Matte-Based Restoration of Vintage Video
Jiading Gai and Sing Bing Kang, Senior Member, IEEE

Abstract—In this paper, we describe a new method for restoring digitized vintage video with film wear artifacts. Such artifacts result in partially or completely missing information. To maximize use of observed data, we cast the problem as that of recovering mattes of artifacts. More specifically, we extract the distributions of artifact color and its fractional (alpha) contribution to the frame. To account for spatial color discontinuity and pixel occlusion or disocclusion, we introduce the alpha-modulated bilateral filter. The problem is solved as a 3-D spatio-temporal conditional random field (CRF) with artifact color and (discretized) alpha as states. Inference is done through belief propagation. Results verify the effectiveness of our method. Furthermore, we can produce a synthetically generated vintage footage using extracted artifact information from actual vintage video.

Index Terms—Artifact removal, belief propagation, bilateral filter, conditional random field, matting and compositing, video restoration.

I. INTRODUCTION

Film archives are worth preserving because of their historical and cultural values. However, the process of digitizing film is a very challenging one, mostly because of film deterioration over time, making it very fragile. Film deterioration and its attendant artifacts are in turn caused by aging and chemical decomposition, and improper storage and handling. The technology of preservation and (manual) restoration of archive film is described in [19].

Film artifacts can be categorized as film wear (scratches, film hair, blotches, dust, line jitter), luminance-chrominance (color grading, color fading, and luminance flicker), image instability, loss of resolution, and noise contamination. In this paper, we address the specific problems of film wear, and to some extent, noise contamination. Please note that we differentiate between film wear artifacts and sensor-based noise (which is popularly referred to as just “noise”), as their models are different. Noise in the conventional sensor-based sense is usually modeled with an additive Gaussian distribution with zero mean, while film wear artifacts tend to be arbitrary (in the worst case, completely obliterates the data).

II. PRIOR WORK

Some of the most common artifacts in vintage video are blotches and scratches. Most approaches assume the artifact areas have been totally corrupted; such areas are identified and replaced by pixels in neighboring frames. In addition, such approaches assume artifacts can be identified in the current frame using only the previous and next frames.

Assuming the missing areas have been correctly identified, techniques have been proposed to fill them. Storey [21] used a three-tap median operation to interpolate the missing pixels. Kokaram et al. [13] extended this idea by introducing a 3-D median filtering operation on a 3 × 3 × 3 motion compensated pixel volume around each missing pixel. Roosmalen et al. [23] combine autoregressive models and Markov random field (MRF) techniques for interpolating missing data. Motivated by the observation that image sequences are usually redundant enough for simple operations, their algorithm copies pixels from neighboring frames directly instead of interpolating the missing pixels using estimates given by AR models. A non-linear interpolator has also been used to fill in the missing data. Khriji et al. [9] use a rational function filter to fill in missing pixels using neighboring information. Their method assumes a perfect defect localization algorithm.

Nadenau and Mitra [16] use rank ordered differences between the current frame and its previous and next (motion-compensated) frames to detect small blotch and scratch artifacts. These artifacts are assumed to completely obliterate the image data, and only one synthetic example was shown. Ren and Vlachos [20] took the interesting approach of segmentation to detect dirt.

Zhou and Lin [25] formulate a dust-lens model to guide the dust removal from single images. With the help of this artifact formation model, they are able to use the partial image information within the dust region directly as guides when filling the corrupted regions. Unfortunately, artifacts in vintage film typically do not follow such a model.

The line scratch artifact is usually seen as a vertical bright or dark line, and is caused by film abrasion. By noticing that line scratches usually represent as local minima, Buisson et al. [4] use morphological filters to detect and remove line scratches. In the approach described in [8], a simple 1-D-extrema detector is used to find line scratches (with the temporal persistence assumption). In [10], line artifacts are detected by generating a median smoothed version of the image and thresholding the difference between it and the original image. Bruni et al. [3] model the visibility of a line scratch as a 1-D function across it. Adding such prior information reduces the ill-posed nature of image restoration. However, models used by such methods are hard to generalize to other artifacts.

Although some film wear artifacts result in completely unusable pixels, some others result in partially degraded pixels.
In the latter case, some of the original image colors can be observed. An example of this is shown in Fig. 1. This source of information provides an important constraint on artifact removal that enables a restored image to closely match the actual scene. In our work, we make use of as much information that is available in the digitized video for restoration.

