instead of inverse difference

$$f_i^l = \frac{f_i^{l-1} - f_{l-1}^{l-1}}{x_i - x_{l-1}}$$
$$l = j+1 \ldots n, i = l, l+1 \ldots n$$

(3) Blending continued fraction is like below:

$$R(x) = \frac{P(x)}{Q(x)} = f_0 + \frac{x - x_0}{f_1^1} + \ldots + \frac{x - x_{j-2}}{f_{j-1}^{j-1}} + \frac{x - x_{j-1}}{\tilde{Q}(x)}$$

Where Q(x) has a Newton interpolation form

$$\tilde{Q}(x) = f_j^j + f_{j+1}^{j+1}(x - x_j) + f_{j+2}^{j+2}(x - x_{j+1}) + \cdots + f_n^n(x - x_{n-1})$$

V. RESULTS

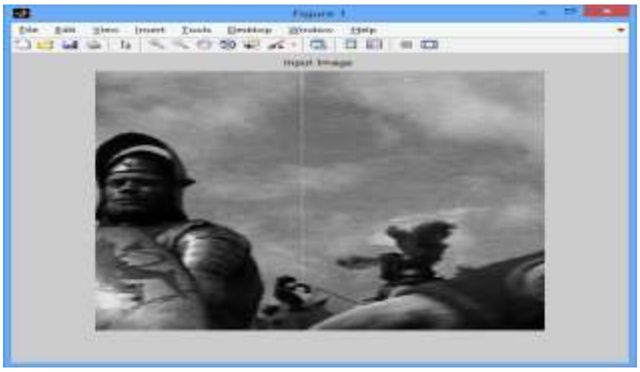

Figure 5.1: Input image

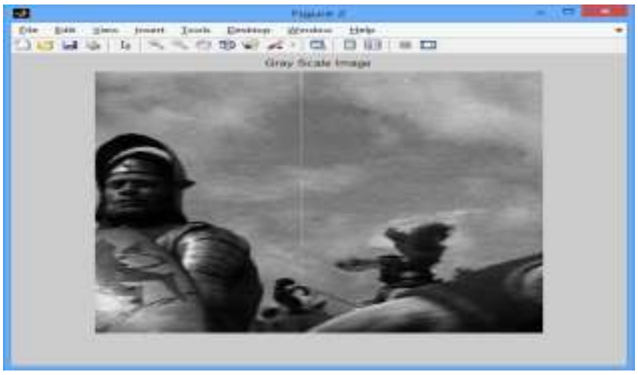

Figure 5.2: Gray scale image.

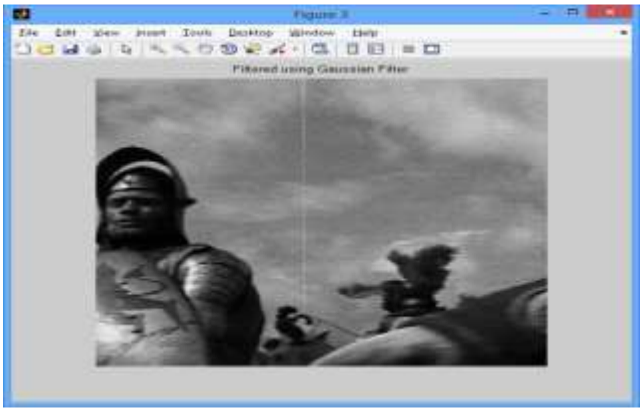

Figure 5.3: Gray scale image filtered using Gaussian filter.

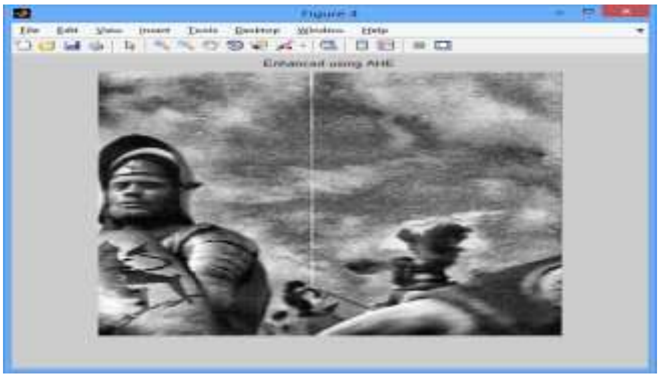

Figure 5.4: Enhanced using AHE.

