A Fast and Efficient Blotch Detection Algorithm for Degraded Image Sequences Based on MRF Models
Sang-Churl Nam, Masahide Abe and Masayuki Kawamata
Department of Electronic Engineering, Graduate School of Engineering, Tohoku University
Aoba-yama 6-6-05, Sendai 980-8587, Japan
Tel: +81-22-795-7095, Fax: +81-22-263-9169

Abstract: This paper proposes an efficient blotch detection algorithm based on a Markov Random Field (MRF) model with less computational complexity and with lower false alarm rate than the existing MRF-based algorithms. The proposed algorithm can save the computational time for detecting the blotches by restricting the attention of the detection process only to the candidate areas. To solve the problem of confusion with the blotches in the vicinity of the moving object, we incorporate the weighting function with respect to the detected moving edge pixels into the formation. Experimental result shows that our method can provide the computational simplicity and an efficient detecting performance for the blotches.

1. Introduction
The digital film restoration at high resolution such as 4K resolution (4096×2160 pixels) for digital cinema has been steadily increasing according to the increasing prevalence of visual digital media. Many of old films are valuable historical and cultural records. However, most of them have undergone a variety of degradations, which reduce their usefulness.

The central issue in digital film restoration is to detect and remove the blotches damaged by dirt and abrasion of the film material. Although the performance of the blotch detectors strongly depends on the sequences themselves, it has been found that the Markov Random Field (MRF) approach performs generally better than other existing detectors, for example, SDIa detector, AR detector, and so forth [1][2][3]. However, if we apply the existing MRF-based blotch detector on high resolution image frames, considerably high computational cost is inevitable. Consequently, a fast and efficient algorithm for the blotch detection is required.

In this paper, we propose an efficient algorithm based on an MRF model for detecting blotches with less computational cost and with lower false alarm rate than the existing MRF-based detectors.

2. Proposed Algorithm
The proposed algorithm consists of the four steps: (I) motion estimation, (II) estimation of the candidate Blotch Detection Mask (BDM), (III) moving edge detection, (IV) solution of the MAP problem.

(I) Motion estimation
In this step, we discuss a motion estimation based on block matching. To begin with, we divide an image frame into blocks. Then, for each current block in the current frame, the block in a reference frame that is the most similar is searched according to minimum mean absolute difference (MAD) criterion. Finally, the displacement estimate is given by

\[ \hat{v} = \arg \min_v \text{MAD}(v) \]  

(1)

In addition, we introduce the method of Boyce [5] in order to prevent erroneous motion vectors caused by noise in image sequences.

(II) Estimation of the candidate BDM
In this step, we present the candidate blotch detection mask (BDM) consisting of binary labels for each pixel by the threshold calculated by performing the significance test [6]. Given the current frame \( I(r) \) and the motion-compensated reference frame \( I_{mc}(r) \), the intensity difference image, \( Z = \{z_r\} \), with \( z_r = I(r) - I_{mc}(r) \), is modeled as Gaussian noise with a variance \( \sigma^2 \) equal to twice the variance of the noise. A intensity difference \( z_r \) is assumed to correspond to noise (null hypothesis \( H_0 \)) and not to the candidate blotch regions (hypothesis \( H_1 \)). Assuming that neighboring pixels are statistically independent, the normalized square sum within a window \( w \) of size \( N \times N \) is known to obey a \( \chi^2 \) distribution with \( N^2 \) degrees of freedom. Thus the threshold \( T_r \) can be determined from

\[ \Pr(T_r > t_\alpha \ | H_0) = \alpha, \quad T_r = \sum \frac{z^2}{\sigma^2} \]  

(2)

where \( \alpha \) and \( t_\alpha \) are a significance level and its corresponding threshold, respectively. Whenever \( T_r \) exceeds \( t_\alpha \), we decide a candidate BDM \( Q(r) = 1 \), otherwise \( Q(r) = 0 \). In this way, we can obtain the candidate forward and backward BDMs, \( Q_{\text{forw}} \) and \( Q_{\text{backs}} \) with respect to the adjacent frames.

The candidate BDMs present regions corresponding to the candidate region of the blotches in the current frame performs satisfactorily. Therefore, the maximization of the MAP problem is performed with respect to the pixels corresponding to \( Q_{\text{forw}}(r) = 1 \) for the forward direction and \( Q_{\text{backs}}(r) = 1 \) for the backward direction.

As a result, by means of restricting the blotch detection region to the candidate BDMs, our proposed method can drastically save the computational time for detecting blotches.