To better exploit the information inside a corrupted region, we need to model the transparency nature of film artifacts. What complicates the digital restoration process is the generative models for different artifacts (e.g., lines versus blotches) are likely to be different. It may not even be clear what these generative models are exactly; for example, existing models for line scratches [3] are likely to be just approximations. We opted for a more general but powerful matting model [18] to render the restoration problem across different types of artifacts tractable. In our matting model, each pixel in the corrupted frames is assumed to be the linear interpolation of clean pixel and artifact colors. A mixing factor $\alpha$ is assigned at every pixel to control the extent of coverage of each component. From experiments, we show that this matting model is capable of producing good results.

III. OUR APPROACH TO VIDEO RESTORATION

Current approaches for joint noise reduction and artifact removal (e.g., [7], [11], and [12]) typically restrict the estimation of artifact mask to binary, i.e., each pixel is roughly marked as corrupted or not corrupted. However, in many cases, the artifacts do appear as mixed pixels—degraded regions generally contain some image information. The partial information that may be available in artifact regions provides an important constraint for artifact removal.

Kokaram [12] assumes first-order dependence on the bidirectional temporal neighborhoods. The immediate neighbors are treated as clean frames throughout the entire restoration process and used to fill in missing pixels in the current frame. Artifacts that persist for more than a frame will be treated as legitimate data.

Our matting approach is also related to the model proposed in [11]. There, the observed degraded pixels are modeled as a binary mixture between the original pixel and a continuous degradation color. The transparent nature of the film artifact is handled by the continuous nature of the degradation. Although modeling the degradation color is a very convenient way to incorporate transparent artifacts into consideration, the size of the label space is enormous. Instead, we propose to make the mixture nonbinary and discretize the degradation. Our model is simpler because modeling the alpha matte is much easier than modeling the degradation since the degrees of freedom are decreased considerably.

Another relevant work is that of Chen and Tang [5]. Here, they estimate and remove digitization noise from video using 3-D spatio-temporal MRF. However, the digitization noise they deal with is assumed to be additive and having an intensity-independent pdf. The artifacts we are trying to remove have arbitrary multiplicative and additive elements as well as irregular shapes.

As mentioned earlier, we propose to restore videos by treating the problem as that of matting [18]: the input video is a result of artifact contamination that is spatio-temporally varying. However, there are fundamental differences from the conventional matting problem. First, we are not separating foreground from background, but rather separate actual colors from contaminated ones. Second, rather than having the user prespecify areas where fractional contributions occur (e.g., as in [24]), our technique automatically computes their full frame distribution.

The single frame version of our technique is depicted in Fig. 2. We do not assume any artifact appears in one frame only. Note that corrupted regions usually do not persist in the same place in two consecutive frames. However, artifacts caused by mechanical devices tend to persist over a few frames. For example, the line scratch is a typical defect caused by contacts with mechanical parts of film projector during the film development process, and the projector usually runs over several frames of a film and causes the scratch to extend over successive frames. We solve using 3-D conditional random field (CRF) regularized by spatial-temporal consistency on both artifacts and clean pixels together with the artifact matte formation model; each pixel’s alpha and artifact color are estimated by propagating messages along spatial and temporal neighborhoods.

The following sections describe the joint use of artifact color and $\alpha$. Also described are the 3-D CRF formulation for video restoration, the estimation of all the unknowns from the input video, and the recovery of the clean video with motion compensation.
A. Joint Estimation of Artifact Color and Alpha

Given \( N \) (typically 5 in our work) consecutive frames in video \( G \), our goal is to find an optimal linear combination of the true (clean) video \( P \) and color artifact \( A \), together with a matte \( \alpha \) in \([0,1]\), so that

\[
G = \alpha P + (1 - \alpha)A \tag{1}
\]

or, on a per pixel basis

\[
g_x = \alpha p_x + (1 - \alpha) a_x \tag{2}
\]

with \( x \) being a 3-tuple indexing space and time. So, the true pixel color is given by

\[
p_x = \frac{1}{\alpha_x} g_x = \frac{1}{\alpha_x} (1 - \alpha_x) a_x \tag{3}
\]

Before we describe our restoration inference framework, we first define, from (3)

\[
g_x = \alpha_p x + \alpha (1 - \alpha) a_x \tag{4}
\]

as the hypothesized \textit{alpha-premultiplied} true color. It can be computed given the hypotheses of \( \alpha_x \) and \( a_x \) without getting unstable (due to division by zero). \( g_x \) will become useful later in both the 3-D spatio-temporal CRF and its inference method.

