Fast High-Definition Video Background Completion Using Features Tracking

Jocelyn Benoit
École des arts numériques, de l’animation et du design
Montreal, Canada
Email: jbenoit@nad.ca

Eric Paquette
Multimedia Lab
École de technologie supérieure
Montreal, Canada
Email: eric.paquette@etsmtl.ca

Abstract—This paper presents an automatic video background completion approach based on invariant features tracking and image registration to find valid replacement regions. Previous exemplar-based methods provide good results for low-resolution video sequences, but suffer from long computation times and large memory consumption for high-definition sequences. We first select a candidate frame to complete a missing region using invariant features tracking and image registration. This greatly reduces computation times as it does not require the lengthy nearest neighbor searches seen in typical video completion methods. To minimize registration errors, we introduce a fast validation approach. Then, we propose an exposure correction method based on histogram specification to eliminate illumination inconsistencies in the completed regions. Finally, we complete the missing region with a multi-band blending approach to minimize boundary discontinuities. Our approach can achieve good quality results on high-definition videos, and it can deal with a variety of real-life problems, such as non-trivial camera movement and illumination changes. Furthermore, the proposed method requires low computation times which represent a 24-54 times speedup over previous methods. In addition to providing specific implementation details, this paper presents experimental results on a variety of videos and compares them to state-of-the-art methods in terms of visual quality and performance.

I. INTRODUCTION

Both image and video completion are frequent and important tasks realized by artists during postproduction for film and television. Their goal is to automatically fill an unwanted or missing region of an image or a video, commonly referred to as a hole, in a visually plausible fashion. Image completion has been studied extensively over the past fifteen years, and most image completion manipulations can be done automatically using standard methods [1]–[3]. Unlike image completion, current video completion methods are far less accomplished, even though some great results have been obtained. This is due to the fact that video completion poses additional problems, such as high time complexity, memory limit, and temporal consistency. Therefore, in production studios, artists handle the majority of their video completion tasks manually. Nevertheless, there is a variety of important applications such as repairing scratches and artifacts from newly digitalized old films [4], [5], removing unwanted persons or objects captured during movie shootings [6], [7] or eliminating logos and watermarks that require better automated tools. Since manual video completion takes hours, and even days, and does not require any artistic vision, it ought to be automated. Like Ilan and Shamir, we believe that: “even a semi-automatic tool that would relieve some of the manual labor can be of great benefit to video-editing professionals” [8, p.120].

In this paper, we propose a fast semi-automatic high-definition (HD) video background completion approach based on features tracking. It has many benefits compared to previous works. Firstly, we take on the problem of long execution times, which is a major challenge for video completion. This is caused by the high computational cost associated with searching for the nearest neighbors of each video patch. We tackle this problem by proposing a video completion framework based on features tracking and image registration, which removes the nearest neighbors search altogether. Secondly, we address the challenge of completing video sequences which contain complex moving backgrounds (zoom, rotation, shaking of the camera, etc.). To that end, we base our robust image registration process on scale-invariant features and introduce a fast image registration validation based on peak signal-to-noise ratio (PSNR). Thirdly, we make sure we maintain good visual quality results and minimize the noticeable boundary around the completed region. To this end, we propose a compositing method based on histogram specification and multi-band blending. Together, these contributions allow our approach to complete large missing regions of HD video sequences in very short time periods, surpassing anything done in previous works.

II. RELATED WORKS

In the past years, many methods have been proposed to fill missing regions of video sequences. We can classify most of these methods in two groups: object-oriented completion approaches, and patch-oriented completion approaches. The first group focuses on the reconstruction of specific objects (generally human bodies) and their motion through consecutive frames by copying the entire object found elsewhere in the existing regions. The second group tries to complete the missing regions incrementally with an iterative process of finding and copying similar patches of the existing regions.

