

# Archive Film Restoration Based on Spatiotemporal Random Walks

Xiaosong Wang and Majid Mirmehdi

Computer Vision Group, Department of Computer Science,  
University of Bristol, Bristol BS8 1UB, UK  
{wang,majid}@cs.bris.ac.uk

**Abstract.** We propose a novel restoration method for defects and missing regions in video sequences, particularly in application to archive film restoration. Our statistical framework is based on random walks to examine the spatiotemporal path of a degraded pixel, and uses texture features in addition to intensity and motion information traditionally used in previous restoration works. The degraded pixels within a frame are restored in a multiscale framework by updating their features (intensity, motion and texture) at each level with reference to the attributes of normal pixels and other defective pixels in the previous scale as long as they fall within the defective pixel's random walk-based spatiotemporal neighbourhood. The proposed algorithm is compared against two state-of-the-art methods to demonstrate improved accuracy in restoring synthetic and real degraded image sequences.

**Keywords:** Video Restoration; Random Walks; Multiscale Refinement.

## 1 Introduction

Archived films suffer damage and quality degradation often through inappropriate storage and wear and tear, but sometimes even at the time of production. The most common types of defects are blotches and scratches which usually appear in one or more (consecutive) frames as black, or white, or semi-transparent regions. However, degrees of degradation and their shape and size can vary due to their random appearance. Quality control and restoration is therefore necessary before such films are broadcastable again and indeed the preferred route to preservation and rebroadcasting is digitisation and automated restoration - a more economical and reversible process compared to the manual and tiresome course of restoration by chemical and physical means, considering the enormous amount of archives there exists. Figure 1 shows an example of a restored frame from a clip we call *Cliff*.

An automated restoration system is usually composed of two modules, defect detection and defect removal. Defect detectors such as [19,15,22,24] provide not only quantitative measures as evidence for quality control but also defect maps which can be used by others to perform defect removal. Thus in this paper, our focus is on restoration and defect removal using defect maps generated from any defect detection work, such as [24].



**Fig. 1.** A degraded frame before and after restoration by the proposed method

One approach to the recovery of a degraded pixel is to replace it with an original corresponding pixel along its projected motion trajectory from (temporal) neighbouring frames. This clearly involves an accurate estimation of the degraded pixel's motion through space and time and helps enforce a local consistency by imposing features besides just image intensities, i.e. motion vectors (leading to consistent optical flows). The chances of more accurate recovery can be increased by recruiting more significant features, e.g. texture features such as the Local Binary Pattern (LBP) [20] (leading to consistent region representation). Unlike previous methods such as [19,23,16,10], we consider multiple features *in an integrated fashion* and show that this provides better restoration than treating the features separately. The computational expense incurred due to the use of more features is an affordable tariff in our archive restoration application where accuracy is of paramount importance.

In order to locate the optimal replacement for a degraded pixel, we establish a region of candidate pixels formed by a number of 3D random walks on the *spatiotemporal* domain, starting from the defective pixel. In [9], spatial-only random walks were applied for noise reduction by taking a weighted average over all spatial pixels visited by the random walks, whereas we select the optimal pixel-exemplar as the pixel which has the maximum likelihood of being the original pixel - as defined by its intensity, motion and texture characteristics - from this dynamically generated spatiotemporal region. We perform this search-and-replace procedure for each degraded pixel in the defect map in a multiscale framework to refine the restored pixels from coarse to fine. This multiscale refinement particularly helps with large degraded regions which are forced to implode gradually through the propagation of reliable outer pixels into the region.

The contributions of our approach are therefore as follows. We present a novel pixel-exemplar based restoration algorithm using spatiotemporal random walks. In comparison to current state-of-the-art archive film restoration techniques, our method is more accurate by using more reliable statistics produced during the random walks. Also, in addition to intensity and motion features, we employ a higher order texture feature, i.e. one that is more complex than raw intensities.

Finally, degraded pixels within a frame are collectively restored in a multiscale framework by updating all their features (intensity, motion and texture), which leads to more effective searching for optimal replacements (and significantly helps in the restoration of degraded regions that are considerably larger than typical defects). This means that at each scale the attributes of a defective pixel are updated with reference to the attributes of normal pixels and other defective pixels updated in the previous higher scale as long as they fall within the defective pixel's random walk-based spatiotemporal neighbourhood. Thus, there are more constraints to contribute to the restoration of intensities.

In Sect. 3, our proposed method is presented. First, the fundamentals of 3D random walks are introduced in Sect. 3.1. Then, our restoration algorithm is described in Sect. 3.2, and is followed by the multiscale restoration scheme which is briefly reviewed in Sect. 3.3. Finally in Sect. 4, we compare and evaluate our proposed method against two state-of-the-art methods, i.e. [15] and [10], on a variety of artificially degraded and real films.