B. Spatio-Temporal CRF

The overview of our approach is shown in Fig. 3. Motion estimation is done only if the scene is known to have relatively significant motion (greater than 1 pixel between frames). If motion estimation is not required, video restoration is done only at the last step; the iteration involves only refinement of artifact color and alpha distributions.

To extract the artifact color and alpha distributions, we treat our input video as a spatio-temporal CRF, with each pixel as a node. The next section details our approach to estimating these unknown distributions.

We wish to recover the clean frames \( P \), alpha maps \( \alpha \), artifact color maps \( A \) and motion field \( D \), given corrupted frames \( G \). (\textit{Note that in our work, we assume that the artifact colors are known but not their spatial-temporal distributions}.)

Our problem can be generally formulated as \( p(p, A, \alpha, D | G) \), assuming that the prior \( p(G) \) is generally unknown and can be set to a constant.

Unfortunately, \( p(P, A, \alpha, D | G) \) is generally intractable. Instead, we extract \( \alpha \) and \( A \) given \( G \) and \( D \), i.e., we wish to maximize \( p(A, \alpha | G, D) \). This has the advantage of recovering from fewer number of states than, say, computing the pixel colors directly. Once \( \alpha \) and \( A \) have been computed, \( P \) can be recovered using the algorithm described in Section III-D. We assume that \( D \) is initially known; we optionally refine it using the latest estimate of \( P \) (Section III-E).

Because we extract \( \alpha \) and \( A \) (and only indirectly \( P \)), we rely on the observed data for evaluation. We model the posterior distribution \( p(A, \alpha | G, D) \) of an \( N \)-frame video (\( N \geq 5 \) in our experiments) as a Gibbs field (which qualifies it as a conditional random field \cite{1, 14})

\[
- \log p(A, \alpha | G, D) \propto w_u \sum_u U + w_b \sum_b B(G) \tag{5}
\]

with \( U \) being the bias on alpha and \( B \) being the smoothness potential. \( w_u \) and \( w_b \) are their respective weights. (Note that the smoothness potentials depend on the observed data \( G \).) We now describe these two terms \( U \) and \( B \).

\textbf{Bias on \( \alpha \)}: The bias term for \( \alpha \) is

\[
w_u \sum_u U = w_1 E_1 = w_1 \sum_x (1 - \alpha_x). \tag{6}
\]

It prevents the solution from converging to a degenerate one.

\textbf{Smoothness potential}: These potentials were designed with the assumption that there is spatio-temporal regularity. We define four potentials

\[
w_b \sum_b B(G) = w_2 E_2 + w_3 E_3 + w_4 E_4 + w_5 E_5 \tag{7}
\]

with \( E_2 \) and \( E_3 \) encouraging the clean frames to be consistent spatially and temporally (respectively). Ideally, to measure the spatial and temporal consistency among clean frames, we
should be comparing the hypothesized true colors $\mathbf{p}_\mathbf{x}$ and $\mathbf{p}_\mathbf{y}$ by taking their difference: $\mathbf{p}_\mathbf{y} - \mathbf{p}_\mathbf{x} = (q_\mathbf{y})/(\alpha_\mathbf{y}) - (q_\mathbf{x})/(\alpha_\mathbf{x})$. However, to avoid division by small (noisy) $\alpha$’s, we premultiply this term by $\alpha_\mathbf{y}/\alpha_\mathbf{x}$ to yield the difference term $\alpha_\mathbf{y}/\alpha_\mathbf{x}(\mathbf{p}_\mathbf{y} - \mathbf{p}_\mathbf{x})$ as used below in (8) and (9). Recall from (4) that $\alpha$ is the alpha-premultiplied color, so that the true color $p = q/\alpha = (q-(1-\alpha)a)/\alpha$. Hence, $E_2$ and $E_3$ both rely on the observed pixel data $g$. $E_4$ encourages $\alpha$ to be continuous, and $E_5$ encourages the artifact color distribution $A$ to be continuous. The terms $E_2$, $E_3$, $E_4$, and $E_5$ are defined as

$$E_2 = \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{x}/\alpha_\mathbf{y} - \alpha_\mathbf{y}/\alpha_\mathbf{x}) \quad (8)$$

$$E_3 = \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{x}/\alpha_\mathbf{y} - \alpha_\mathbf{y}/\alpha_\mathbf{x}) \quad (9)$$

$$E_4 = \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{y} - \alpha_\mathbf{x}) \quad (10)$$

$$E_5 = \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{y} - \alpha_\mathbf{x}) \quad (11)$$

where $N_\alpha$, $N_\alpha$ denote all the spatial and temporal neighboring pairs existing in the observed $N$-frame video neighborhood (typically $N = 5$ in our experiments). $ho(\cdot)$ is the robust function given by $ho(x) = (x^2)/(x^2 + \sigma_p^2)$, with $\sigma_p = 15.3$.

