Adaptive Line Scratch Detection in Degraded Films

Alasdair Newson, Patrick Pérez
Technicolor
1 Avenue Belle Fontaine
35, Cesson-Sévigné
alasdair.newson@technicolor.com
patrick.perez@technicolor.com

Andrés Almansa, Yann Gousseau
Télécom ParisTech - LTCI CNRS
43 rue Barrault
75013 Paris
{almansa,gousseau}@enst.fr

ABSTRACT

In this paper, a robust, automatic and pixel-precision spatial line scratch detection algorithm is proposed. This algorithm deals with still images and may be followed by a temporal analysis to improve detection performances. By relaxing some of the hypotheses used in previous algorithms, detection of a wider range of scratch types is possible. The algorithm's robustness and lack of external parameters is ensured by the combined use of an *a contrario* methodology and local statistical estimation. In this manner, over-detection in textured or cluttered areas is greatly reduced. Experiments demonstrate the algorithm's ability to deal with difficult situations, in particular in the presence of noise, texture and slanted or partial scratches. Comparisons show the algorithm's advantages over previous work.

Keywords

Film restoration, line scratches, adaptive detection, *a contrario* methods

1. INTRODUCTION

Digital film restoration is a subject of increasing interest to researchers and film archives alike. Old films, including cultural heritage masterpieces, are being digitally remastered and transferred into new, higher quality formats and distributed through various means such as DVD, BluRay or HD cinema. A recent example of this is George Mélies's "Voyage dans la Lune" (1908).[1] Furthermore, with recent improvements in the areas of video indexing and retrieval, online access to vast archives such as those of the BBC and the INA is becoming ever easier. Combined with this is the tremendous amount of work required for the manual restoration of even one film. For instance, the restoration of the aforementionned fifteen minutes Mélies movie took about one year. For these reasons, it is crucial to develop automatic or semi-automatic tools for film restoration.

Some of the most common defects in films include dust/dirt, blotches, flicker and line scratches. While the first three are temporally *impulsive*, line scratches are persistent, meaning that they remain for several frames in the same spatial position. This means that detection and restoration algorithms need to be tailored specifically to this defect. Spatially speaking, line scratches are thin bright or dark lines which are roughly straight and vertical. In this paper we present a new approach to spatial line scratch detection with an algorithm that provides a pixel-precision detection. We relax several of the hypotheses made in the state-of-the-art algorithms (such as the verticality and straightness of scratches), allowing the algorithm to detect a wide range of scratch types. We rely on the *a contrario* methodology for the automatic setting of parameters. Furthermore, a local statistical estimation makes the method adaptive and able to deal with textured regions that are prone to over-detection. To our knowledge, there is no previous example in the literature where the *a contrario* methodology is combined with a local statistical estimation. The advantages of the approach will be demonstrated in Section 4 on a series of degraded film sequences and discussed in comparison with previous work.

The plan of the paper is as follows. In Section 2 we briefly recall previous work. In Section 3 we present the new method suggested for scratch detection. Finally, experimental validations are presented in Section 4.

2. PRIOR WORK

Scratch detection algorithms can be divided into two categories : spatial and temporal. As acknowledged in a very recent review [16], both approaches are complementary and benefit from one another's advantages.

Temporal approaches may be found in [12], [10], [11], [8] and [18]. The main goal of temporal filtering is to reject false detections which may arise during the spatial detection stage. In the first three papers, Joyeux *et. al* rely on a sinusoidal hypothesis concerning scratch motion for this rejection step. Another method which exploits temporal information is [8], where the authors use the local matching error between frames to determine if a detection is a true scratch or a vertical structure. Finally, in [18], Wang *et. al* use hidden Markov models to detect defects in films, based on temporal changes in the pixel's greyscale or colour values. Unfortunately, this last work supposes that the defects are temporally impulsive, which is usually not the case for line scratches.

In this work, we decided to concentrate on a spatial approach, which could be followed by temporal filtering. The spatial step still presents some challenging problems, as explained below.

The first work concerning spatial line scratch detection was carried out by Kokaram [14], and introduced a scratch model which is widely used in other papers (*eg.* [4] and [2]). This scratch

[1] Partly restored by Technicolor, presented at the Cannes Film Festival 2011.

CVMP '12, December 5 – 6, 2012, London, United Kingdom

model is based on an empirical observation concerning the shape of scratches, notably the presence of "side-lobes" either side of a scratch. In [14] and [2] the horizontal profile of a scratch is modelled by a damped sinusoid caused by light diffraction during the film scanning process. These approaches are still considered to be among the most efficient for scratch detection (see the recent review in [16]) and will be compared to the approach proposed in this paper. In other methods, such as [1] and [17], scratches are detected in the wavelet domain. The Hough transform is used in both [14] and [5] to detect prominent lines. Finally, Kim *et. al* use neural networks in [13] to establish scratch texture characteristics, which is followed by morphological filtering.

There are several unresolved problems in the literature concerning spatial scratch detection. Firstly, the scratch is represented as a straight, vertical line. While this may be the case in the examples which are commonly shown in most of the papers, it is not true in general. In some examples, the slant of the scratch may be quite significant, which renders the vertical representation almost useless for restoration purposes. Also, line scratches may be constituted of several segments of differing slopes, making the common vertical representation inefficient. We relax several of the hypotheses found in other work, allowing our algorithm to detect a wider variety of scratches.

Furthermore, experiments show that existing algorithms cope badly in noisy or textured regions. This problem is directly addressed in the present paper, through the automatic and adaptive setting of detection parameters.

3. PROPOSED ALGORITHM

The proposed algorithm is composed of two steps. The first is a pixel-wise scratch detection step, where we decide whether each pixel is potentially part of a line scratch. After this, we use *a contrario* methods to group the scratch pixels into visually significant scratch segments. The resulting grouping procedure is both automatic and adaptive and could be applied to different detection tasks.