## 2 Background

The task of filling in missing regions in single or consecutive frames is often referred to as *Inpainting*, which originates from restoration in the world of Art. It was first introduced into digital image restoration by Bertalmio et al. [1] who adapted the original idea of artistic inpainting by propagating the surrounding colour and structure into the missing area. Since then, inpainting has become a popular topic in computer vision and most of the research is concentrated on mainly two directions, i.e. image structure (non-texture) propagation methods and exemplar (texture) based methods.

Examples of methods developed to recover image structure information in degraded regions are [3,8] for edges and [18,4] for level lines. These usually require complex image models with high order partial differential equations or calculus of variations. Although such methods have proven to be effective solutions to restoring small gaps in degraded images, they suffer from blurring side-effects when dealing with large missing areas, e.g. they can fail to restore textural details within the missing regions they recover.

Exemplar based inpainting methods [2,5] attempt to overcome these side-effects. In a similar fashion to texture synthesis methods, e.g. [14,6], Criminisi et al. [5] performed the propagation of textures using a block-based sampling process, pointing out that the order of the filling process is critical for achieving simultaneous recovery of image structure and texture. Wexler et al. [25] and Patwardhan et al. [21] extended the algorithm in [5] by enlarging the sampling region to a number of temporal neighbouring frames (forwards and backwards) and both methods are designed to fill in space-time holes in video sequences with stationary background and moving foreground in periodic motions. Patwardhan et al.[21] further considered scenes with restricted camera motions by including a motion segmentation procedure. In [10], instead of using a global search

as performed in [25,21], Gangal et al. limited their search region to temporal motion compensated neighbourhoods. These three methods ([25], [21], and [10]) inherited the shortcomings of [5]: (a) it is difficult to choose the optimal size of an exemplar patch, considering that a larger patch will possibly bring artifacts while a smaller one may cause mismatching, and (b) a mismatching of patches in early stages will cause an incremental effect to the detriment of the final results.

We now focus on algorithms developed to restore missing scene information in archive films specifically. There is a class of methods that have used filter-based techniques applied to the entire image regardless of a defect map, e.g. the LUM filter [13], the ML3Dex filter [17] and the SMF filter [12]. These methods are able to go a long way in eliminating the defects but result in artifacts elsewhere in the image by removing texture detail. Recently, a series of methods [19,23,16,15] have applied statistical modelling to preform the defect detection and removal stages under a single framework. As the state-of-the-art, Kokaram's Bayesian framework [15] attempted to model noise and scratches, and perform motion adjustment together. Three binary variables were used for each pixel to mark if a pixel is degraded, forward occluded or backward occluded. These variables, together with restored image intensities and motion vectors, were defined as unknowns. Given the pixel values of degraded frames and initial motion estimations, a two-stage procedure was designed to estimate the variables and image intensities first and then adjust the motion vectors according to neighbouring motion vectors, before repeating this process for a fixed number of iterations. It is worth noting that in [15], to perform motion adjustment for a degraded pixel, the method relies on the accuracy of the pixel's surrounding motion vectors. During the iterative processing, motion information is improved separately and with no reference to the improved intensities. However as stated earlier, in our work, we update the motion vectors of a defective pixel in a multiscale process with reference to all attributes (i.e. intensity, motion and texture) of normal pixels and other defective pixels (updated in the higher scale) as long as they fall within its spatiotemporal random walk-based neighbourhood.

We assume defect maps  $\mathbf{D} = \{d_{\mathbf{x}}, d_{\mathbf{x}} \in \{0, 1\}\}$  for an archive film sequence are available using any reasonably accurate defect detection algorithm, e.g. the HAFID-STC defect detector proposed by Wang and Mirmehdi [24]. The label  $d_{\mathbf{x}} = 1$  states that pixel  $\mathbf{x}$  is degraded. The method proposed in [24] first trains a Hidden Markov Model (HMM) for defect-free temporal pixel sequences across a large number of frames which is then applied to unseen temporal pixel sequences to detect defects. However, this results in a considerable number of false alarms, which are then eliminated in [24] through a two-stage removal process based on (a) MRF modelling for false alarms that have strong correlation with their neighbours and (b) localised feature tracking for those that can be traced temporally. They achieved improved results in comparison to other techniques such as [15] and [22] and hence their approach is used to generate the input to our restoration process described here, although binary defect maps from any other technique will also be applicable.

### 3 Proposed Method

Traditional pixel-exemplar or patch-exemplar based restoration methods such as [7,25,10], search for the optimal exemplar amongst a square or rectangular region of pixels using sliding windows. A novel feature of our proposed method is that for each defective pixel examined, we explore a dynamically generated, random-walk based region of candidate pixel-exemplars to select the optimal replacement from. Every pixel in this region shares a significant similarity with the previous pixels in the region as defined by their features, i.e. intensity, motion and texture. A random walk starts from a degraded pixel and stops when it reaches a strong boundary in terms of a significant change in all the pixel features. The size of the region is thus determined on-the-fly and is based on the length of all the random walks (for the current defective pixel). We perform an empirically-determined fixed number of random walks for a degraded pixel to form a region (see Sect. 4 for details).