The method of Cheung et al. [9] falls in the first group. The method utilizes background subtraction and object segmentation to extract the object template and uses object interpolation...
to complete the moving objects. The method is subsequently improved by enabling the completion of video with simple parallel-to-scene camera motion [10]. While these methods give satisfactory results, they present many restrictions for the moving object motion (e.g. periodicity), and the camera must be stationary or with little motion, making them unusable for most general cases.

However, most state-of-the-art methods belong to the second group. Patwardhan et al. [11] proposed a local method that uses a simple optical flow to separate the foreground and background. A static camera with a static background is mandatory in the method. The background is filled using temporal offset with the information available in nearby frames. The foreground is then completed frame by frame through a search for a replacement patch only in the moving region of the video. This method was later improved for constrained camera motions [12], and with the use of the motion-compensated neighbor embedding [13]. Wexler et al. [14] presented a video completion method as a global optimization problem which considers a patch like a space-time cube that contains both color and motion information. Even with optimization methods, the high dimensionality of the problem makes this method unusable for HD video sequences. Global optimization methods require long search times and too much memory to enable the completion of HD video. Benoit and Paquette [15] tackled this problem with a localized search at higher resolution based on the principle of coherence [1] to reduce the search space, but failed to complete complex moving backgrounds. Newson et al. [16] accelerated the work of Wexler et al. [14] by extending the PatchMatch algorithm [2] to the 3D domain to search for the nearest neighbors. They later improved their method [17] and were able to reconstruct dynamic videos correctly. To the best of our knowledge, the completion results of Newson et al. [17] are the best to date, and we shall therefore compare our method with theirs. Granados et al. [6] extended an image editing method [18] to complete video sequences. Their method requires user assistance to identify the foreground objects in order to reduce the search space and enable the completion of higher resolution video than most previous works (up to 1120 × 754 pixels). Unfortunately, the method still requires long computational times (up to 90 hours for some videos). Afifi et al. [19] introduced a method based on patch-based synthesis and image registration, but the completed region sometimes shows blurring artifacts. Herling et al. [20] proposed a real-time video completion method that reduces the computation times. However, the method only considers the previous and current frames in completing missing regions. It is thus not suitable for more complex video sequences where the missing information cannot be found in those two frames.

Current state-of-the-art video completion methods require long computation times and large amounts of memory, making most of them unusable for HD sequences. Furthermore, the majority of the methods fail to complete video sequences which contain complex moving backgrounds. Our approach tackles these limitations, and provides the following four main contributions:

- A framework for video completion using invariant features tracking and image registration-based morphing of the source video sequence;
- A fast validation approach to identify target frame registration error;
- A histogram specification method to adjust target frame and limit illumination variations;
- A multi-band blending approach to hide visible boundaries.

### III. Proposed Approach

Starting with a source video sequence $S$ containing a missing region or hole $H$ (identified by a binary mask $M$), our approach fills $H$ in a visually plausible manner by copying similar regions found in the existing region $E \ (E = S \setminus H)$, thus creating a completed video sequence $S^*$. The proposed approach is based on invariant features tracking and image registration in order to deal with the problem of moving background. The method consists of four steps: (A) Target frame registration and initial validation, (B) Exposure correction, (C) Advanced target frame validation, and (D) Final blending (see algorithms 1, 2).

**Algorithm 1** Video completion algorithm.

**Input:** Sequence of frames $S$ and binary masks $M$  
**Output:** Completed sequence $S^*$

```plaintext
1: $L \leftarrow$ number of frames in $S$
2: for $i \leftarrow 0$ to $L$ do
3:    $F_i \leftarrow$ extract SURF features($S_i$);  \(\triangleright\) Section III-A
4: end for
5: for $i \leftarrow 0$ to $L$ do
6:    $T \leftarrow$ ordered target frame index list($i, L$);
7:    $isFilled \leftarrow$ false;
8:    $j \leftarrow 0$;
9:    for $j \leftarrow 0$ to $L$ do
10:       $isFilled, S_i^* \leftarrow$ fill region$(S, M, i, T_j, F)$;
11:    end for
12: end for
```