3.1 Pixel-wise detection criteria

First of all, we introduce a test to distinguish potential scratch points from other pixels. There are several ways such a scratch profile could be detected, such as morphological filters ([11], [13]), or 1D extrema detection. The method presented here is a variant of the classical test used by Kokaram [14], which thresholds the following difference :

$$e(x,y) = G_s(x,y) - M_s(x,y), \tag{1}$$

where $G_s(x,y)$ is a vertically sub-sampled version of the input image, and $M_s(x,y)$ is a horizontally median filtered version of $G_s(x,y)$. This criterion determines whether the considered pixel is visibly out of sync with its horizontal surroundings. We change this difference slightly, by taking the median value of a local horizontal neighbourhood without the value of the pixel in question. This avoids the pixel having any influence on the median value. Also, we prefer to use a 3x3 Gaussian filter for noise reduction rather than vertical sub-sampling, in order to retain as much information about the scratch as possible. While this step induces some loss in precision, it is necessary due to the high amounts of noise and film grain in old films.

Unfortunately, this criterion alone may detect unwanted edges. To avoid this, another criterion concerning the average grey-level values either side of the scratch is used. We stipulate that these averages must be coherent to a certain extent, to avoid detecting intensity fronts. A visual illustration of our pixel-wise detection criteria may be seen in Figure 1.

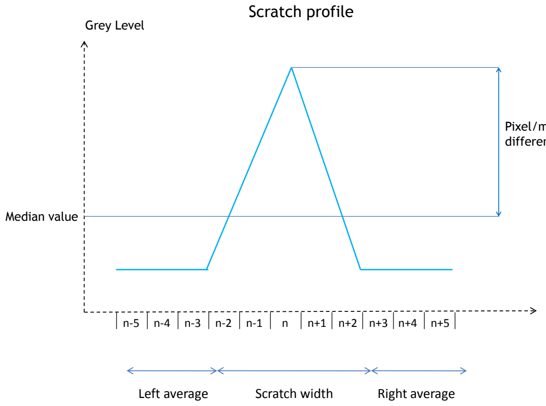
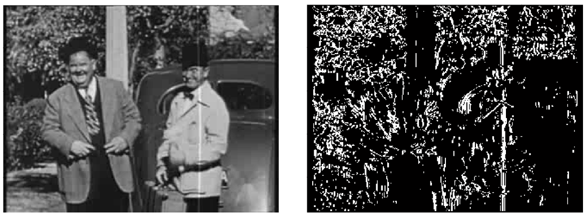


Figure 1: Line scratch profile and pixel-wise detection criteria.



(a) Original frame (b) Binary detection image

Figure 2: Binary detection image from "Laurel and Hardy". White pixels are detected and black pixels are not.

The previous criteria may be expressed as follows. Let $I_g(x,y)$ be the Gaussian filtered (we use a standard deviation of 1 pixel) grey level image. Let $I_m(x,y)$ denote the median value over a local horizontal neighbourhood of pixel (x,y), and $I_l(x,y)$ and $I_r(x,y)$ be the left and right horizontal averages. The two Boolean criteria are

$$\begin{array}{lcl} c_1(x,y) & : & |I_g(x,y) - I_m(x,y)| \geq s_{med} \\ c_2(x,y) & : & |I_l(x,y) - I_r(x,y)| \leq s_{avg} \end{array} \tag{2}$$

where, s_{med} and s_{avg} are grey-level thresholds. We can therefore define the binary image as

$$I_B(x,y) = \begin{cases} 1 & \text{if} \quad c_1(x,y) \quad \text{and} \quad c_2(x,y) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In the present algorithm, the median filter has a width of 5 pixels and the value of s_{med} is set to 3. These values are the same as in [14] and also appeared to us to be good empirical choices. The left and right averages are each taken over 3 pixels on either side of the 5 central pixels, and s_{avg} has been experimentally set to 20.

Once this binary detection image is obtained (see Figure 2). In the figure, the white points are detected as potential scratch pixels whereas the black points are not. Once such a detection image is obtained, the points must be grouped into significant scratch *segments*.

3.2 Scratch point grouping and validation

Because of false detections due to noise and texture (see Figure 2), a robust approach is needed to group the pixels into segments. One may do this by using the Hough transform, as in [14] and [5]. Unfortunately this method has some serious drawbacks, such as thresholds which need frequent tuning. These problems can be addressed by the use of *a contrario* methods as used for alignment detection by Desolneux *et al.* in [7].

3.2.1 A contrario line segment detection

In a word, the *a contrario* methodology is a generic way to detect visual objects in digital images. Detection thresholds are set in order to control the number of false detections in a white noise image, or more generally under a *background model*. This model usually relies on an independence assumption on the basic elements to be grouped for the detection.

First of all, we present the *a contrario* approach as it is used to detect line segments [6]. Given a line segment made of l pixels, $\mathbf{x}_1, \ldots, \mathbf{x}_l$, a random variable X_i is associated with each pixel $\mathbf{x}_i$. X_i is equal to one if the pixel is aligned with the segment and 0 otherwise. Aligned pixels are those whose gradient orientation is orthogonal to the direction of the segment, up to some precision. Let $S_l = \sum X_i$ be the number of aligned pixels. Detection thresholds are set by computing the distribution of S_l under some *background model*. In the case of line segments, the background model specifies that the X_i's are *independent* random variables, each having a Bernoulli distribution of parameter p (the precision on the gradient orientation alignment). Under the background model, the distribution of S_l is the binomial law, and hence

$$\Pr(S_l \geq k_0) = \sum_{k=k_0}^{l} \binom{l}{s} p^k (1-p)^{l-k} =: B(p;k_0,l). \qquad (4)$$

The last step of the detection procedure relies on a control of the expected number of false detections. For this, given a segment of length l having k_0 aligned pixels, its *number of false alarms* (NFA) is defined as

$$\mathrm{NFA}(l,k_0) = N_{tests} B(p;k_0,l), \qquad (5)$$

where N_{tests} is the total number of segments to be tested. With the exploration strategy used in [6], we have $N_{tests} = M^2N^2$, with M and N the linear dimensions of the image. A segment is detected if $\mathrm{NFA}(l,k_0) \leq \varepsilon$ for some parameter ε. The rationale for this threshold is that it implies that the expected number of detections under the background model (false detections) is less than ε, see [7].

