Detection and Correction of Mistracking in Digitalized Analog Video

Filippo Stanco, Dario Allegra, and Filippo Luigi Maria Milotta

Dipartimento di Matematica e Informatica, University of Catania,
viale A. Doria no. 6, 95125 Catania
fstanco@dmi.unict.it, {darioalltalk,fmilotta}@gmail.com

Abstract. Nowadays video technology is basically digital, but in the last half century the most diffused devices have been analog magnetic tapes. Since this is an old storing technique, it is necessary to convert these data in digital form. Moreover, analog videos may be affected by particular defects, called *drops*. Despite there are many hardware to perform the digitalization, few implement the correction of drops. In this paper, the drop also known as "Mistracking" is focused. A method to detect and correct this artifact is developed.

Keywords: Mistracking, Tracking Error, Analog Video, Drop Out.

1 Introduction

Image processing techniques have been successfully applied in many fields related to cultural heritage [1–3] and can be also exploited in video storage and restoration as part of the "Archeomatica Project" [4]. Since 1950s videotapes are used as data storage, and nowadays there are huge archives which contain considerable information. Today the magnetic tapes are obsolete and replaced by digital devices. Hence, in order to preserve these data it is necessary to convert them in digital form. For example, TV networks have a lot of news report stored on magnetic tapes only. In particular, Italian public broadcasting company RAI (Radiotelevisione Italiana [5]), our partner in this research, needs to solve this problem by software.

Videotape technology is based on electromagnetism. In particular using an electric field to align magnetic domains on the material of tape so that they remain aligned, even in absence of it. In this way, thanks to electromagnetic induction principles, it is possible to retrieve information stored on the tape through the magnetic fields of domains. Usually, reader devices consist in mechanical heads and reels for winding the tape. Data information is not recorded longitudinally and it does not fill all tape surface, so a synchronization signal, usually called CTL (Control Track Longitudinal), is required. A wrong synchronization or general mechanical problems could introduce some defects during reading or writing video flow. The Fig. 2 shows some examples of these analog drops: Dropout, Head Clog, Tracking Error or Mistracking, Skew Error, Comets,

Sync Loss, Tape Crease, Head Switching Noise, Dot Crawl, AC Beat. In this paper, a method to correct the Mistracking is presented. Mistracking error is one of the most common defect originated by CTL, and it is independent from video coding format (PAL or NTSC) [11]. In particular, we propose a method to detect and correct this artifact (Fig. 1).

In literature, at the best of our knowledge, there are not work related on Mistracking. However, there are many papers about film defects. A. Kokaram et al. [7, 8] focused just in Dirt, Lines, Shake, Flicker or Noise. Also Buisson et al. [9] and Rosenthaler et al. [10] develop techniques to solve video restoration problems. Unfortunately, none of these algorithms can be applied to Mistracking.

The paper is organized as follow. Section 2 reports an overview on Mistracking. Then, Section 3 describes the detection phase and Section 4 the correction phase. Finally, experimental results are shown in Section 5. Conclusions ends the paper.

2   An Overview on Mistracking

Mistracking, also called Tracking Error, consists in random rows of noise in the frame, usually white or black (Figs. 6(a), 5(a)). It is caused by misreading of video tracks recorded on a tape for an error in synchronization signal (CTL). In this way video heads could read a wrong area of tape, which may contains no data or inconsistent information, like a previous subscribed record. When interlaced mode is used, each frame consists of two distinct fields (odd rows and even rows), and Mistracking could affect just one field or both of them (Fig. 4). In some cases is possible to solve Mistracking modifying synchronization analogically by device components, but if this is allowed by the device and CTL skew is uniform along all the track. Otherwise, if the video is already converted in digital form, some Image Processing technique is required to solve this problem.