**Algorithm 2** Fill region algorithm.

**Input:** Sequence of frames $S$, binary masks $M$, source index $i$, target index $t$, SURF features $F$  
**Output:** Boolean indicating whether the frame is completed, completed frame $S_i^*$

```plaintext
1: $C \leftarrow$ estimate camera($F_i, F_t$);  \(\triangleright\) III-A
2: $s', m' \leftarrow$ warp image and mask($S_t, M_t, C$);
3: $s' \leftarrow$ correct exposure($S_t, M_t, s', m'$);  \(\triangleright\) III-B
4: if invalid target image($S_t, M_t, s', m'$) then
5:    return false, null $S_i^*$;  \(\triangleright\) III-C
6: $S_i^* \leftarrow$ blend($S_t, M_t, s', m'$);  \(\triangleright\) III-D
7: return true, $S_i^*$;
```
A. Target Frame Registration and Validation

The first step of the proposed method is to extract the invariant features from each frame of the source video sequence. Then, for each frame $S_i$ containing a missing region, we seek a target frame $S_t$ in which a valid replacement region exists (Sec. III-C). In order to compensate for the possible viewpoint difference between $S_i$ and $S_t$, the approach finds a valid transformation $T_{it}$ that aligns matching features ($S_i = T_{it} S_t$). There are many methods used to estimate $T_{it}$; we choose the homography method, and parameterized each camera by a rotation vector $\theta = [\theta_1, \theta_2, \theta_3]$ and focal length $f$, like Brown and Lowe [21]. For the invariant features, we choose the Speed-Up Robust Features (SURF) [22].

To find this transformation between $S_i$ and $S_t$, we first match the SURF features [22] of $S_i$ and $S_t$ together. Then, a homography $H_{it}$ is estimated iteratively from this set of feature matches with the fuzzy RANSAC (random sample consensus) algorithm [23], and a first registration validation is done using a probabilistic model for image match verification [21]. The frame registration quality is crucial to ensure good completion results. Even with this first validation, some registration errors will still be present. To avoid these cases, the proposed approach introduces a second registration validation method in Sec. III-C. This validation relies on pixel-to-pixel color differences, and thus, is computed after exposure correction (Sec. III-B).

B. Exposure Correction

Many video sequences, especially outdoor scenes, exhibit illumination variations. Using unadjusted target frames directly sometimes causes illumination inconsistency artifacts, particularly noticeable around the hole boundary (see Fig. 1).

To minimize these illumination variations, we adjust the target frame $s'$ using a histogram specification technique [24]. The idea is to match the histogram of the regions just outside the missing region $H_i$, which is obtained with a morphological dilation of the mask $M_i$. These regions are $\Omega_i$ ($\Omega_i \subset S_i$) and $\Omega_t$ (corresponding region of $\Omega_i$ in the warped target frame $s'$). Then, we evaluate the color transformation $G_{it}$ between these two histograms. Finally, $G_{it}$ is applied to $s'$. Fig. 1 shows an example of a completed region with and without exposure correction.

C. Advanced Target Frame Validation

Once exposure correction is completed, we must validate that $s'$ is a good target to complete $S_i$. First, the target frame is considered invalid if one pixel from the source hole lands in another hole region in the warped target frame, that is if some pixel $p \in H_i$ corresponds to a hole identified by $m'$ (see Fig. 2(c)). Secondly, the target frame is also considered invalid if one pixel from the source hole lands outside of the warped target frame, that is, if some pixel $p \in H_i$ is outside of the boundary of the frame quadrilateral in $s'$ (see Fig. 2(d)).