3.2.2 Locally adaptive grouping for line scratch detection

We now rely on the same principles to group pixels that have been detected by the pixel-wise procedure of Section 3.1. We must first define a background model to represent the binary image obtained with Equation (3). In the case of orientation grouping (see Section 3.2.1), it is sound to assume that any pixel in the image has the same probability to be aligned with a given segment, whatever its position. In the present case, the probability that a pixel is labelled as being a potential scratch point ($I_B(x,y) = 1$, with the notations of Section 3.1) varies greatly depending on the area of the image. In particular, strongly textured or cluttered areas yield many more detections than smooth regions, as seen in Figure 2. Moreover, the visibility of scratches depends on their local neighbourhood. To account for this, the background model is now considered as a binary image in which labels are independent and the label probability of each pixel varies spatially. The computation of this probability will be based on a *locally adaptive estimation*.

The label probability for a given pixel is estimated as the maximum detection density on four squares of equal size surrounding the pixel. Detection density is defined as the proportion of pixels, contained within a square, whose labels equal 1. In all experiments, the square size is the width of the image divided by a constant, which we set to 30. We use four squares for the estimation in order to deal with cases where the pixel is on the edge of several areas of differing detection density.

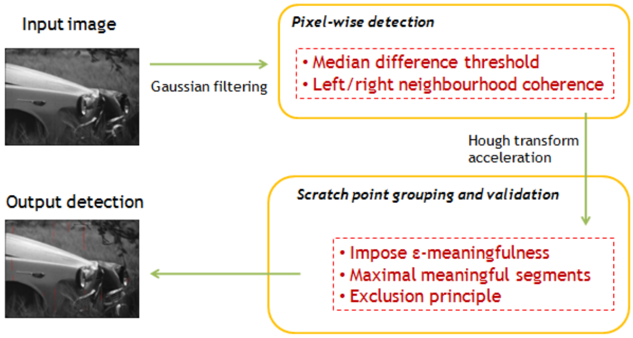

Input image
Gaussian filtering
Pixel-wise detection
• Median difference threshold
• Left/right neighbourhood coherence
Hough transform
acceleration
Output detection
Scratch point grouping and validation
• Impose ε-meaningfulness
• Maximal meaningful segments
• Exclusion principle

Figure 3: Summary diagram of the proposed algorithm.

Under this background model - where the probability that a pixel is labelled is spatially varying - the probability for a given segment to have more than k_0 pixels with a label value of 1 is no longer given by a binomial distribution, and is quite costly to estimate. An approximation is therefore needed. In [7], Desolneux *et. al* suggest the use of Hoeffding's inequality ([9]), which provides an upper bound on the probability that the sum of some independently distributed random variables exceeds a certain value. In the present case, the interest of this approximation is that it it still holds when the variables are not identically distributed ([9]). Therefore, it provides us with an approximation of $\Pr(S_l \geq k_0)$, where again S_l is the number of pixels having a label value of 1 along a segment of length l. The approximation is the following :

$$\Pr(S_l \geq k_0) \leq H(l,k_0) := e^{-l(r\log\frac{r}{pm}+(1-r)\log\frac{1-r}{1-pm})}, \qquad (6)$$

where p_m is the average detection probability along the segment, $r = \frac{k_0}{l}$, and $p_m l < k_0 < l$. We therefore define the number of false alarms of a segment as

$$\mathrm{NFA}(l,k_0) = N_{tests} H(l,k_0). \qquad (7)$$

A segment is detected if its NFA is smaller than ε (such a segment is said to be "ε-meaningful"). Thus, the expected number of detected segments under the background model is smaller than ε. In all experiments, we use the parameter $\varepsilon = 1$.

This choice is reasonable, since ε is a bound on the expected number of false detections under the background model. However, it may be further tuned to fit the user's needs, depending on whether he or she wishes to place importance on recall or precision.

Furthermore, as explained in [6], detection results vary slowly with respect to ε, meaning making it an easy parameter to tune, if so desired. An illustration of this may be seen in Figure 4. In this figure, ε varies by several orders of magnitude, yet the f1-score remains high for values of ε in the interval $[10^{-30}, 10^{10}]$.

Since scratches are roughly vertical, we test all segments with a maximum deviation from the vertical direction of ± 10 degrees. We discretise these angles by 0.5 degrees. The N_{tests} parameter is therefore set to $M^2N\Theta$, where Θ is the number of angles tested. In the present paper, we have chosen $\Theta = 40$.

3.2.3 Maximality

With the previous detection procedure, many nested segments

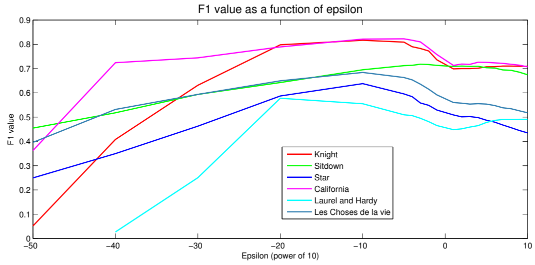


Figure 4: f1-score curves with respect to ε.

are detected. In order to keep only one detection for such nested segments, we use the notion of *maximality*, as introduced in [6]. A segment is *maximal meaningful* if it neither contains nor is contained by a segment which is more meaningful (that is, a segment with a smaller NFA). Therefore, we only accept segments which possess this property.