3   Detection Phase

The Mistracking consists in a quick variation of luminance between close horizontal lines in a rectangular area. This property suggests a connection with edge detection filters, which may be implemented through convolution masks. A known example is Laplacian kernel, which allow to detect vertical and horizontal edges of the image [6]. Effectively if we use this approach, the result isn't bad. In fact, the damaged area, is highlighted exactly such as an edge, but unfortunately it can't be distinguished by other regular edges (Fig. 3(b)). This problem is slightly attenuated if we use an edge detection filter for horizontal edges only, as Prewitt or Sobel, but not enough to distinguish the tracking line by other edges. In some case, mainly in images with very few details, this approach could show satisfactory results, because the edge filters would detect less edges with respect to a typical detailed image, allowing the distinction. However, it is not a good solution. To solve this problem we propose a new filter, with kernel:

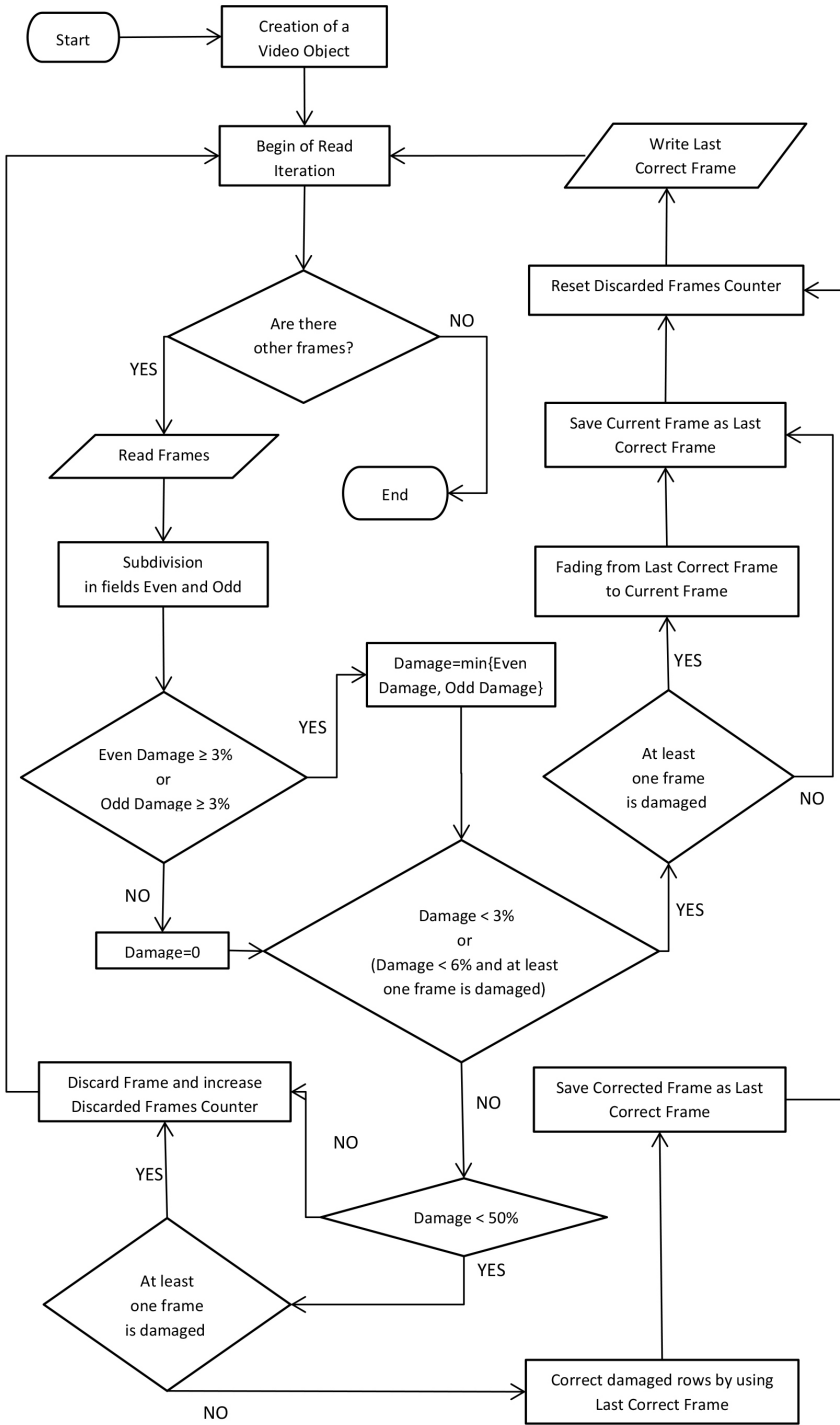$$\begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \end{bmatrix} /6 \tag{1}$$