After building the region of candidate pixel-exemplars for a degraded pixel, we assign to each of them a likelihood of being the optimal replacement for the degraded pixel. This is obtained for each pixel-exemplar by first computing the average (geometric mean) of transition probabilities during each random walk which starts from the degraded pixel and visits the pixel-exemplar. Then the averaged probabilities from these random walks are summed up to get a measure of the similarity between the pixel-exemplar and the rest of the pixel-exemplars in the region (recalling that the transition probabilities are an indication of pixel similarities in a path). The higher this value, the higher is the similarity. This is then weighted by a reliability value, which measures the degree of degradation for each pixel-exemplar, to obtain its likelihood value. The pixel-exemplar with the maximal likelihood will be selected to replace the target degraded pixel. This means that the selected pixel is the optimal representation of the spatiotemporal random walk-based region of candidates - with relatively low (to possibly no) degree of degradation - to restore the current degraded pixel. The above processing is performed in multiscale for all degraded pixels within a frame along with their reliability values, refining the updated pixels' features from coarse to fine.

#### 3.1 Preliminaries and Definitions

Next, we state the fundamentals of a 3D random walk on an image sequence and then express the probability of a random walk sequence in the context of our application. We define the input image sequence as an undirected and weighted graph  $G = (V, E)$  with vertices (nodes)  $v_x \in V$  and edges  $e_{\mathbf{x}', \mathbf{x}''} \in E \subseteq V \times V$ . Each edge  $e_{\mathbf{x}', \mathbf{x}''}$  is assigned a weight  $w_{\mathbf{x}', \mathbf{x}''}$  where  $w_{\mathbf{x}', \mathbf{x}''} > 0$  and  $w_{\mathbf{x}', \mathbf{x}''} = w_{\mathbf{x}'', \mathbf{x}'}$ . An image pixel  $\mathbf{x}$  at location  $(i, j, t)$  ( $1 \leq i \leq Width, 1 \leq j \leq Height, 1 \leq t \leq Length$ ) is represented as a node  $v_x (v_x \in V)$  in graph  $G$  where  $Width \times Height \times Length$  defines the image sequence volume.

A random walk sequence  $Path_{0,K} = \{\mathbf{x}^0, \mathbf{x}^1, \dots, \mathbf{x}^K\}$  with length  $K + 1$  on graph  $G$  is specified as a sequence of nodes (pixels) which is a Markov process.

The probability of the transition  $p(\mathbf{x}^k|\mathbf{x}^{k-1})$  between consecutive pixels  $\mathbf{x}^{k-1}$  and  $\mathbf{x}^k$  is given as the weight  $w_{\mathbf{x}^{k-1},\mathbf{x}^k}$  on the edge  $e_{\mathbf{x}^{k-1},\mathbf{x}^k}$ . According to the Markov property of  $Path_{0,K}$ , the probability of a  $Path_{0,K}$  starting at pixel  $\mathbf{x}^0$  is defined as

$$p(Path_{0,K}) = \prod_{k=1}^K p(\mathbf{x}^k|\mathbf{x}^{k-1}) = \prod_{k=1}^K w_{\mathbf{x}^{k-1},\mathbf{x}^k}, \tag{1}$$

where  $P = \{Path_{0,K_m}^m\}_{m=1}^M$  is a set of  $M$  random walks on graph  $G$ , with each walk starting from  $\mathbf{x}^0$ . Furthermore, we define the region of candidates or pixel-exemplars  $\mathbf{R}_{\mathbf{x}^0} = \bigcup_{m=1}^M Path_{0,K_m}^m$  as the set of all pixels visited by the random walks in  $P$ . The neighbourhood for a pixel, the associated edge weights, and the walk length are expressed as follows:

*Neighborhood  $N_{\mathbf{x}}$* : For each pixel  $\mathbf{x}$  on a walk, we define a  $3 \times 3 \times 3$  spatiotemporal motion compensated neighbourhood  $N_{\mathbf{x}}$  centred at  $\mathbf{x}$ . In  $N_{\mathbf{x}}$ , we denote the connection between pixel  $\mathbf{x}$  and  $\mathbf{x}' (\mathbf{x}' \in N_{\mathbf{x}}, \mathbf{x}' \neq \mathbf{x})$  as edge  $e_{\mathbf{x},\mathbf{x}'}$  with a weight  $w_{\mathbf{x},\mathbf{x}'}$ . For each step in a random walk, a transition from the current pixel  $\mathbf{x}$  to one of its 26 direct neighbours  $\mathbf{x}' (\mathbf{x}' \in N_{\mathbf{x}})$  is permitted.