It may be shown (see [7]) that maximal meaningful segments start with a detected pixel preceded by an undetected pixel, and end similarly (see [7]). Knowing this, the number of segments tested, and therefore the computational cost, is *greatly* reduced. This result relies on the original definition of the NFA (Equation (5)) and the properties of the binomial law, however it may be shown (see Annex A) that this property remains true using Hoeffding's approximation.

3.2.4 Exclusion principle

Since scratches have a width of several pixels, many slanted segments correspond to the same scratch. In order to refine the detection, we use an *exclusion principle* as defined in [7], which states that a pixel may belong to one scratch only. If a pixel s is contained by several segments, then the most meaningful segment retains s. All other segments which contain s have it removed. The NFAs of the modified segments are recalculated, and those that are no longer ε-meaningful are thrown away. The entire algorithm is represented in Figure 3.

3.2.5 Algorithm speed-up

In order to speed-up the procedure, we apply a pre-selection of scratches candidates. For this, we apply a very permissive Hough transform to I_B, and only analyse the lines which correspond to local maxima. Tests show that no or very few real scratches are lost by this pre-processing.

4. RESULTS

In this section, the results of the algorithm are shown, and its performance is compared with a state-of-the-art algorithm (see [3]) with respect to three criteria : recall, precision and the F1-score. While another recent method [13] exists, it is a supervised algorithm (contrary to Bruni's and ours, which are automatic) and contains several parameters which are not given in the paper, such as the number of nodes in the input and hidden layers of the neural network, making implementation impossible without testing a series of architectures. In order to show the importance of having a *locally adaptive* noise model, we also compare our results to those of our algorithm when a global noise estimation is used. We have estimated this by taking the average probability of a pixel having a label value of 1 in the whole image.

Tests were carried out on six film sequences of varying length and difficulty. The first three ("Knight", "Sitdown" and "Star") are common in the scratch detection literature, and are found in Kokaram's book, referenced in [15]. "California" and "Laurel and Hardy", contain straight, vertical scratches, similar to the first three examples. The last one ("Les Choses de la Vie") contains scratches which are more difficult to detect (not completely straight, slanted and/or faint). Figure 5 shows some examples of our detection results, compared with those of the algorithm presented in [3].

In these experiments, we have retained the parameter values as presented throughout the paper, in particular $\varepsilon = 1$. The parameters for Bruni's algorithm are those given [2] and [3], apart for the scratch colour parameter (black or white), which was set manually for each sequence. In these experiments, for the evaluation of recall and precision purposes we allow a distance of 2 pixels between the annotated centre of the scratch and the detected centre. Therefore, we do not enforce the exclusion principle, which would be redundant. All sequences were annotated by hand, by one of the authors. This was done by manually noting the beginning and end point of a scratch segment. For scratches which were not completely straight, separate segments were annotated. The annotations, comparisons and also a short demonstration video, may be found at the following address : http://www.enst.fr/~gousseau/scratch_detection.

4.1 Recall

Recall is based on the percentage of annotated scratch pixels detected. Bruni's method provides a column index to locate the scratch. From a restoration point of view then, the detected pixels are all those in the given column. The recall results can be seen in Table 1.

The results show that for the first five sequences, recall is similar in both algorithms. In the remaining ones, we see our algorithm's strong points : it is able to detect the difficult scratches. This may be explained by our algorithm's ability to detect and represent slanted and disjointed scratches as a collection of segments with varying length and angle. Our recall remains high for all sequences.

4.2 Precision

In order to be fair for precision evaluation, we judge our algorithm on a pixel-wise basis, whereas we will consider a column given by Bruni's algorithm to be correct if it touches at least one scratch pixel. Thus, our evaluation methodology confers a considerable advantage on Bruni's method, since a column has only to touch one scratch point in order to be correct. Nevertheless, our algorithm our algorithm is able to outperform Bruni's in five out of six of the sequences. The results of the precision evaluation may be seen in Table 1. In should be noted that the sequences "Sitdown" and "Star" are highly degraded, and not all the defects detected are annotated as scratches. True precision is therefore underestimated.

Our algorithm is able to outperform Bruni's in terms of precision due to the *a contrario* grouping and validation process, which limits the number of false detections as long as our noise model holds. The adaptive nature of the model means that false detections may be avoided in a wide range of texture and noise.

Furthermore, the use of local noise estimation becomes evident with these results. In images with textured areas, such as "Laurel and Hardy" (see Figure 2), global noise estimation produces many false detections.

4.3 $F1$-score

The $F1$-score is defined as the harmonic mean of the recall and precision :

$$F = 2\frac{recall * precision}{recall + precision} \tag{8}$$

Table 1: Recall, precision and F1 values comparison, in percentage. We compare Bruni's results, and ours with global and local noise estimation. Note that local estimation degrades recall only slightly, while precision improves greatly.

Film	Recall (%)			Precision (%)			f1-score (%)		
	Bruni's	Global	Local	Bruni's	Global	Local	Bruni's	Global	Local
Knight	**100.0**	90.10	81.97	29.54	37.09	**63.47**	45.61	52.55	**71.54**
Sitdown	**80.93**	85.02	77.66	56.47	63.49	**71.73**	66.52	72.69	**74.58**
Star	95.00	**96.92**	90.00	**56.87**	37.10	41.59	**71.15**	53.66	56.88
California	82.07	**94.91**	88.71	10.79	22.51	**69.24**	19.07	36.39	**77.77**
Laurel and Hardy	41.87	**81.83**	66.01	8.84	24.75	**44.41**	14.60	38.01	**53.10**
Les Choses de la vie	43.43	**88.44**	82.54	25.06	43.06	**51.40**	31.78	57.92	**63.35**

This score illustrates the performances of the methods more clearly than either the recall or precision alone. The results show that our algorithm retains a good F1-score for all of the sequences, and outperforms Bruni's algorithm in five out of six sequences.