D. Final Blending

Even with the introduced exposure correction method and registration validation technique using PSNR, some boundaries are sometimes still visible. To address this problem, we adapt the multi-band blending technique of Burt and Adelson [25]. As stated by Brown and Lowe [21]: “The idea behind multi-band blending is to blend low frequencies over a large spatial range, and high frequencies over a short range” [21, p.7]. We adapt the method of Brown and Lowe [21] to the particular case of filling a missing hole. Thus, we initialize blending weights for $S_i$ and $s'$ using the following equations:
\[ W_{S_i}(p) = \begin{cases} 1 & \text{if } p \notin H_i, \\ 0 & \text{otherwise.} \end{cases} \] (3)

\[ W_{c_i}(p) = \begin{cases} 1 & \text{if } p \in H_i, \\ 0 & \text{otherwise.} \end{cases} \] (4)

IV. EXPERIMENTAL RESULTS

The objective of the proposed method was to achieve high-quality video completion on various complex HD videos in reduced execution times. We present the results of the proposed approach on six HD video sequences and compare the completion with other methods. Each sequence highlights the robustness of the method in addressing complex camera movement and illumination variation. The software is written in C++ and no particular optimization was made. The experiments were computed on a 3.50 GHz Intel® Core™ i7-4770K CPU with 32 GB of RAM. The results of Newson et al. [17] were obtained on the same computer with the Matlab provided by the authors.

The registration application was achieved by using SURF [22] with a Hessian threshold of 500 and fuzzy RANSAC [23]. The parameters used to obtain the results of Newson et al. [17] were the same as those from their paper except for the resolution level, which was set at six (HD sequence).

A. Visual Quality

The video completion result quality is hard to evaluate with state-of-the-art objective evaluation tools such as PSNR. Therefore, the quality of the results is assessed based on human visual inspection [6], [17]. The visual results of our approach compared to those of Newson et al. [17] can be seen in Figs. 3, 4, 5, and 6.

We provide results to show the ability of the proposed approach to deal with various real-life situations. In certain cases, both methods obtain similarly high-quality results (see Fig. 3). However, the proposed approach shows better result quality for video sequences with more complex camera movements (e.g., zooming; see Figs. 4, 5, and 6).

B. Performances

Execution time and memory consumption are among the biggest problems in the video completion field, making most previous works unusable in real-life situations. Table I compares our execution times to those of Newson et al. [17]. It is important to note that the method of Newson et al. [17] failed to complete the longer video sequences because it ran out of memory (32 GB). Therefore, we needed to manually cut up the source videos into multiple sequences of 100–120 frames. Even without taking this manual editing time into account, we obtain a 24 to 54 times speedup over the method of Newson et al. [17].

The results show the completion of large missing regions (400 × 400) in HD resolution (1920 × 1080) sequences, which has never been seen in previous works. The fact that our approach is not limited by the resolution and the length of the source video sequence represents a significant advantage over previous works.

C. Limitations

Our approach works well in most situations where the background has distinctive keypoints to track. However, when the video presents dynamic textures (such as water with waves), the approach fails to find a valid replacement region, and the completed region is easily noticeable to the viewer. Moreover, for image registration using homography to be valid, the scene must contain only one plane. Future work will consist in finding a target frame for each plane in the source frame. Granados et al. [6] represent a good attempt in this direction.

V. CONCLUSION

In this paper, we have proposed a fast video completion approach that fills a missing region of moving background which produces excellent quality results for many camera movements, and which works on HD video sequences. This approach, which is based on invariant features tracking and image registration, does not need heavy memory structures to accelerate the search for the nearest neighbors, as do most previous works; it thus increases the resolution and length of video sequences that can be handled by orders of magnitude. It is our hope that the short execution time and the high quality of the results will make this method usable by video-editing professionals, and relieve them from some of the manual tasks they must accomplish.

ACKNOWLEDGMENT

The authors would like to thank Dominic Mercier, Benoît Melançon, and all the reviewers for their precious help.