*Edge Weights*: In the same fashion as previous graph-based methods, e.g. [11], the edge weights are defined by a function that evaluates the similarity of two consecutive pixels during a random walk so as to bias it to stop the walk when a significant decrease in similarity is observed. Here, we define edge weights as the probability of pixels  $\mathbf{x}$  and  $\mathbf{x}'$  being identical, measured by using a number of different pixel features,

$$w_{\mathbf{x},\mathbf{x}'} = \frac{1}{T} \prod_{q=1}^Q \exp\left\{-\frac{\varphi_q^2(\mathbf{x}, \mathbf{x}')}{2\sigma_q^2}\right\}, \tag{2}$$

where  $T$  is a normalization constant,  $\sigma_q$  is the standard deviation for pixel feature  $q$ , and  $\varphi_q(\cdot)$  measures the Euclidean distance between pixel  $\mathbf{x}$  and  $\mathbf{x}'$  in feature space  $\mathfrak{F}_q$ . A variety of pixel features can be used to measure the similarity between two pixels and here we apply four (i.e.  $Q = 4$ ); these are intensity, forward and backward motion, and the local LBP texture pattern:

$$\varphi_q^2(\mathbf{x}, \mathbf{x}') = \frac{1}{J_q} \sum_{j=1}^{J_q} (Z_q^j(\mathbf{x}) - Z_q^j(\mathbf{x}'))^2, \tag{3}$$

where  $Z_q = \{\mathbf{I}, \mathbf{V}_{\mathbf{x}}^f, \mathbf{V}_{\mathbf{x}}^b, \mathbf{L}\}$  for  $q = \{1..4\}$ ,  $\mathbf{I}$  represents RGB intensity maps with  $J_1 = 3$ ,  $\mathbf{V}_{\mathbf{x}}^f$  and  $\mathbf{V}_{\mathbf{x}}^b$  represent forward and backward motion vector maps with  $J_2 = J_3 = 2$  respectively, and  $\mathbf{L}$  represents maps of 2D image LBP patterns in a spatial  $3 \times 3$  neighbourhood with  $J_4 = 8$ . The addition of a texture feature, along with a more integrated contribution of all the features used through (3), and subsequently (5), is an essential improvement on other works in archive film restoration, such as [15] and [10]. The extra texture feature is specifically appropriate to enforce a constraint in textured regions to help select the pixels that can be included in the region of candidates during the random walks.

*Walk Length.* We control the length of a random walk by monitoring  $p(Path_{0,K})$  in the same manner as proposed in [9]. Since we are performing a biased random walk by encouraging transitions between similar neighbours, the random walk will be terminated if  $p(Path_{0,K})$  is smaller than a threshold. This will prevent random walks from stepping across strong boundaries in terms of significant changes of all pixel features. A walk will also be terminated if it hits a hard boundary, i.e. the image boundaries on the spatial domain and the first and last frames on the temporal axis.

### 3.2 Restoration of Degraded Pixels

We restore all pixel features of a degraded pixel by replacing the degraded pixel with the optimal pixel-exemplar selected from its region of candidates, which has the maximal likelihood of being the original pixel. The selection procedure is as follows.

For each pixel-exemplar  $\mathbf{x}$  in a degraded pixel’s region of candidates, i.e.  $\mathbf{R}_{\mathbf{x}^0}$ , the similarity between  $\mathbf{x}$  and the rest of the pixel-exemplars in the region is measured based on the probabilities of random walk paths which start from the degraded pixel and visit the pixel-exemplar, represented as

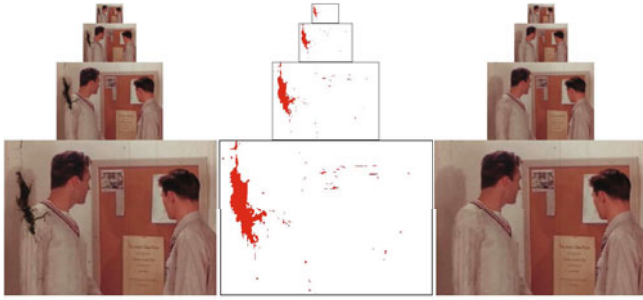
$$A_{\mathbf{x}} = \sum_{m=1}^M \sum_{k=1}^{K_m} \left( p(Path_{0,k}^m)^{1/k} \cdot \delta(\mathbf{x}^k = \mathbf{x}) \right), \tag{4}$$

where  $\delta(\cdot)$  is the Dirac *delta* function. In order to measure the similarity among all pixel-exemplars in a random walk path regardless of the length of the path, we compute the geometric mean of their transition probabilities. Provided we perform a sufficient number of spatiotemporal random walks, the sum of their averaged probabilities suggests the similarity between the pixel-exemplar and the rest of the pixel-exemplars in the region. This is influenced by the way spatial random walks are used in [9] to examine the transition probabilities (i.e. similarity) of pixels along a path in their image denoising application. The reason why we use this value instead of using other measurements, e.g. a count of random walks that visit the pixel-exemplar, is because this value indicates if the pixel-exemplar provides random walks with a smooth transition from their previous locations to this pixel-exemplar, e.g. if the probabilities of random walk paths decrease significantly after they visit this pixel-exemplar, this value will be probably small even though this pixel-exemplar has been visited by a large number of random walks. The optimal pixel-exemplar is then selected as