We briefly discuss the role of each terms defined above. Our method is based on the assumption that each pixel has the same color as its four spatial neighbors. This assumption translates to an energy term that forces spatial consistency in intensities and supress the tendency to drive adjacent pixels to different values. This is addressed by $E_2$. Similarly, the spatial smoothness assumption can be extended to the temporal domain. In other words, we also assume that each pixel does not change its value between clean frames in both temporal directions. The expression in $E_3$ encourages smoothness in intensities along the bi-directional temporal neighborhoods. Note that both $E_2$ and $E_3$ use the artifact matte model to capitalize on the image information within the degraded region.

Unfortunately, $E_2$ and $E_3$ are not sufficient to uniquely compute all the unknowns. There exists a degenerate solution where all the unknowns are zero. In order to better condition the video restoration process, we add the bias term $E_4$ on $\alpha$. $E_4$ and $E_5$, on the other hand, address the smoothness priors for corruption (e.g., blotches, line scratches and dust). Since film artifacts generally tend to be “convex” clumps of degradation, the priors should force contiguous areas of both $\alpha$ and the artifact color $a$ to form. Therefore, it is reasonable to place a similar continuity constraint on both alpha maps ($E_4$) and artifact color maps ($E_5$).

Putting everything together in (5), we have

$$-\log p(A,\alpha | G, D) \propto w_1 \sum_{\mathbf{x}} (1 - \alpha_\mathbf{x}) + w_2 \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{x}/\alpha_\mathbf{y} - \alpha_\mathbf{y}/\alpha_\mathbf{x}) \quad (12)$$

$$+ w_3 \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{x}/\alpha_\mathbf{y} - \alpha_\mathbf{y}/\alpha_\mathbf{x}) \quad (12)$$

$$+ w_4 \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{y} - \alpha_\mathbf{x}) + w_5 \sum_{\mathbf{y} \in N_\alpha(\mathbf{x})} \rho(\alpha_\mathbf{y} - \alpha_\mathbf{x}) \quad (12)$$

which we wish to minimize. Recall that $p$ (the estimated true color), $q$ (the estimated alpha-premultiplied color), and $\alpha$ are linked to $g$ (the observed color) and $a$ (the artifact color) via (4). In our experiments, we choose the optimal values for the weights manually: $w_1 = 6.2, w_2 = 1.5, w_3 = 2.44, w_4 = 0.01$, and $w_5 = 1.0$.

In the joint estimation problem specified by (12), each pixel is treated as a node in the CRF and connected to its spatial and temporal neighbors. We implemented a version of the CRF with 6 connectivity: 4 spatial neighbors and 2 temporal neighbors.

C. Loopy Belief Propagation

To solve the CRF and estimate $A$ and $\alpha$, we implemented loopy belief propagation [17], which gives an approximation to the maximum a posteriori (MAP) solution at every pixel.

1) Setting Up the CRF: The proposed 3-D CRF has this probabilistic model

$$p(A,\alpha | G, D) \propto \prod_{\mathbf{x}} \psi(\alpha_{\mathbf{x}}, \alpha_{\mathbf{y}}, \alpha_{\mathbf{z}}, \alpha_{\mathbf{w}}) \quad (13)$$

$$\phi(\alpha_{\mathbf{x}}, \alpha_{\mathbf{y}})$$ denotes the data term (or more precisely, bias term in our case), which is defined to be

$$\phi(\alpha_{\mathbf{x}}, \alpha_{\mathbf{y}}) \propto \exp\{-w_3(1 - \alpha_{\mathbf{y}})\} \quad (14)$$

while $\psi(\alpha_{\mathbf{x}}, \alpha_{\mathbf{y}}, \alpha_{\mathbf{z}}, \alpha_{\mathbf{w}})$ is the potential function (or regularization term) defined as

$$\psi(\alpha_{\mathbf{x}}, \alpha_{\mathbf{y}}, \alpha_{\mathbf{z}}, \alpha_{\mathbf{w}}) \propto \left\{ \begin{array}{ll} \exp\{-w_3(1 - \alpha_{\mathbf{y}})\} & \text{if } \mathbf{y} \in N_\alpha(\mathbf{x}) \\ \exp\{-w_3(1 - \alpha_{\mathbf{y}})\} & \text{if } \mathbf{y} \in N_\alpha(\mathbf{z}) \end{array} \right. \quad (15)$$

If motion estimation is enabled in the loop (Fig. 3), computation of $\psi$ in (13) is done a little differently for nodes in the previous and next frames. For these nodes, $\mathbf{p}_\mathbf{x}$ and $\mathbf{p}_\mathbf{y}$ are estimated using the bilinearly interpolated color input values $g_\mathbf{x}$ and $g_\mathbf{y}$ rather than using the nearest neighbors.