In an interactive restoration process, this parameter (ε) can be changed by the user to fit his or her needs. For instance, even though the F1-score may favour one result, the user may want to achieve a certain recall, or conversely, ensure that few false alarms occur. This is highly dependant on the subsequent restoration which is available, and therefore linked to the final quality of the restored film. In Figure 4, the evolution of the f1-score with respect to ε may be seen.

4.4 Robustness to noise and texture

In this section, some visual evidence is given which shows our algorithm's robustness to noise and texture, and the usefulness of having such properties. This can be seen in the images in Figure 6. Due to the highly textured nature of the images, false positives are detected, when we try to use our algorithm with no local noise estimation. When we introduce local noise estimation, detections are limited in areas with high noise density, which illustrates one of the strong points of our algorithm. In the image of the monkey, we produce one false alarm with local noise estimation, which is coherent with our threshold.

4.5 Scratch detection in high definition images

In Figure 7, an example of a high definition image containing scratches is given. This example is interesting since, as was mentioned at the beginning of the paper, the restoration of films is being done for formats of high resolution. To the best of the authors' knowledge, no other high definition example exists in the literature. It can be seen that the algorithm from [3] is unable to detect the faint, white scratches present on the right hand side of the image, whereas the proposed method locates them with a high degree of spatial precision.

5. CONCLUSION

In this paper we have presented a precise line scratch detection algorithm which uses an *a contrario* validation step to determine if the detected segments are visually significant or not. Our algorithm provides a precise description of the detected scratches, which is not given by any other fully automatic algorithm. Furthermore, it has similar performance to the state-of-the-art in simple cases, and outperforms the latter considerably in more difficult situations. This validates one of the main goals of detecting as many types of scratch as possible. Finally, our algorithm is parameterless, and its evaluation was carried out without any sequence-dependant tuning. Future research will address the problem of distinguishing thin vertical structures from line scratches, which is a source of many false detections. The use of temporal information such as scratch and scene motion should help to achieve this reliably.

6. ACKNOWLEDGMENTS

The authors would like to express their gratitude to Vittoria Bruni and Domenico Vitulano for having made an implementation of their method available and for having answered several questions about their line scratch detection. The authors would also like to thank Jesse Crowder of Dijifi for making his high definition images available. Thanks also go to Studio Canal for allowing the use of the frames from "Les Choses de la Vie". Finally, the authors would like to acknowledge the internet archive http://www.archive.org for the video material which is made available online.

7. REFERENCES

[1] T. Bretschneider, O. Kao, and P. Bones. Removal of vertical scratches in digitised historical film sequences using wavelet decomposition. *Proceedings of Image and Vision Computing New Zealand*, 2000.

[2] V. Bruni and D. Vitulano. A generalised model for scratch detection. *IEEE Trans on Image Proc.*, 13(1), 2004.

[3] V. Bruni and D. Vitulano. Removal of colour scratches from old motion picture films exploiting human perception. *EURASIP*, 2008.

[4] V. Bruni, D. Vitulano, and A. Kokaram. Line scratches detection and restoration via light diffraction. *IPSA*, 1, 2003.

[5] K. Chishima and K. Arakawa. A method of scratch removal from old movie film using variant window by hough transform. *Proceedings of the 9th international conference on Communications and information technologies*, 2009.

[6] A. Desolneux, L. Moisan, and J.-M. Morel. Meaningful alignments. *IJCV*, 40(1), 2000.

[7] A. Desolneux, L. Moisan, and J.-M. Morel. *From Gestalt Theory to Image Analysis : a Probabilistic Approach.* Springer-Verlag, 2008.