$$\hat{\mathbf{x}}^0 = \arg \max_{\mathbf{x} \in \mathbf{R}_{\mathbf{x}^0}} (A_{\mathbf{x}} \cdot \mathbf{r}(\mathbf{x})), \tag{5}$$

where  $\mathbf{r}(\cdot)$  indicates the reliability of a pixel-exemplar based on its degree of degradation. For normal pixels,  $\mathbf{r}(\cdot)$  is 1 while a degraded pixel is initialised to the likelihood of being identical to all its defect-free neighbours in  $N_{\mathbf{x}}$ :

$$\mathbf{r}(\mathbf{x}) = \begin{cases} \frac{1}{\sum_{\mathbf{x}' \in N_{\mathbf{x}}} \delta(d_{\mathbf{x}'})} \sum_{\mathbf{x}' \in N_{\mathbf{x}}} w_{\mathbf{x}\mathbf{x}'} \delta(d_{\mathbf{x}'}) & d_{\mathbf{x}} = 1 \\ 1 & d_{\mathbf{x}} = 0 \end{cases} \tag{6}$$



**Fig. 2.** (from left) A sample image pyramid, the defect map pyramid, and the restored results using the proposed method. The degraded regions are gradually recovered from coarse to fine and from the boundaries to their inner part.

Note also that by this definition, a false alarm pixel is more likely to be initialised with a high  $\mathbf{r}(\cdot)$  value, given it is likely to be more similar to its defect-free spatiotemporal neighbours than to real degraded pixels. After a degraded pixel is replaced with a specific pixel-exemplar, its reliability value is updated with:

$$\hat{\mathbf{r}}(\mathbf{x}) = \frac{1}{|N_{\mathbf{x}}|} \sum_{\mathbf{x}' \in N_{\mathbf{x}}} \mathbf{r}(\mathbf{x}'). \quad (7)$$

During the multiscale updating algorithm (reviewed next), the  $\mathbf{r}(\cdot)$  value for a degraded pixel will approach 1 after a number of updates. For a degraded pixel near the boundary of a degraded region, the  $\mathbf{r}(\cdot)$  value will reach 1 faster than an inner pixel considering it is surrounded by more reliable spatiotemporal neighbours (normal pixels). Thus, during the multiscale refinement, a degraded region will gradually implode through the propagation of more reliable outer pixels in the region. For an example see Fig. 2.

Additionally, since a pixel in a false alarm region may be initialized with a larger  $\mathbf{r}(\cdot)$  value (as noted above) than a real degraded pixel, then the false alarm pixel is more likely to obtain an optimal replacement considering more reliable candidates are present in its random walk-based neighbourhood.

### 3.3 Multiscale Refinement

Given an image sequence and its defect map, we build pyramids for each frame and its corresponding defect map by downsampling the original by a factor of 2 after smoothing with a  $5 \times 5$  Gaussian kernel. A sample image pyramid and its associated defect pyramid are shown in Fig. 2. After restoring the degraded pixels' features on a current level of the pyramid, we upsample these pixels to the next level and then update their corresponding pixels' features in that level. This level-by-level refinement and restoration process continues until it reaches the lowest level of the pyramid (see Algorithm 1).



**Algorithm 1.** The multiscale restoration algorithm

---

```

Build pyramids  $\{\mathbf{I}_s\}_{s=1}^S$  and  $\{\mathbf{D}_s\}_{s=1}^S$ ;
Initialize scale  $s = 1$ : Compute motion vector maps  $\mathbf{V}_1^f, \mathbf{V}_1^b, \mathbf{L}_1$ , and  $\mathbf{r}_1$ ;
while  $s \leq S$  do
  if  $s > 1$  then
    Update  $\mathbf{I}_s, \mathbf{V}_s^f, \mathbf{V}_s^b, \mathbf{L}_s$  by  $\mathbf{I}_{s-1}, \mathbf{V}_{s-1}^f, \mathbf{V}_{s-1}^b$ ; /*Only on degraded sites*/
  end if
   $(\mathbf{I}_{s+1}, \mathbf{V}_{s+1}^f, \mathbf{V}_{s+1}^b, \mathbf{L}_{s+1}) = \text{Restoration}(\mathbf{I}_s, \mathbf{V}_s^f, \mathbf{V}_s^b, \mathbf{L}_s, \mathbf{D}_s)$ ;
  Update  $\mathbf{r}_{s+1}$  using equation (7);
  if  $s < S$  then
    Upsample  $\mathbf{I}_{s+1}, \mathbf{V}_{s+1}^f, \mathbf{V}_{s+1}^b, \mathbf{r}_{s+1}$  by factor 2; /*For every scale but the last*/
  end if
   $s = s + 1$ ;
end while

```

---

## 4 Experimental Results and Discussion

We present the restoration performance of the proposed algorithm on both artificially degraded and real sequences, and compare our results against two state-of-the-art techniques: Kokaram’s Bayesian framework [15] and Gangal and Dizardoglu’s exemplar-based method [10], hereafter referred to as Kokaram04 and GD06 respectively. The defect maps for both GD06 and the proposed method were produced in advance using the HAFID-STC defect detector [24] while Kokaram04 has an integrated defect detector. All methods were tuned for optimal performance using constant parameter values across all experiments.