2) Solving the CRF: The message update rule is (for the message from node $\mathbf{x}$ to $\mathbf{y}$)

$$m_{\mathbf{x} \rightarrow \mathbf{y}} (\alpha_\mathbf{y}, \alpha_\mathbf{z}) \leftarrow \sum_{\alpha_{\mathbf{w}}} \psi(\alpha_{\mathbf{x}}, \alpha_\mathbf{y}, \alpha_\mathbf{z}, \alpha_\mathbf{w}) \times \phi(\alpha_{\mathbf{x}}, \alpha_\mathbf{y}) \prod_{\mathbf{z} \in N_\alpha(\mathbf{x}) \setminus \{\mathbf{y}\}} m_{\mathbf{z} \rightarrow \mathbf{x}} (\alpha_\mathbf{z}, \alpha_\mathbf{w}) \quad (16)$$

As (16) shows, we multiply all the incoming messages $m$ (except the one coming from the node we are sending the message to) with the local evidence term $\phi$ for this node. We then multiply that times the potential function $\psi$ connecting this to the neighboring node, and sum over the states of this node. In our implementation, the number of states for the artifact color is $n_A = 2$ or 3 (white/black or white/black/green), while the value of alpha ($\in [0, 1]$) is discretized to $n_\alpha = 11$ bins. The total
number of states is $n_A n_\alpha = 22$ or 33. Note that each message passing term $m_i$ after each iteration is normalized so that it sums to one over all labels. This is to avoid any over- or under-flow problems. The number of iterations for message passing ranges from 3 (for simple sequences) to 10 (for sequences with more complex artifacts).

The belief, or the approximated marginal distribution is

$$b(a,\alpha) \propto \phi(a,\alpha) \prod_{y \in N(x)} m_{xy}(a,\alpha)$$  \hspace{1cm} (17)

which can be computed after all the messages have been updated. The most likely values for alpha and artifact color are chosen as the approximate solution for our joint labeling problem.

D. Correcting Corrupt Data

Given $\alpha$ and $A$, we can then attempt to recover $P$. However, it is unwise to compute $P$ directly using (3), due to sensitivity to noise, especially when $\Omega$ is small. We instead estimate the true color of a pixel using its immediate spatio-temporal neighborhood. To account for color discontinuity (in space) and

Fig. 4. Line artifacts added to clean videos as simulation (five frames). The same line artifacts were added to three “clean” videos: merry-go-round, kangaroos, and Sydney Opera House. The restored videos appear practically artifact-free.
pixel occlusion or disocclusion (along time), we introduce the alpha-modulated bilateral filter. The estimated restored color $\hat{p}_x$ at $x$ is

$$\hat{p}_x = \frac{\sum_{y \in N_{st}(x)} G_{BF}(x, y, \alpha_x, \alpha_y, \sigma_x, \sigma_y) p_y}{\sum_{y \in N_{st}(x)} G_{BF}(x, y, \alpha_x, \alpha_y, \sigma_x, \sigma_y)}$$

Note that $N_{st}(x) = N_s(x) \cup N_t(x)$, with $N_s(x)$ being the spatial neighborhood of $x$ and $N_t(x)$ the corresponding temporal neighborhood. Note that the weight $\alpha_x$ is used to both avoid division by zero and to downweight its contribution when it is small. $G_{BF}(\cdot)$ is the alpha-modulated bilateral filter and is defined as

$$G_{BF}(x, y, \alpha_x, \alpha_y, \sigma_x, \sigma_y) = \alpha_y G_S(x, y) G_R(q_x, \alpha_x, q_y, \alpha_y)$$

with $G_S(\cdot)$ being dependent on the spatio-temporal coordinates while $G_R(\cdot)$ depends on alpha-premultiplied colors and alpha values.