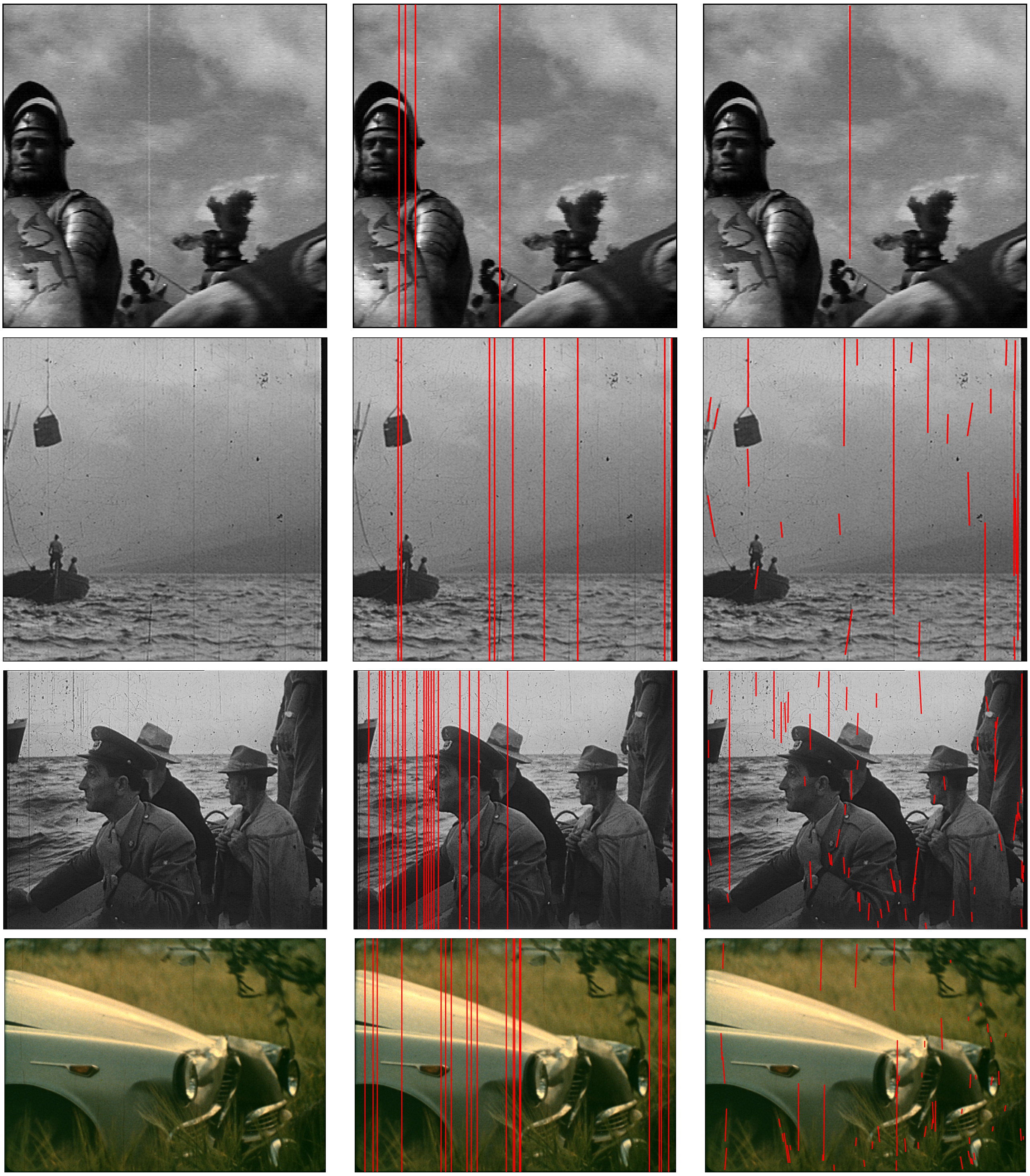
[8] M. Güllü, O. Urhan, and S. Ertürk. Scratch detection via temporal coherency analysis and removal using edge priority based interpolation. *ICAS 2006*, 2006.

[9] W. Hoeffding. Probability inequalities for sums of bounded random variables. *JASA*, 58(301), 1962.

[10] L. Joyeux, S. Boukir, and B. Besserer. Film line scratch removal using kalman filtering and bayesian restoration. *Applications of Computer Vision, 2000, Fifth IEEE Workshop on.*, pages 8–13, 2000.

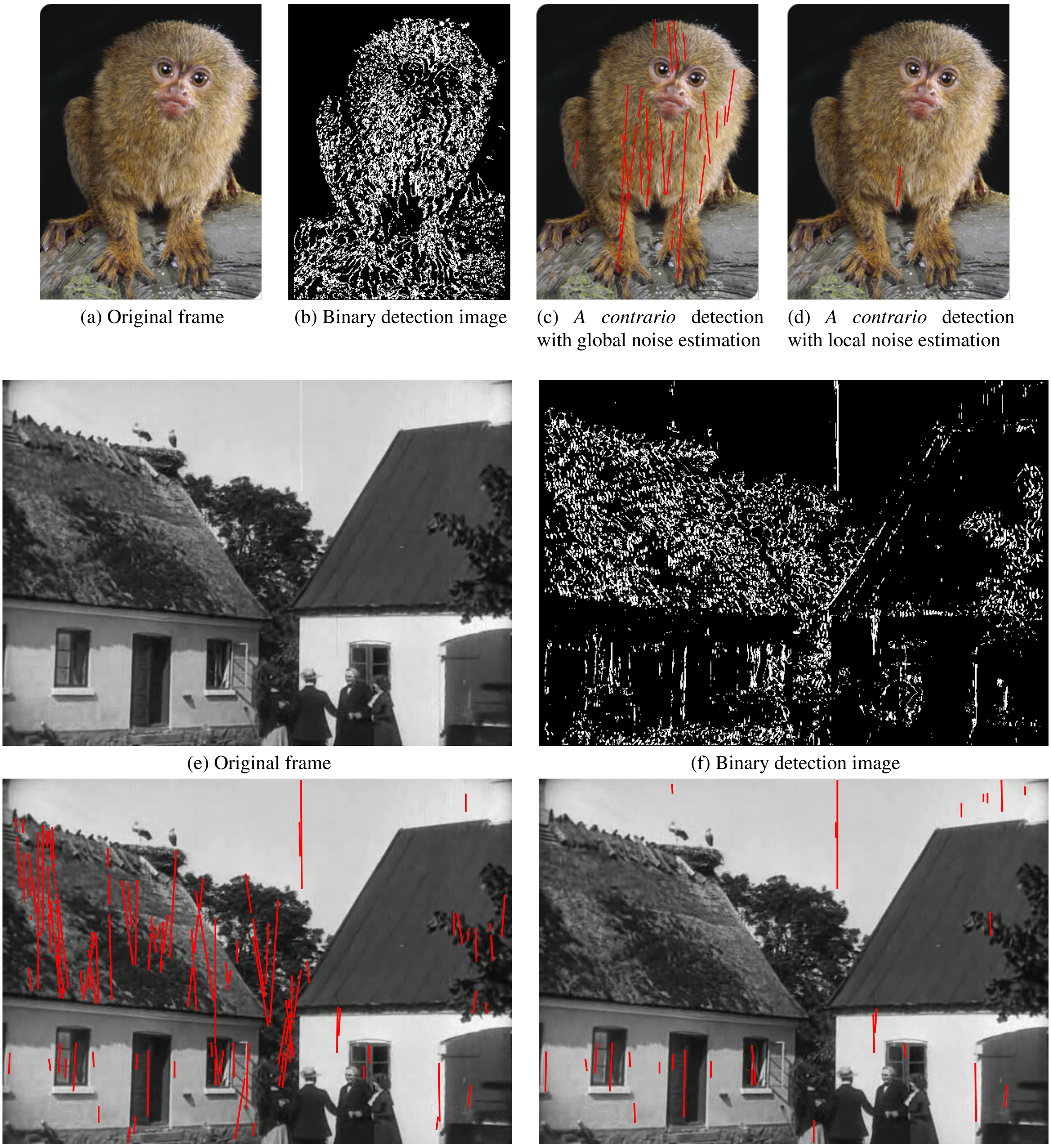
[11] L. Joyeux, S. Boukir, and B. Besserer. Tracking and map reconstruction of line scratches in degraded motion pictures. *Machine Vision and Applications*, 13(3), 2002.