**Synthetic defects** - The proposed method was compared against Kokaram04 and GD06 on restoring five artificially degraded real sequences totalling 1500 frames, namely *Mobile Calendar*, *Container*, *Foreman*, *News* and *Paris*. The degraded sequences were produced by adding synthetic black and white defects of sizes of between 1 and 6000 pixels on a random basis. For each method, the Mean Square Error (MSE) to measure the difference between the original defect-free frame  $F$  and the restored frame  $\hat{F}$  was computed:

$$\text{MSE}(F, \hat{F}) = \frac{1}{\text{Width} \times \text{Height} \times 3} \sum_{\mathbf{x} \in F} \sum_{i=1}^3 (F_{\mathbf{x}}^i - \hat{F}_{\mathbf{x}}^i)^2 \quad (8)$$

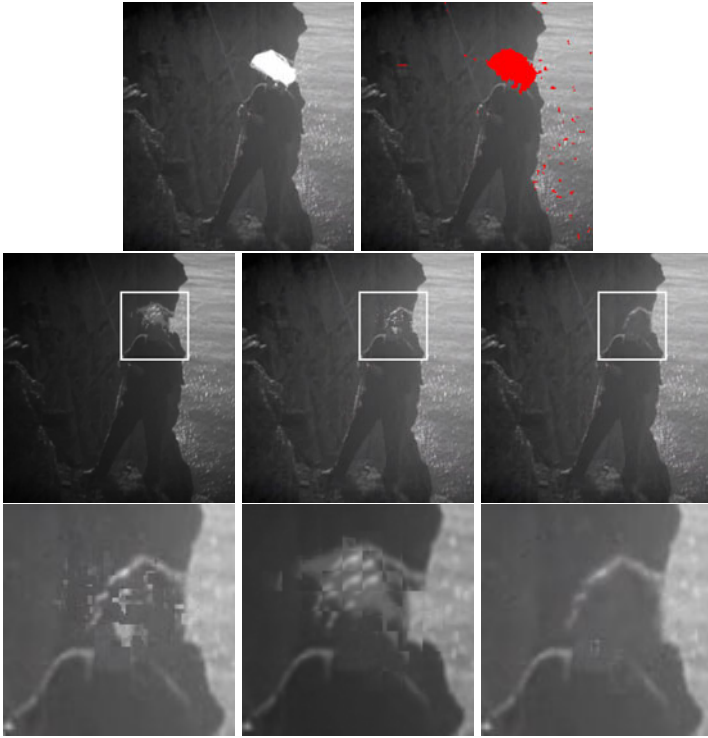
Columns 2 through to 5 in Table 1 show the MSEs for four randomly selected sample frames from the *Mobile Calendar*, *Container*, *Foreman*, *News* and *Paris* sequences respectively. The percentage of degraded pixels in each frame is listed along with the frame number along the top row. The raw, unrestored frame error rate is shown along the ‘Degraded’ row in each case. The last column in Table 1 shows the average MSEs across all frames in each of the synthetic-error sequences for each method; for example for the *Foreman* sequence, given the average true MSE rate of 153.4, the proposed method resulted in the lowest error at 44.7 compared to Kokaram04 and GD06 at 130.3 and 103.1 respectively. The proposed method performed much better in all the experiments, avoiding

the creation of too many artifacts (see e.g. Fig. 3) compared to Kokaram04 and GD06, and was also more capable of restoring large defects (e.g. Frame 233 of the *News* sequence). Note the introduction of artifacts during restoration by all three methods, may lead to MSE errors that are larger than the raw original whose MSE is only based on synthetic defects.

**Real defects** - We compared the three methods on restoring a variety of real degraded image sequences, including greyscale and colour, indoor and outdoor scenes, and slow and fast motions, and in all cases the proposed method produced the best results. In the following, three sets of sample results are illustrated to inspect three aspects of the proposed method, i.e. recovering a large degraded region and substantially avoiding artifacts in Fig. 3, handling defect-free (false alarm) pixels in Fig. 4, and correcting motions in Fig. 5.