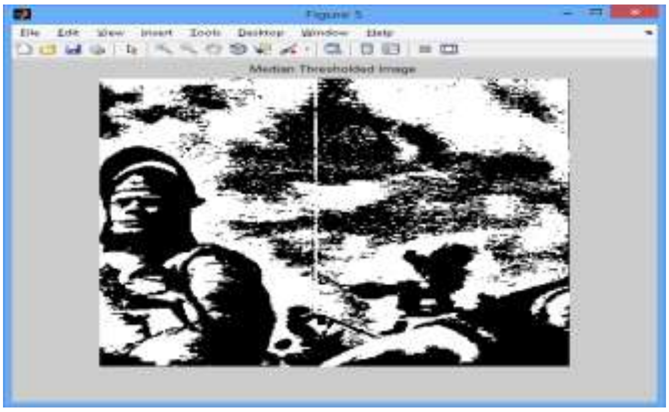

Figure 5.5: Median Threshold image.

ISSN: 2454-4248
69 – 78

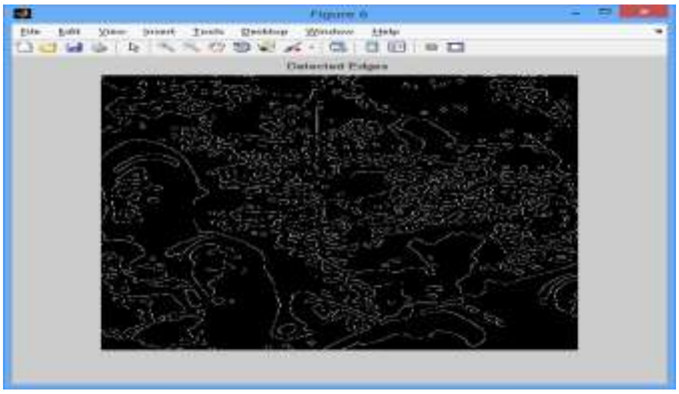

Figure 5.6: Detected Edges.

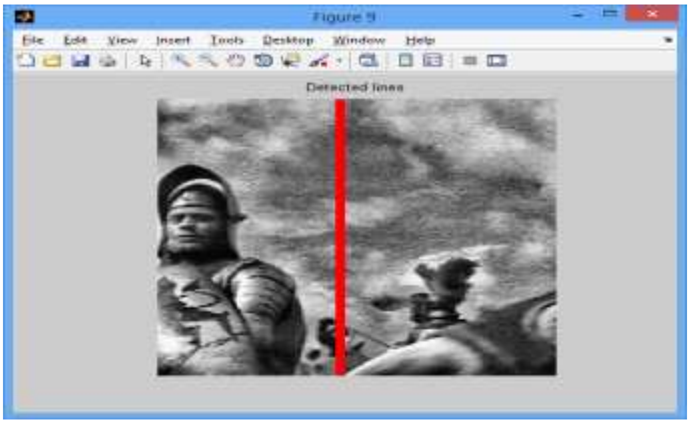

Figure 5.7: Detected scratch line.

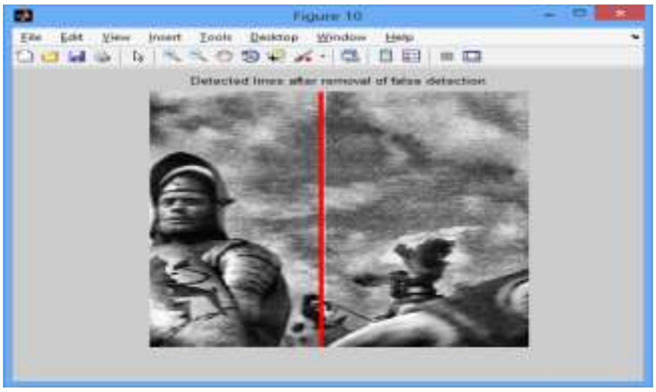

Figure 5.8: Detected scratch line after removal of false detection.

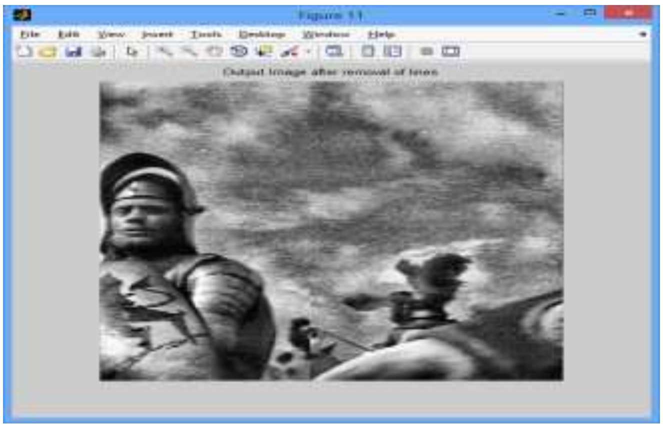

Figure 5.9: Detected scratch line after removal of false detection.