[12] L. Joyeux, O. Buisson, B. Besserer, and S. Boukir. Detection and removal of line scratches in motion picture films. *CVPR*, 1999.



(a) Original frame (b) Detection from [3] (c) Our detection

Figure 5: From top to bottom : "Knight", "Sitdown", "Star" and "Les Choses de la Vie".



(a) Original frame

(b) Binary detection image

(c) *A contrario* detection with global noise estimation

(d) *A contrario* detection with local noise estimation

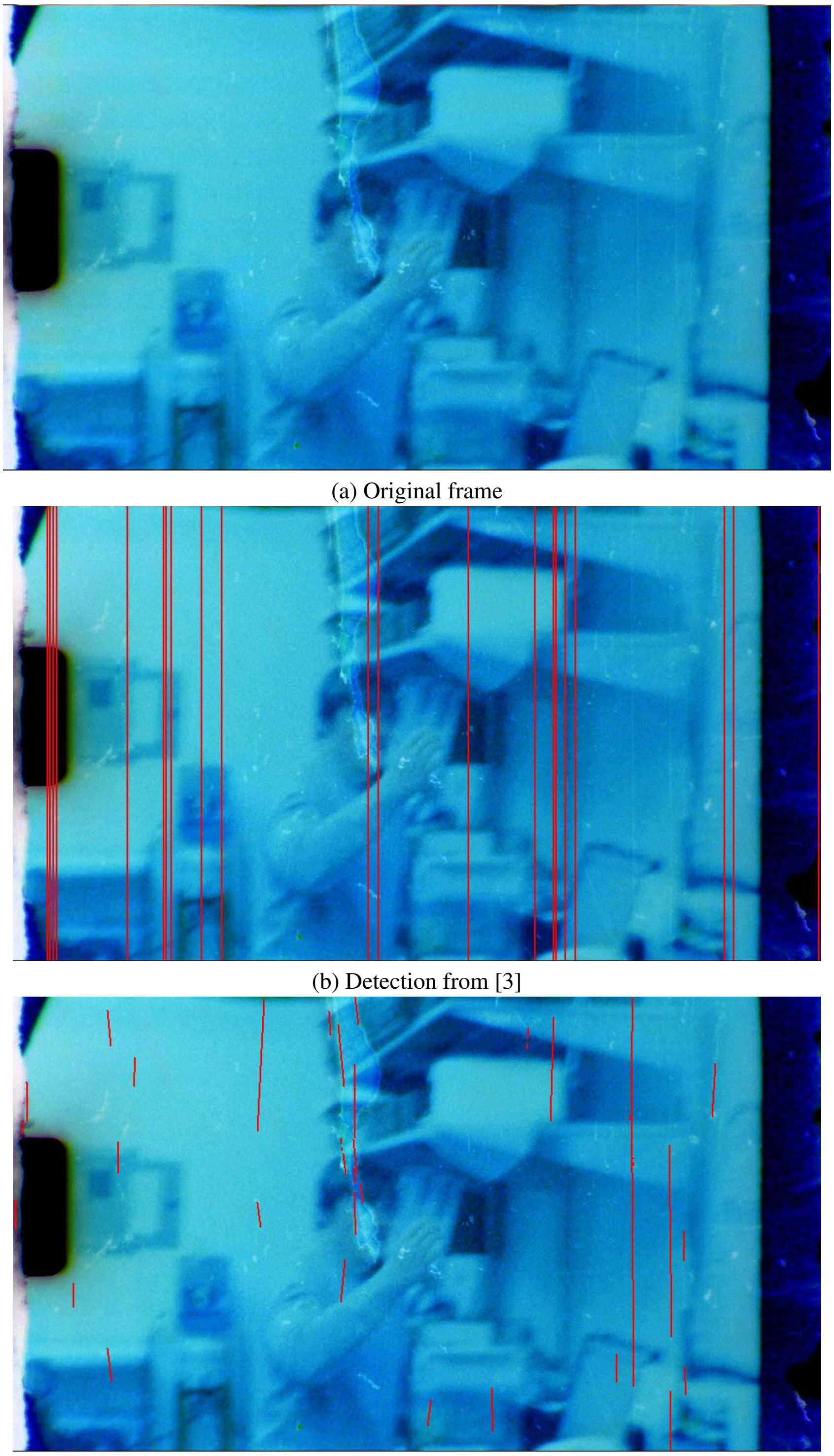
(e) Original frame

(f) Binary detection image

(g) *A contrario* detection with global noise estimation

(h) *A contrario* detection with local noise estimation

Figure 6: An illustration of our method's robustness to texture and noise on two highly textured test images. In the first example, our locally adaptive method detects only one scratch, whereas with a global noise estimation, the algorithm fails. The second example illustrates this principle on an example with real scratches.