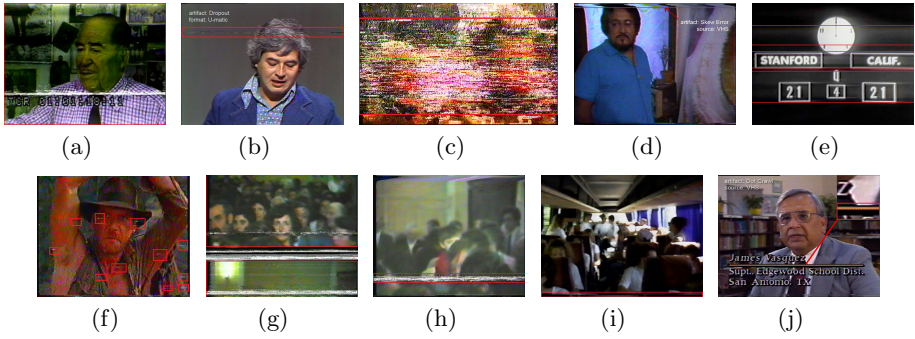**Fig. 1.** Flow Chart of the proposed algorithm

Fig. 2. Some examples of Analog Video Artifacts. (a) Head Clog: black or dark frame due to bad contact between heads and tape; (b) Drop Out: white and/or black lines of missing information in the picture; (c) Mistracking: horizontal lines of random noise; (d) Skew Error: frame hooked to the right or to the left; (e) AC Beat: bar of milky rolling luminance; (f) Comets: white or black dots with tail on the frame; (g) Sync Loss: loss of vertical hold during switching interval; (h) Tape Crease: horizontal rolling band of noise; (i) Head Switching Noise: loss of information in the lower border of the frame; (l) Dot Crawl: colored dots on horizontal borders.

This kernel, as well as detects horizontal edges only, sharpens above all the horizontal lines close together with marked luminance difference. Now, the filtered image consists in a grey scale frame with a great pixel value in the damaged area and a low value in other areas (Fig. 3(c)). The next step consists to find an index to describe the damage level. Our idea takes into account rectangular area whose length is the same as that of the image. The height depends on other criteria. Although this solution is definitely better than the previous proposals, it could fail when the image contains pattern similar at the tracking lines.
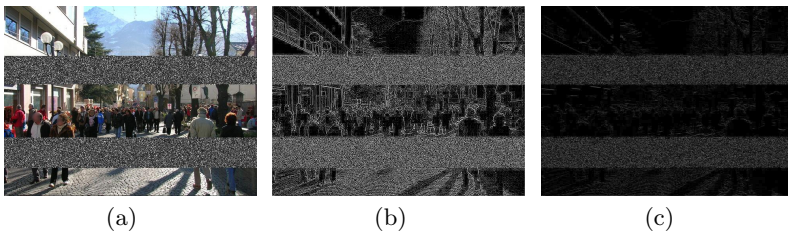

Fig. 3. (a) Input image; (b) Frame filtered with Laplacian kernel. (c) Frame filtered with the proposed kernel. The new kernel selects the corrupted lines but less details of the image. The figures (b) and (c) are gamma-modified to permit to see the not zero values.

3.1   Processing Filtered Image

Once we gain the filtered frame we have to establish a principle to determine which part of this one must be considered damaged or not. Due the convolution, now there are higher luminance values where an artifact appears in the new image. Since these areas extend horizontally from one side to the other of frame, is easy to see that the sum of luminance values of one or more damaged rows will be higher. Try to locate a damaged line using only the values of its pixels can be unreliable, but since the drop usually occurs on several consecutive rows, should also be considered for each row few lines adjacent to it, for a better estimation. We decided to analyze for each row the two upper and two lower, that is the same to use the values of the rectangle for each row r:

$$\Omega(r) = \{f(x,y)|1 \leqslant x \leqslant c; r - 2 \leqslant y \leqslant r + 2\} \tag{2}$$

where c is the number of all columns of the image. For each row r, the Equation (2) defines a set of luminance values where the average is the index related to row r.