VI. CONCLUSION AND FUTURE WORK

In our system we have presented a technique for scratches detection and removal of line scratches from video. For scratch detection we proposed a line scratches detection algorithm based on edge detection. Edge detection is an important pre-processing step in image analysis. The proposed algorithm first uses the operator which has the largest response to the vertical edge in Sobel operator to detect edges, and then by using canny operator we detect edges further. Then probabilistic Hough transform useful for detection of vertical lines in the image. Finally, morphology and width constraints used to obtain the true locations of the vertical lines scratches. We collect the detected scratch pixel and identifies the shape, using which we remove false detections. The novel non-linear interpolation technique helps to remove scratches in the frame.

In future we will work on color videos for scratch detection and removal.

REFERENCE

[1] A.C.Kokaram, "Detection and removal of line scratches in degraded motion picture sequences," Signal Processing VIII, Vol. I , No.9, pp.5-8, 1996.

[2] A.C. Kokaram, "Motion Picture Restoration Digital Algorithms for Artifact Suppression in Degraded Motion Picture Film and Video," Publisher: Springer-Verlag, Berlin, 1998, pp.5-50.

[3] L. Joyeux, S. Boukir, and B. Besserer, "Film line scratch removal using kalman filtering and bayesian restoration," Proc.5th : IEEE Workshop on Applications of Computer Vision, IEEE Computer Society,2000, pp.8-13.

[4] V. Bruni, D. Vitulano, "A generalized model for scratch detection," IEEE Transactions on Image Processing, 2004, 13(1): 44-50.

[5] V. Bruni, P. Ferrara, and D. Vitulano, "Removal of Color Scratches from Old Motion Picture Films Exploiting Human Perception," EURASIP Journal on Advances in Signal Processing, 2008,16: 1-9.

[6] Canny, "A Computational approach to edge detection," IEEE Trans on Pattern : Analysis and Machine Intelligence, 1986, 8(6): 679-698.

[7] L.Fan, "Edge detection algorithm based on the Canny operator," Knowledge Economy, 2009(16): 113-114.

[8] R. Liu, S. Yu, Open CV Basics Tutorial, Beijing: Beijing University of Aeronautics and Astronautics Press, 2007, pp. 305-319.

[9] Q. Ruan, Y. Ruan transl., Digital Image Processing, 2nded, Beigjing:Publishing House of Electronic Industry, 2003, PP. 2-20

[10] Bertalmio M, Sapiro G, Ballester C etal. “Image Inpainting,”. Proc.27th annual conference on computer graphics and interactive techniques (SIGGRAPH’00), ACM Press,Jul. 2000, pp. 417-424, doi: 10.11 45 /344 779.344972.

[11] Chan T F, Shen J,“Mathematical Models of Local Non-texture Inpaintings,”. SIAM Journal on Applied Mathematics, Vol. 62, No. 3,Feb. 2002 ,pp. 1019 -104 3 ,doi: 10.1137/S0036139900368844

[12] Chan T F, Shen J. “Non-Texture inpainting by Curvature Driven Diffusions,”. Journal of Visual Communication and

Image Representation, Vol 12,No.4, Mar. 2002,pp.436-449,doi: 10.1006/jvci.2001.0487

[13] Xiaowei Shao,Zhenkai Liu,Bi Song, "An adaptive method of image inpainting based on TV model," Journal of Circuits and Systems,No. 2,Apr. 2004,pp. 113-117.

[14] Weiyan Zhao,LinXiang Li, " A fast method of image inpainting based on TV model,". Microelectronics & Computer, No. 6, Jun. 2009,pp. 253-260.

[15] Vijaykumar VR, Jothibasu P, "Decision based Adaptive median filter to remove blotches, scratches, streaks,stripes and impluse noise in images,". Proc.17th IEEE InternationalConference on Image Processing (ICIP 2010), IEEE Press, Sept. 2010,pp. 117-120, doi: 10.1109/ICIP.2010.5651915.

[16] NieShiliang, Zhang Hongying, Zhang Liping, Fan Yang, Vincent Brost, "Vertical Scratches Detection based on Edge Detection for Old Film", 2nd International Conference on Industrial and Information Systems,2010.

[17] Xing Huo, Jieqing Tan, "A Novel Non-linear Method of Automatic Video Scratch Removal",,Fourth International Conference on Digital Home,2012

[18] Alasdair Newson, Andrés Almansa, Yann Gousseau, and Patrick Pérez, "Robust Automatic Line Scratch Detection in Films", IEEE Transaction on Image Processing. Vol. 23, No. 3, March 2014

[19] Komal S. Mahajan, Prof. M. B. Vaidya "Review: Line Scratch Detection in Motion Films", International Journal of Emerging Technology and Advanced Engineering. Volume 3, Issue 3, March 2013