(III) Moving edge detection
In this step, we detect the moving edge field to avoid the confusion with the blotches caused by the poorly motion-compensated pixels. In order to find the pixels caused by being falsely estimated by motion vectors, we take advantage of the forward and backward motion-compensated frames. Assuming that there are no blotches in the two motion-compensated neighboring frames, we generate the edge field \( E(r) \) for the neighboring motion compensated frames using the significant test as stated above.

Whenever the moving edge field, \( E(r) = 1 \), the weighting function is defined as follows [4]:

\[ \psi(r) = \begin{cases} \kappa \max \{ CE(r) \} & \text{if } \sum CE(r) > \text{Th} \\ 0, & \text{otherwise} \end{cases} \]  

(3)

Here, \( \kappa \) and \( \text{Th} \) denote a weighting constant and a threshold, and \( CE(r) \) is defined as the set of four elements belong to
the square difference between two adjacent pixels in the horizontal, vertical, and two diagonal direction with respect to the centered pixel \( r \), respectively.

IV) Solution of the MAP problem

In this step, we discuss the solution of the MAP problem. Let \( S \) denote the pixel lattice of two adjacent frames and \( i(r) \) denote the observed intensity. Let \( \hat{i}(mc) \) denote the motion-compensated pixel of temporal neighboring frame. The a posteriori distribution for a binary blotch mask \( D \) is defined as follows [4]:

\[
P(D = d | I = i) = \frac{1}{Z} \exp \left( -\frac{1}{T} \sum_{r \in S} \left[ \alpha (1 - d(r))i(r) - i(\hat{r}_{mc}) \right]^2 \right. \\
\left. - (\beta_1 + \psi(i(r)))f(d(r)) + (\beta_2 + \psi(i(r)))\delta(1 - d(r)) \right)
\]

where \( Z \) is a normalizing constant, \( T \) is a temperature, \( f(d(r)) \) is the number of the four neighbors of \( d(r) \) with the same value as \( d(r) \), \( \delta(\cdot) \) is the delta function, and \( \alpha, \beta_1 \) and \( \beta_2 \) are the parameters to determine the characteristics of detectors.

The above MAP problem for detecting blotches can be maximized using simulated annealing method using a logarithmic annealing schedule. It is maximized once with respect to the pixels corresponding to \( Q_{\text{fmc}}(r) \) for the forward direction and once with respect to the pixels corresponding to \( Q_{\text{bmc}}(r) \) for the backward direction. The resulting blotch mask is the set of all elements belonging to both backward and forward detection mask.

3. Experimental Results

To compare the efficiency of the proposed detector, we used the test sequence, “Intersection”. A 480×720 subset of the sequence was artificially corrupted with blotches of varying size and shape and random gray value [2]. In addition to this, to make the test sequence similar to real degraded image data, white Gaussian noise with variance 10 was added after the blotches were added. We have performed three blotch detection algorithms using MATLAB running on an Intel XEON 3.4 GHz machine with a Linux operating system.

In our experiment, Based on the significance level \( \alpha_r = 10^{-2} \), we obtained a threshold of \( t_r \) for the square sum via a \( \chi^2 \) distribution of 81 degrees of freedom in Eq. (2), and \( \chi^2 = 0.03 \) and \( Th = 50 \) in Eq. (3). The parameters were found by setting 0.2 ≤ \( \alpha \) ≤ 2.0, \( \beta_1 = 30 \) and \( \beta_2 = 50 \).

Figure 1-3 show the results of the application of the detectors to the problem of detecting the blotches. Green indicates correctly detected blotched pixels, red represents falsely detected blotched pixels, and blue stands for a missed blotched pixels. It shows a decrease of the false alarm rate of detector [4] as compared with detector [2]. However, it remains the falsely detected blotched pixels in the proximity of moving objects. On the contrary, our proposed method is capable of detecting well the blotched pixels with the fewest number of falsely detected blotch pixels. In addition, the computation time required for detecting blotches per frame takes on average 34.14s, whereas the computation time of Ref. [4] and Ref. [2] take on average 178.71s and 179.59s, respectively, except for the computation time of motion estimation.

Fig. 1. Detection result using Ref. [2].

Fig. 2. Detection result using our method.

Fig. 3. Detection result using our method.

4. Conclusions

In this paper, we have proposed the blotch detection algorithm with quite low computational time and with low false alarm rate. The main idea is to restrict the attention of the detecting process only to the candidate blotch areas. In addition, in order to solve the problem of the confusion with the blotches due to the poorly estimated motion vectors, the weighting function with respect to the detected moving edge pixels into the MRF model has been incorporated. The experimental result has shown a fast and efficient detecting performance of the proposed method compared with the existing MRF-based methods.

References