REFERENCES

TABLE I: Statistics of the video sequences used in our tests

<table>
<thead>
<tr>
<th>Name</th>
<th>Resolution</th>
<th>Length</th>
<th>Missing region / frame</th>
<th>Avg. missing pixels</th>
<th>Newson Time (min.)</th>
<th>Our approach Time (min.)</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apartment01</td>
<td>1920 × 1080</td>
<td>240</td>
<td>26,293,310</td>
<td>~ 5.2%</td>
<td>470.6</td>
<td>13.0</td>
<td>36.1</td>
</tr>
<tr>
<td>000-112</td>
<td>1920 × 1080</td>
<td>113</td>
<td>12,379,728</td>
<td>~ 5.2%</td>
<td>252.0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>113-239</td>
<td>1920 × 1080</td>
<td>127</td>
<td>13,913,582</td>
<td>~ 5.2%</td>
<td>217.7</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Cafe001</td>
<td>1920 × 1080</td>
<td>97</td>
<td>797,639</td>
<td>~ 4%</td>
<td>147.4</td>
<td>2.7</td>
<td>54.4</td>
</tr>
<tr>
<td>Cafe002</td>
<td>1920 × 1080</td>
<td>298</td>
<td>49,362,182</td>
<td>~ 8%</td>
<td>646.1</td>
<td>22.1</td>
<td>29.3</td>
</tr>
<tr>
<td>000-100</td>
<td>1920 × 1080</td>
<td>101</td>
<td>16,729,736</td>
<td>~ 8%</td>
<td>195.4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>101-201</td>
<td>1920 × 1080</td>
<td>101</td>
<td>16,730,549</td>
<td>~ 8%</td>
<td>249.6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>202-297</td>
<td>1920 × 1080</td>
<td>96</td>
<td>15,901,807</td>
<td>~ 8%</td>
<td>201.1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CanadianPacific008</td>
<td>1920 × 1080</td>
<td>241</td>
<td>33,519,208</td>
<td>~ 6.7%</td>
<td>788.5</td>
<td>21.8</td>
<td>36.2</td>
</tr>
<tr>
<td>000-120</td>
<td>1920 × 1080</td>
<td>121</td>
<td>16,829,012</td>
<td>~ 6.7%</td>
<td>274.8</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>121-241</td>
<td>1920 × 1080</td>
<td>121</td>
<td>16,690,256</td>
<td>~ 6.7%</td>
<td>513.7</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CanadianPacific006</td>
<td>1920 × 1080</td>
<td>306</td>
<td>27,813,536</td>
<td>~ 4.4%</td>
<td>604.7</td>
<td>25.1</td>
<td>24.1</td>
</tr>
<tr>
<td>000-102</td>
<td>1920 × 1080</td>
<td>103</td>
<td>9,362,276</td>
<td>~ 4.4%</td>
<td>253.6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>103-206</td>
<td>1920 × 1080</td>
<td>104</td>
<td>9,453,185</td>
<td>~ 4.4%</td>
<td>216.0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>207-305</td>
<td>1920 × 1080</td>
<td>99</td>
<td>8,998,075</td>
<td>~ 4.4%</td>
<td>135.1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CanadianPacific007</td>
<td>1920 × 1080</td>
<td>121</td>
<td>16,473,205</td>
<td>~ 6.6%</td>
<td>306.6</td>
<td>10.2</td>
<td>30.0</td>
</tr>
</tbody>
</table>

Fig. 3: Comparison with the method of Newson et al. [17] on the frames 03, 29, and 93 of sequence Cafe001.


Fig. 4: Comparison with the method of Newson et al. [17] on the frames 1, 81, and 237 of sequence Cafe002.

Fig. 5: Comparison with the method of Newson et al. [17] on the frames 16, 157, and 212 of sequence CanadianPacific008.

Fig. 6: Comparison with the method of Newson et al. [17] on the frames 12, 60, and 104 of sequence CanadianPacific007.