**Table 1.** Comparison of MSEs on real sequences with synthetic errors

<i>Mobile Calendar</i>					
Frame #	32 (0.07%)	58 (0.11%)	181 (3.47%)	233 (0.03%)	Avg (0.62%)
Degraded	16.3	31.8	651.5	56.6	183.5
Kokaram04	210.6	89.9	293.6	92.5	157.3
GD06	128.7	80.7	196.8	123.5	135.9
Proposed	23.4	19.5	105.9	46.7	49.2
<i>Container</i>					
Frame #	9 (0.06%)	23 (0.21%)	138 (2.44%)	210 (0.06%)	Avg (0.60%)
Degraded	15.7	80.7	451.1	15.7	119.1
Kokaram04	7.6	3.5	93.4	2.8	33.9
GD06	0.9	0.8	96.3	3.2	23.8
Proposed	0.6	0.6	45.1	3.3	10.5
<i>Foreman</i>					
Frame #	33 (0.12%)	65 (0.24%)	98 (0.23%)	199 (0.02%)	Avg (0.55%)
Degraded	29.7	45.8	40.8	5.3	153.4
Kokaram04	70.5	155.32	89.84	74.0	130.3
GD06	95.3	149.12	95.87	68.5	103.1
Proposed	21.4	51.38	40.37	10.7	44.7
<i>News</i>					
Frame #	113 (0.08%)	165 (0.05%)	233 (2.47%)	291 (0.21%)	Avg (0.61%)
Degraded	31.5	15.2	751.5	66.9	154.5
Kokaram04	159.9	218.3	289.1	89.5	140.7
GD06	140.7	125.7	205.3	113.5	125.3
Proposed	49.5	27.7	119.8	54.4	54.2
<i>Paris</i>					
Frame #	81 (0.16%)	123 (0.01%)	158 (1.34%)	280 (0.10%)	Avg (0.60%)
Degraded	80.7	15.7	537.1	85.3	122.1
Kokaram04	30.5	25.6	107.1	11.3	63.9
GD06	21.8	27.9	122.3	15.4	83.8
Proposed	13.4	14.3	69.3	6.3	33.5



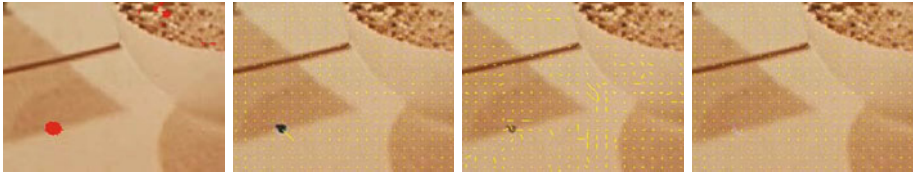
**Fig. 3.** *Cliff* - Comparing large missing area recovery. Top: original frame and the defect map in red; Middle: restoration results from Kokaram04, GD06, and the proposed method; Bottom: enlargement of selected areas.

Fig. 3 shows the results on a sample degraded frame with a large missing area. The original frame and the defect map (in red) are shown in the top row. The results by Kokaram04, GD06, and the proposed method are in the middle row and a close-up of the degraded area is shown in the bottom row. Kokaram04 results in a considerable number of artifacts in the restored frame because its performance strongly depends on the accuracy of motion information. Its motion correction procedure is not designed for such large missing areas, but rather for small degraded areas with accurate motion information provided in their spatial neighbouring regions. While GD06 is able to restore the outline of the man's head, it introduces some artifacts in the inner region due to the mismatching of patches in an early stage. Although the proposed method still causes some small artifacts, both the image structure and texture are recovered well.

In Fig. 4 we investigate the restoration performance of the three methods on handling false alarm pixels. A sample degraded frame and its corresponding defect map are in the top row, and restoration results from Kokaram04, GD06 and the proposed method follow in the bottom row. In this example all methods do well in restoring the real degraded pixels. However, both Kokaram04 and



**Fig. 4.** *Policeman* - Comparing restoration of false alarms. Top: original and its defect map; Bottom: restoration results from Kokaram04, GD06, and the proposed method.



**Fig. 5.** *Coffee* - Comparing motion correction. (from left) Original frame with overlaid defects, Original motion vectors, Corrected motion vectors from Kokaram04 and the proposed method.

GD06 lose considerable detail, e.g. the policeman’s hand is missing in the frame restored by Kokaram04, and artifacts are introduced across the telephone and the policeman’s hand in the frame restored by GD06.

The final example presents a comparison between Kokaram04 and the proposed method on correcting motions for degraded pixels. The motion vectors overlaid on an original frame are shown in the second image from left in Fig. 5. The correction results from Kokaram04’s integrated motion correction algorithm and the proposed method follow this respectively. During Kokaram04’s iterative process, motion information is improved separately and with no reference to the improved intensities; this means its correction is limited by the accuracy of initial motion estimations which are often inaccurate by the presence of defects. The proposed method outperforms Kokaram04 by achieving more accurate motion correction for each defective pixel by reference to their spatiotemporal random-walk neighbours through the multiscale process.