We define $G_S(\cdot)$ as

$$G_S(x, y) = \exp\left(\frac{-(i-j)^2 + (j-j')^2}{\sigma_{S_t}^2} - \frac{(t-t')^2}{\sigma_{S_t}^2}\right)$$

Note that $(q/\alpha)$ yields the predicted color $p$ after correction [see (4)], so that the bottom row of (21) takes into account the deviation in predicted colors. In our work, we set $\epsilon = 0.25$ and $\sigma_R = 6.2$.

The resemblance of (19) to the bilateral filter [22] is the reason for calling it the alpha-modulated bilateral filter. In the limit where all the spatial information at the current frame is uninformative (small $\alpha$’s), we rely only on the immediate temporal neighbors. For the moment, let us assume that motion estimation between successive time frames are approximately known.

### E. Handling Motion

If the amount of motion between successive frames is a pixel or less, assuming stationary scene is not unreasonable. However, in many cases, there is film unsteadiness, which causes the frame to globally shift or warp. In such cases, we estimate global motion across successive frames (pairwise) and establish links between pixels across time based on the estimated motion. (We round up the position of the pixels.)

To estimate the global motion (typically affine), we compute Harris corners and perform RANSAC to estimate the motion parameters. This works well as long as the artifacts do not dominate the entire frame.

In many cases, however, global motion compensation is also inadequate (as seen in Fig. 14), due to significant local motion. In such a case, we compute per pixel flow. Hierarchical Lucas-Kanade optical flow algorithm [15], [2] is used in this step. To reduce the errors introduced by artifacts, we smooth our flow.

We then use the estimated optical flow to construct our 3-D CRF to improve the alpha and artifact estimates. Our framework alternates between estimating optical flow, inferring alpha and artifact values using 3-D CRF and video restoration. This process is repeated until convergence.

### IV. RESULTS

In this section, we show results of applying our video restoration algorithm on videos with a variety of artifacts. As a proof-of-concept, we first demonstrate our algorithm on synthetic examples with ground truth. Next, we show results of
using our restoration algorithm on real digitized vintage videos. Finally, we compare results with a state-of-the-art technique for video restoration.

A. Synthetic Examples

We demonstrate our algorithms on two types of artifacts: lines and blotches. Three examples with line artifacts are shown in Fig. 4. The merry-go-round video was generated by repeating an image, while the kangaroo and Sydney Opera House videos were actual footage taken with a hand-held camera. Each result shown here was obtained after ten iterations, which took about 100 min on a 3.2-GHz PC. Here we assume the color artifact is black, and the alpha value is evenly divided into 30 discrete values. The RMS intensity errors for the merry-go-round video are 10.5 (over all pixels) and 12.4 (over artifact regions only). The RMS intensity errors for the kangaroo video are 6.4 (over all pixels) and 12.2 (over artifact regions only). The RMS intensity...
errors for the Sydney Opera House video are 5.1 (over all pixels) and 8.7 (over artifact regions only).

In the motion picture industry, line artifacts are defects that remain in the same place in consecutive frames, because the scratch is usually spread over many frames, and appears at same location during film projection. This kind of film damage is readily seen by the viewer and much harder to remove than blotches that move about the video. The results in Fig. 5 demonstrate that our restoration process is an effective way of removing line scratches that are in the same place in successive frames. The same input videos are used as in Fig. 4, except that this time the line artifacts from the second frame are repeated in third frame. Results are generally similar, and here we show the result for only the third frame. Most of the line artifacts were successfully removed. Note, however, that there are faint remnants of the line artifacts in the darker areas and areas with mo-
Fig. 10. Result of restoring the kangaroo video (five frames). From top to bottom: Input video frames (corrupted using artifacts shown in Fig. 6), restored frames, extracted distributions of artifact color, extracted alpha distributions (the darker the pixel, the higher the artifact contribution).

Fig. 11. Result of applying our method on a video with white artifacts (circled). This video was shot in the 1970s.

tion. It is clear that persistent artifacts are harder to completely remove.

The synthetic blotch artifacts shown in Fig. 6 were actually modified versions of the extracted artifact distributions from Fig. 12. We enhanced the alpha distribution and modified the artifact color of two large regions (second and third frames) from black to green. There are now three artifact colors: black, white, and green. Note that these artifacts are not trivial to remove: there is significant overlap of artifacts between the second and third frames and some overlap between the third and fourth frames.