(a) Original frame

(b) Detection from [3]

(c) Our detection

Figure 7: A high definition scratched film (1074x1920). Note that a minimum segment size of 50 pixels was imposed here.

[13] K. Kim and E. Kim. Film line scratch detection using texture and shape information. *Pattern Recognition Letters*, 31(3), 2010.

[14] A. Kokaram. Detection and removal of line scratches in degraded motion picture sequences. *Signal Processing VIII*, 1(9), 1996.

[15] A. Kokaram. *Motion Picture Restoration: Digital Algorithms for Artefact Suppression in Degraded Motion Picture Film and Video*. Springer Verlag, 2001.

[16] A. Kokaram, F. Pitie, D. Corrigan, D. Vitulano, V. Bruni, and A. Crawford. Advances in automated restoration of archived material. In F. Stanco, S. Battiato, and G. Gallo, editors, *Digital Imaging for Cultural Heritage Preservation: Analysis, Restoration, and Reconstruction of Ancient Artworks*. CRC Press, 2011.

[17] S. Müller, J. Bühler, S. Weitbruch, C. Thebault, I. Doser, and O. Neisser. Scratch detection supported by coherency analysis of motion vector fields. *Proceedings of the 16th IEEE international conference on Image processing*, 2009.

[18] X. Wang and M. Mirmehdi. Hmm based archive film defect detection with spatial and temporal constraints. *BMVC*, 2009.

APPENDIX

A. MAXIMALITY PROPERTY AND HOEFFDING'S APPROXIMATION

In this section, we prove two properties of the meaningful segments defined using the NFA relying on Hoeffding's approximation, Formula 7. These properties are necessary for speeding up the search for maximal segments, as explained in Section 3.2.3. These properties are as follows :

- If one appends a 0 (non-detected pixel) to the segment, its meaningfulness decreases
- If one appends a 1 (detected pixel) to the segment, its meaningfulness increases

Using Formula 7, this reduces to :

$$H[l,k] \leq H[l+1,k], \tag{9}$$

and

$$H[l,k] \geq H[l+1,k+1], \tag{10}$$

where H is defined as

$$H(l,k) = \exp(-k\log\frac{k}{lp} - (l-k)\log\frac{l-k}{l(1-p)}), \tag{11}$$

where $lp < k < l$. Since the exponential function is strictly increasing, we need to study the following function :

$$f(k,l) = -k\log\frac{k}{lp} - (l-k)\log\frac{l-k}{l(1-p)}. \tag{12}$$

Now, let us try to prove Equations (9) and (10).

For Equation (9) to be true, we need for the partial derivative of f with respect to l to be positive :

$$\frac{\partial f(k,l)}{\partial l} = k\frac{1}{l} - [\log\frac{l-k}{l(1-p)} + (l-k)\frac{\partial}{\partial l}\log\frac{l-k}{l(1-p)}]. \tag{13}$$

We have the partial result :

$$\frac{\partial}{\partial l}\log\frac{l-k}{l(1-p)} = \frac{k}{l(l-k)}.$$

Therefore,

$$\begin{aligned}\frac{\partial f(k,l)}{\partial l} &= \frac{k}{l} - \log\frac{l-k}{l(1-p)} - (l-k)\frac{k}{l(l-k)} \\ &= \log\frac{l-lp}{l-k}\end{aligned}$$

We know that $l - lp > l - k$, because $lp < k$ (condition for the Hoeffding approximation). Therefore, the right hand term of Equation (13) is strictly positive, so that $f(k,l)$ increases strictly when l increases. This means that when a 0 is appended to a segment, its meaningfulness decreases (since its probability increases). We have proven the first inequality (Equation (9)) in the case of Hoeffding's approximation.

Now we prove Equation (10). This inequality corresponds to the increase in meaningfulness (and therefore the decrease in probability) when a 1 is appended to a segment. This case is slightly more complicated, since two variables (k and l) vary at the same time. We know, however, that they vary at the same rate, since we are adding 1 to both of them. We can therefore consider k and l fixed, and add a new variable to each of them, which we shall name t. We can therefore study the partial derivative of $f(k+t,l+t)$ with respect to t.

We have

$$f(k+t,l+t) = -(k+t)\log\frac{k+t}{(l+t)p} - (l-k)\log\frac{l-k}{(l+t)(1-p)},$$

so that :

$$\begin{aligned}\frac{\partial f(k+t,l+t)}{\partial t} &= -\log\frac{k+t}{(l+t)p} - (k+t)\frac{\partial}{\partial t}\log\frac{k+t}{(l+t)p} \\ &- (l-k)\frac{\partial}{\partial t}\log\frac{l-k}{(l+t)(1-p)}.\end{aligned}$$

Now

$$\frac{\partial}{\partial t}\log\frac{k+t}{(l+t)p} = \frac{l-k}{(k+t)(l+t)},$$

and

$$\frac{\partial}{\partial t}\log\frac{l-k}{(l+t)(1-p)} = -\frac{1}{l+t},$$

so that

$$\begin{aligned}\frac{\partial f(k+t,l+t)}{\partial t} &= -\log\frac{k+t}{(l+t)p} - (k+t)\frac{l-k}{(k+t)(l+t)} \\ &+ (l-k)\frac{1}{l+t} \\ &= \log\frac{lp+tp}{k+t}.\end{aligned}$$

This is strictly negative, since $lp < k$ and $tp < t$. Therefore, $f(k,l)$ decreases when k and l increase together. Therefore, we have in particular, that $H(l+1,k+1) < H(l,k)$, meaning that meaningfulness increases if a 1 is appended to a segment. Thus, the second inequality (Equation 10) holds.

We have proven the two necessary properties for a segment to be maximal meaningful in the case of Hoeffding's approximation, thus we can safely prune the search for maximal meaningful segments.