3.2   Application Threshold

Indexes show the level of noise in each row so we have to decide which can be considered a correct line or not. The values obtained experimentally are always between 0 and 10 (theoretical values are between 0 and 64). Areas affected by other types of noise or containing details not have a value equal to 0, but in any case the value is still low, so is obtained experimentally that a good threshold value is between 1.5 and 3.5, according to the quality of the image. With the use of a low threshold we will have a most sensitive detection, on the other hand will be higher the chance of getting a false positive. A higher value would cause a failure detection of the drop.
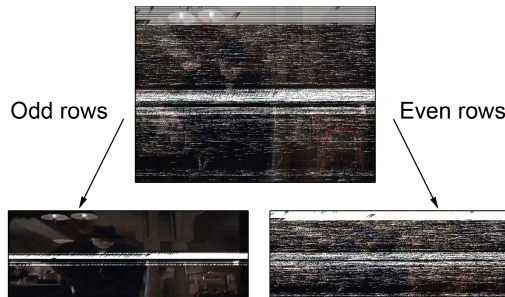

Odd rows   Even rows

Fig. 4. Interlaced frame splitted in its two fields

4   Correction Phase

After the tracking lines have been detected, some property of video is used to allows to correct this problem (Fig. 1). Two main characteristics are considered:

- Two adjacent frames are very similar each other. This doesn't happen when there is a "scene transition".
- Each frame consists in two fields: pair rows and odd rows (height is half of the entire frame, Fig. 4).

There are two coding modes: progressive or interlaced. When interlacement is used each frame consists of two fields, stored in two distinct tracks. Devices read the tracks individually but the frame rate is high enough that a unique frame is perceived. In progressive mode, each frame is not splitted in two fields but there is just a single track.

Our solution uses the similar information in adjacent frames and in the most genuine field. The original video could be interlaced, while after the conversion, a progressive video is obtained. First we split each frame in correspondent two fields again (Fig. 4), and use it as input of detection algorithm. Basically, to repair damaged frames, the single tracking lines or the entire frame/field can be replaced. Now, there are three different cases:

1. Detection of a strongly damaged frame;
2. Detection of a slightly damaged frame or half-frame (field);
3. Detection of a correct half-frame.

We fix that when a frame has over 50% of tracking lines in both fields (even and odd), is "strongly damaged"; when at least one half-frame has between 3% and 50% of tracking lines, is "slightly damaged"; otherwise the frame is "correct". Moreover, a discarded frames counter is used for correction in case 3. When discarded frames counter has a value higher than 0 is not a good choice to correct the single lines of a "slightly damaged" frame because the last "correct" frame could be temporally incoherent with the current scene. In particular:

Case 1. This case happens when we detect at least 50% of tracking lines in the even field, and at least 50% of tracking lines in the odd field. In this condition is very difficult to restore the damaged area, because we have very few information. For this, the best solution is to remove the entire frame. We will explain how replaces this damaged frame in the third case. Finally, the counter for discarded frames is increased.

Case 2. When at least one of two fields has a number of tracking lines between 3% and 50%, it's possible try to correct it. First, the field with the lowest damage is chosen and restored. If the counter of discarded frames is 0, the damaged lines of chosen field is replaced with correspondent genuine lines in the last correct field. Finally, the new field is resized and becomes the "the last" corrected frame. Else, if the counter of discarded frames isn't 0, to replace the single lines isn't a good solution, because the last corrected frame could be temporally too far by the field under consideration. The result would be a bad image. To solve this,
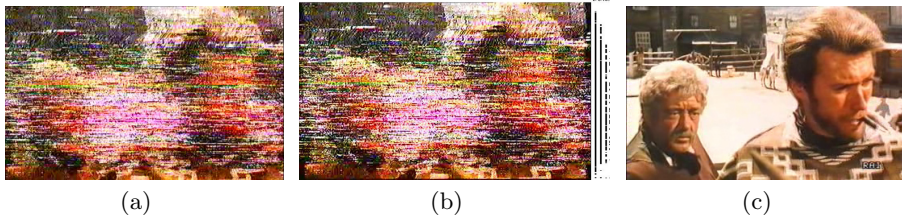
Fig. 5. Correction of Mistracking of frame called "Clint". (a) Original; (b) Detection on a non blurred frame, with several threshold values on the right; black lines under each value show which defected rows have been detected; (c) Restored frame using threshold equal to 1.5.

we do a further consideration: if the number of tracking lines is over 6%, we fall back in the case 1, else go to the case 3.