We applied the artifacts shown in Fig. 6 to two different videos and processed them using our restoration algorithm. The video resolution is 320 × 240 and the number of frames is 5. The results are shown in Figs. 7 and 10 (after ten iterations, which took about 100 min on a 3.2-GHz PC). Here, because the number of artifacts has tripled, we reduced the number of states for the alpha to 10, to keep
Fig. 12. Result of applying our method on a video (shot in 1906) with severe artifacts (circled). We assume that the artifact color is black, and affine motion compensation is used. From top to bottom: Input video, restored video, alpha distribution (the darker, the higher the alpha contribution).

Fig. 13. Result of applying artifacts (recovered from video shown in Fig. 12) to a new video. The new video is converted into black and white, and some of the planted artifacts are circled in the bottom row.

the memory requirement about the same throughout the experiments.

Notice that the results for these two (synthetically) corrupted videos are slightly different. It is not surprising that the extracted artifact color and alpha distributions depend on the original color distribution. The RMS intensity errors for the merry-go-round video are 11.4 (over all pixels) and 17.7 (over artifact regions only). The RMS intensity errors for the kangaroo video are 6.3 (over all pixels) and 10.0 (over artifact regions only).

B. Three Real Vintage Videos

Fig. 11 contains white partial ring artifacts that persists over two frames. Our algorithm was able to significantly reduce their appearance. There is some zooming in the scene, which we compensate using the affine motion model. Fig. 12 shows a clip that dates back to 1906. This clip is of historical significance because it was taken by Thomas Edison soon after the San Francisco earthquake that year. Notice the significant amount of dirt on the film. This footage has jitter, and we used affine motion compensation (Fig. 12).

A significant advantage of using artifact color and alpha maps to represent the artifact is that use of original data is maximized. In addition, with this representation, the extracted artifact maps can be easily used to generate synthetic vintage effects: Fig. 13 shows the result of applying artifacts from Fig. 12 to another video clip.

Finally, we show the benefit of using local motion estimation. There is significant local motion in Fig. 14 (due to the chewing action), which even global motion cannot handle. Notice that not accounting for the local motion causes new artifacts to show up and incorrect estimation of both artifact color and...
Fig. 14. Effect of using (locally smoothed) per-pixel motion. Top row: Original video (shot in 1960). Notice the whitened region in the ear region in the third frame. Second and third rows: Result of restoration without any motion compensation. The third row shows the estimated artifact color distribution. There are many errors. Fourth and fifth rows: Result of restoration with motion compensation.

C. Comparison With Kokaram’s Technique [12]

In this section, we show two results for a representative state-of-the-art technique for video restoration. We chose Kokaram’s technique [12] for comparison because it is a good representation of video restoration techniques currently available. Many other Bayesian based video restoration techniques are variations or extensions of this technique.

In [12], Kokaram presented a nice Bayesian framework for film artifacts removal. His algorithm assumes that the artifacts do not occur at the same location in consecutive (stabilized) frames. So, given its previous and next frames, each current frame is conditionally independent of the rest of the frames in a video (i.e., restoration is based on information within 1st-order temporal neighborhood only). In addition, a binary variable is assigned to each pixel to indicate if it is clean or corrupted. Two more binary labeling variables are used, one to indicate occlusion in the backward temporal direction, the other for the forward direction. The state of a pixel is one of 6 valid states,
each of which is defined as a combination of those three binary label variables. The posterior probability over label variables is written in terms of a product of a likelihood and proper priors. Iterated conditional mode (ICM) is used to generate solutions by alternating between estimating the unknown labeling variables, restoring the degraded frame, and computing the pixel correspondence across time. These processes are iterated until reasonable convergence (in our experiments, we ran 16 iterations for each frame).

Fig. 16 compares our results with those of Kokaram’s algorithm. His algorithm works well when the underlying assumptions are met. However, in cases where an artifact persists for more than a frame (e.g., the circled regions), it is retained. In addition, because of the assumption of a pixel being either totally valid or corrupted, this algorithm either keeps or tosses out pixels with partial contamination, depending on a difference threshold. (The threshold of 31 intensity levels seem to generate visually best results.) The areas within white rectangles are example regions with partial contamination left behind.

It would be interesting to compare the results of our approach with those of [11]. However, we expect similar results, since both [12] and [11] do not handle temporally persistent artifacts.

D. Comments on Computational Complexity

The main disadvantage of this algorithm is that, by adopting 3-D CRF model and loopy belief propagation inference method, the computation for the state estimation is very high per pixel. Our loopy belief propagation runs in \(O(nk^2T)\) time, where \(n\) is the number of pixels in 5 frames, \(k\) is the total number of possible states (i.e., 22 or 33) for each pixel and \(T\) is the number of iterations. Basically, it takes \(O(k^2)\) time to compute each message and there are \(O(n)\) messages per iteration.