**Performance and Implementation Issues** - All methods were implemented in MATLAB on a laptop with Intel Core Duo 2.4 GHz and 2GB RAM. The average speed for a degraded frame of average size of  $480 \times 360$  was 406 seconds for our proposed method, while Kokaram04 and GD06 needed 174 and 265 seconds respectively. Our proposed algorithm is slower but more accurate than Kokaram04 and GD06, since it considers an extra feature and requires considerable sampling by the random walks. We experimented with different values of  $M$  by using different random seeds for each random walk.  $M = 800$  was found to provide stable results and reasonable computing costs. The number of steps in each random walk often varied from 2 to 48. Since accuracy is critical for the restoration of archive films, the extra computational burden is a tolerable cost.

## 5 Conclusion

We presented a novel pixel-exemplar based restoration algorithm using spatiotemporal random walks. The random walks are formed by considering pixel similarities using multiple features. The method is applicable given a defect map generated by any archive film defect detection algorithm. While the use of multiple features adds to our computational costs, we obtain much more accurate and artifact-free results than current state-of-the-art techniques.

**Acknowledgments.** We thank Great Western Research, ITV and Bristol University for funding this project.

## References

1. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proc. SIGGRAPH, pp. 417–424 (2000)
2. Bornard, R., Lecan, E., Laborelli, L., Chenot, J.H.: Missing data correction in still images and image sequences. In: Proc. MULTIMEDIA, pp. 355–361 (2002)
3. Chan, T.F., Kang, S.H., Shen, J.: Euler’s elastica and curvature driven diffusions. *SIAM J. App. Math.* 2, 564–592 (2002)
4. Chan, T.F., Shen, J.: *Image Processing and Analysis*. SIAM, Philadelphia (2005)
5. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. IP* 13(9), 1–13 (2004)
6. Efros, A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: Proc. SIGGRAPH, pp. 341–346 (2001)
7. Efros, A., Leung, T.: Texture synthesis by non-parametric sampling. In: Proc. ICCV, vol. 2, pp. 1033–1038 (1999)
8. Esedoglu, S., Shen, J.: Digital inpainting based on the mumford-shah-euler image model. *Euro. J. App. Math.* 13(4), 353–370 (2002)
9. Estrada, F., Fleet, D., Jepson, A.: Stochastic image denoising. In: Proc. BMVC (2009)
10. Gangal, A., Dizdaroglu, B.: Automatic restoration of old motion picture films using spatiotemporal exemplar-based inpainting. In: Blanc-Talon, J., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2006. LNCS, vol. 4179, pp. 55–66. Springer, Heidelberg (2006)

11. Grady, L.: Random walk for image segmentation. *IEEE Trans. PAMI* 28(11), 1768–1783 (2006)
12. Hamid, M.S., Harvey, N., Marshall, S.: Genetic algorithm optimization of multi-dimensional grayscale soft morphological filters with applications in film archive restoration. *IEEE Trans. CSVT* 13(5), 406–416 (2003)
13. Hardie, R., Boncelet, C.: Lum filters: a class of rank-order-based filters for smoothing and sharpening. *IEEE Trans. SP* 41(3), 1061–1076 (1993)
14. Heeger, D.J., Bergen, J.R.: Pyramid-based texture analysis/synthesis. In: *Proc. SIGGRAPH*, pp. 229–238 (1995)
15. Kokaram, A.: On missing data treatment for degraded video and film archives: a survey and a new bayesian approach. *IEEE Trans. IP* 13(3), 397–415 (2004)
16. Kokaram, A., Godsill, S.: Mcmc for joint noise reduction and missing data treatment in degraded video. *IEEE Trans. SP* 50(2), 189–205 (2002)
17. Kokaram, A., Morris, R., Fitzgerald, W., Rayner, P.: Interpolation of missing data in image sequences. *IEEE Trans. IP* 4(11), 1509–1519 (1995)
18. Masnou, S., Morel, J.M.: Level lines based disocclusion. In: *Proc. ICIP*, vol. 3, pp. 259–263 (1998)
19. Morris, R.: Image Sequence Restoration using Gibbs Distributions. Ph.D. thesis, Cambridge University (1995)
20. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE trans. PAMI* 24(7), 971–987 (2002)
21. Patwardhan, K., Sapiro, G., Bertalmio, M.: Video inpainting of occluding and occluded objects. In: *Proc. ICIP*, vol. 2, pp. II: 69–72 (2005)
22. Ren, J., Vlachos, T.: Efficient detection of temporally impulsive dirt impairments in archived films. *Signal Processing* 87(3), 541–551 (2007)
23. Roosmalen, P.M.B.V.: Restoration of Archived Film and Video. Ph.D. thesis, Delft University of Technology (1999)
24. Wang, X., Mirmehdi, M.: HMM based archive film defect detection with spatial and temporal constraints. In: *Proc. BMVC* (2009)
25. Wexler, Y., Shechtman, E., Irani, M.: Space-time video completion. In: *Proc. CVPR*, vol. 1, pp. 120–127 (2004)