Case 3. In the last case if the percentage of at least one field is lower than 3% or is lower than 6% and the consecutive damaged frame counter is higher than 0, then we consider this field as a correct field. First of all we have to consider this frame (or field) as the new "correct frame" (or "correct field"). Then if the consecutive damaged frame counter is equal to d we have to correct previously discarded frame. To avoid an abrupt replacement of all these frames with the current correct frame, and the consequent effect of *freeze* frame, we use a fading transition that weights the previous correct frame (start) and the current one (end), based on the temporal position of appearance. The fading is described by the following process:

Let m be the consecutive damaged frame counter, a_i be the generic frame with $i = 0, 1, \ldots, m + 1$ and $k = 1, \ldots, m$, where "a_0" is the previous correct frame and a_{m+1} is the current correct frame; so each a_k frame of fading is obtained by:

$$a_k = a_0 \left(1 - \frac{k}{m+1} \right) + a_{m+1} \left(\frac{k}{m+1} \right) \tag{3}$$

Instead if the counter is equal to 0, than we don't make any correction operation. The choice of considering a frame as a "correct frame" even with a damage up to 6% due to the fact that if we find a slightly damaged frame after at least one strongly damaged one, then the slightly damaged frame will be discarded with a greater margin of error. In this way we avoid long sequence of fading, as seen in case 2. We must keep in mind that we can consider a "correct frame" or a "correct field". During the correction phase we have to compare the size of the damaged frame with the correct frame or correct field, so if these are different we have to resize the smallest vertically, or if these are equal we can correct them immediately, and only at the end if they were fields, resize the result to the size of a frame.
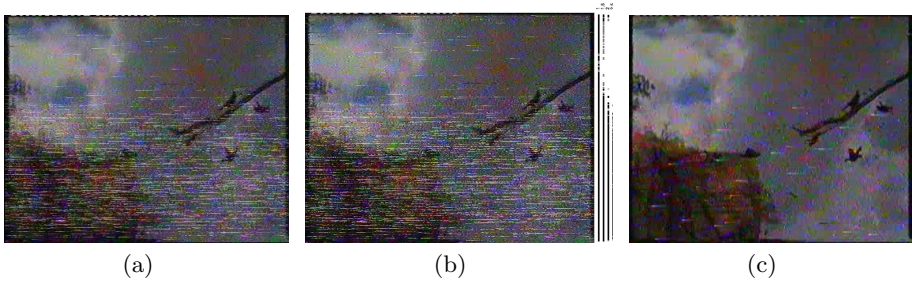
(a)   (b)   (c)

Fig. 6. Correction of Mistracking of frame called "Indy". (a) Original; (b) Detection on a blurred frame, with several threshold values on the right; black lines under each value show which defected rows have been detected; (c) Restored frame using threshold equal to 2.

5   Experimental Results

The proposed algorithm is tested on several videos. In some cases the Mistracking is properly generated to know percentage of total defect in each frame, in this way is possible to verify the correctness of detection algorithm.

From the Table 1 it can be seen that in almost all cases good results are obtained with a threshold value equal to 4. However, this threshold is not recommended because it does not take into account the correct rows adjacent to the damage. These rows often show a slight deterioration, so it is better to detect and correct them too. For this a threshold value such as 2.5, which also detects those lines, will provide better results. Very low threshold values, like 1.5, should be used in the case of very blurry images and poorly detailed, which are common in many analog video. In the Figs. 6(b) and 5(b) are reported two images taken from video affected by real Mistracking and are presented some results of the detection algorithm with threshold values from 1 to 2.5. The Fig. 6(c) and Fig. 5(c) shows results of the correction algorithm. Two restored videos can be found in [4].

5.1   Other Considerations about Correction Phase

The correction algorithm works well in most of the cases treated. However, there are situations in which the results do not coincide with those expected; the most significant cases are those in which:

- Many other types of drop are present in the same video;
- The content of the video is not homogeneous throughout the duration;
- There are undamaged horizontal lines identical to tracking lines;
- There are black bands in the sides of the frame.

When there are drops of other kinds, to detect mistracking can become difficult, and during the correction phase the areas affected by another defects could be used to make the replacement. Anyway, this usually enhances these drops. If

Table 1. Experimental results. T is the threshold values used in detection phase.