There are two possible avenues that could be pursued to improve the computational efficiency. One approach is to use older, less computationally intensive algorithms, such as iterated conditional mode (ICM). ICM uses a deterministic “greedy” strategy to find a local minimum. It starts with an estimate of the labeling and then, for each pixel, it chooses the label giving the largest decrease of the energy function. This process is repeated until convergence, which is guaranteed to occur, and in practice, is rapid, since it runs in \(O(nkT)\) time. Unfortunately, the results are extremely sensitive to the initial estimate, especially in high-dimensional spaces with nonconvex energies (such as the energy function used in this paper) due to the huge number of local minima. Fig. 8 shows the ICM inference results when starting with a good initial guess and Fig. 9 shows the ICM inference results when starting with a noninformative initial guess. Both Figs. 8 and 9 use the same input video as in Fig. 7. ICM in Fig. 8 converges on the correct solution when the initial estimate is set to a modified version of the ground truth. ICM in Fig. 9, on the other hand, is trapped at a noticeably wrong local minimum when the alpha maps are initialized to 1 and the artifact maps are initialized to 0. As it can be readily seen, a bad initial guess causes ICM to converge on a local minimum that is far away from the optimal solution. Note that ICM is much faster than loopy belief propagation. ICM in both Figs. 8 and 9 converges after 20 iterations, which took approximately 304 s on a 3.2-GHz PC. While, in Fig. 7, loopy belief propagation took about 100 min on the same machine.

Another approach is to reuse some intermediate computations during the computation of other messages. This requires careful analysis of the functional form of each energy term. This study is definitely important, but beyond the scope of this paper. Felzenszwalb and Huttenlocher [6] provide a principled way...
for speeding up belief propagation for a large class of energy functions.

V. DISCUSSION

The weakest link in our algorithm is in establishing pixel correspondence across time. We expect our algorithm to fail when there is significant local motion and the artifacts occupy large areas (which would pose a significant problem for other restoration algorithms as well). Here, we expect that user interaction (say to specify what areas are artifacts) would be necessary for effective artifact removal.

The motion and smoothness assumptions may not be applicable to certain types of scenes. For instance, the method may fail if the scene has repetitive, Brownian, or chaotic motion such as a river, sea waves, trees, or people dressed in highly patterned clothes.

We currently estimate the artifact color and alpha independently in time, and we do not impose any spatial prior on them as well. It would be interesting to be able to first estimate these maps and use them for artifact identification. For example, if the artifacts seem to be line artifacts, stronger priors on the spatial pattern would be more helpful.

VI. CONCLUDING REMARKS

In this paper, we formulated the restoration process of vintage videos as that of matting. The artifact color and alpha distributions over all the pixels are recovered using 3-D CRF with spatio-temporal connections. We introduced the \textit{alpha-modulated bilateral filter} to handle spatial color discontinuity and pixel occlusion or disocclusion. The color-alpha artifact representation has the advantage of a smaller search space (compared to directly estimating the clean colors). Furthermore, it can be used as an effect on other videos. Belief propagation is used as the inference mechanism to recover this representation. Results show the effectiveness of our technique on video clips with different types of artifacts.

REFERENCES


Jiadong Gai received the B.S. degree in automation and the M.E. degree in pattern recognition from the Department of Automation, Tsinghua University, China, in 2001 and 2004, respectively, and the M.S. degree in applied mathematics (MSAM) from the Department of Mathematics, University of Notre Dame, South Bend, IN, in 2007, where he is currently pursuing the Ph.D. degree.

His research interests include video processing, machine learning, and mathematical physics. He did summer internships at Microsoft Research (Redmond, WA) and Sharp Labs of America (Camas, WA) in 2007 and 2008, respectively.

Sing Bing Kang (SM’01) received the Ph.D. degree in robotics from Carnegie Mellon University, Pittsburgh, PA, in 1994. He is currently a Principal Researcher at Microsoft Corporation working on image-based modeling as well as image and video enhancement. He has coedited two books in computer vision (\textit{Panoramic Vision and Emerging Topics in Computer Vision}), and co-authored a book on image-based rendering.

Dr. Kang has served as area chair and member of technical committees for the three major computer vision conferences (ICCV, CVPR, and ECCV). He has also served as papers committee member for SIGGRAPH’07 and SIGGRAPH Asia’08. He is currently CVPR’09 program cochair and IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE Associate Editor.