IMAGE	RESOLUTION	% DAMAGED	% T = 2.5	% T = 3.5	% T = 4
Geometric	416x330	**8.78**	**9.39**	7.87	7.87
		40.60	**40.90**	**40.30**	40.00
		60.30	**60.30**	60.00	60.00
Lawn	470x352	**13.92**	15.62	14.48	**13.92**
		41.19	42.61	42.04	**41.19**
		68.75	69.31	68.75	**68.75**
People	641x480	**18.95**	29.58	21.04	**19.79**
		32.29	42.91	32.70	**32.50**
		83.12	85.83	83.54	**83.33**
Text	400x320	**13.43**	16.25	13.75	**13.43**
		37.81	39.37	38.43	**37.81**
		61.25	63.43	**62.50**	**62.50**
Cartoon	720x576	**12.67**	14.40	**12.84**	12.50
		29.51	30.38	29.68	**29.51**
		62.15	**62.32**	61.97	61.97

the video quality is not uniform or there are very different scenes/contexts, it may happen that there is not a single threshold value suitable for the correct detection of each frame of video. In this case it is possible to split the video into several parts and correct them separately using different threshold. Among the other problems mentioned, there is the difficult identification of the tracking lines when the frames contain many lines similar in shape. This situation is very common in the case of cartoons, or video with a few shades of color or sharp lines. This usually results in false positives. Similar problems occur when the video presents the classic black bars to adapt to a certain aspect ratio. Their presence has different effects depending on the position. When these are located at the top or bottom of the frame, the percentages of the damage is altered. The "black" part of each frame always will be correct even if not contain any information. The case in which the bands are present in the right or left side will cause an incorrect calculation of the average of each rectangle described in detection algorithm, with the effect of false negatives. Moreover, in our implementation we use always a previous frame for the correction.

6   Conclusions

Although the analog technologies of recording and playback are intended to be completely replaced by digital ones, the development of algorithms for the correction of defects typical of analog media is required to ensure the preservation in the years ahead. Our experiments show that the proposed algorithm detects and corrects Mistracking in many cases. Moreover, some other drops like Tape Crease, Drop Out or Head Switching Noise, which have an error pattern similar to Mistracking one, are also partially corrected.

References

1. Stanco, F., Battiato, S., Gallo, G.: Cultural Heritage Preservation. Analysis, Restoration and Reconstruction of Ancient Artworks. Digital Imaging and Computer Vision. CRC Press (2011)
2. Stanco, F., Tanasi, D., Gallo, G., Buffa, M., Basile, B.: Augmented perception of the past. The case of hellenistic Syracuse. Journal of Multimedia 7(2), 211–216 (2012)
3. Stanco, F., Tenze, L., Ramponi, G.: Virtual restoration of vintage photographic prints affected by foxing and water blotches. J. Electronic Imaging 14(4), 043008 (2005)
4. Select "Analog video", `http://www.archeomatica.unict.it/`
5. RAI Radiotelevisione Italiana, `www.rai.it`
6. Gonzalez, R.C., Woods, R.E.: Digital image processing. Pearson Prentice Hall (2008)
7. Kokaram, A., Morris, R., Fitzgerald, W., Rayner, P.: Detection of missing data in image sequences. IEEE Transactions on Image Processing 4, 1496–1508 (1995)
8. Kokaram, A., Morris, R., Fitzgerald, W., Rayner, P.: Interpolation of missing data in image sequences. IEEE Transactions on Image Processing 4, 1509–1519 (1995)
9. Buisson, O., Besserer, B., Boukir, S., Helt, F.: Deterioration detection for digital film restoration. In: Proceedings of IEEE International Conference Computer Vision and Pattern Recognition, vol. 1, pp. 78–84 (June 1997)
10. Rosenthaler, L., Gschwind, R.: Restoration of movie films by digital image processing. In: Proceedings of IEEE Seminar on Digital Restoration of Film and Video Archives 2001 (2001)
11. Gupta, R.G.: Television Engineering and Video Systems. McGrawn-Hill Ed. (2005)
12. AV Artifact Atlas, `http://preservation.bavc.org/artifactatlas/index.php`