100+ TIMES FASTER VIDEO COMPLETION BY OPTICAL-FLOW-GUIDED VARIATIONAL REFINEMENT

Alexander Bokov, Dmitriy Vatolin

Lomonosov Moscow State University, Russia

ABSTRACT

Despite the higher video-completion quality that recently proposed methods have enabled for a wide variety of cases, their computational complexity remains a major concern. These methods typically frame video completion as an optimization problem over the whole spatiotemporal domain—a problem that is expensive to solve both in time and space. In this paper we propose a lighter-weight multipass video-completion pipeline that replaces global spatiotemporal optimization with simpler frame-by-frame motion reconstruction and refinement. We achieve a processing speed of 2.6 seconds per frame on Full HD content while delivering nearly state-of-the-art completion quality for a wide range of dynamic scenes captured using a free-moving camera. To validate the performance of our proposed method, we conducted a subjective comparison of different video-completion results for 26 test sequences from the DAVIS data set.

Index Terms— Video completion, inpainting

1. INTRODUCTION

Video completion is an important problem in video processing and has a wide variety of applications, including video restoration, rig removal and occlusion filling in virtual-view synthesis. Compared with more-widespread image inpainting, video completion introduces additional challenges, such as the need to handle vastly larger amounts of data and to maintain temporal coherency in the presence of arbitrary camera motion. At the same time, better inpainting results are typically possible using information available in other input-video frames. Recently, work by Newson et al. [1] addressed the problem of handling complex motion such as crashing waves, but it employed simple affine realignment to handle camera motion. Huang et al. [2] showed the limitations of such an approach and proposed joint estimation of optical flow and color in the missing region as a more general way to handle camera motion. Both techniques, however, take several hours to inpaint a moderately large missing region in a 90-frame 480p video sequence (see Figure 1).

Fig. 1. Speed/quality tradeoff comparison. Running time is measured for the "Camel" sequence (854×480, 90 frames). Subjective-quality scores were computed for the DAVIS data set using the Crowd Bradley-Terry model.

To address the high computational complexity, we took a simpler greedy approach instead of jointly estimating the flow and color through global spatiotemporal optimization. Our approach estimates the optical flow for the missing region in each frame independently (Section 3.1) and accumulates the results from several passes over the input sequence to incrementally construct a mapping from the missing regions to the known regions of the appropriate source frames to enable reconstruction. We ameliorate the resulting accumulation of flow and color error by applying frame-by-frame variational refinement (Section 3.2) and illumination adjustment (Section 3.3). Doing so dramatically reduces computational complexity and memory requirements, and it allows us to process long high-resolution videos in a reasonable time while still supporting reconstruction by copying from any input-sequence frame.

Despite the inherent limitations of such greedy frame-by-frame processing, it provides competitive video-completion results for a wide range of dynamic videos with a freely moving camera, except for the specific cases we discuss in Section 4.1. To demonstrate this claim we applied three video-completion methods to 26 test videos from the DAVIS

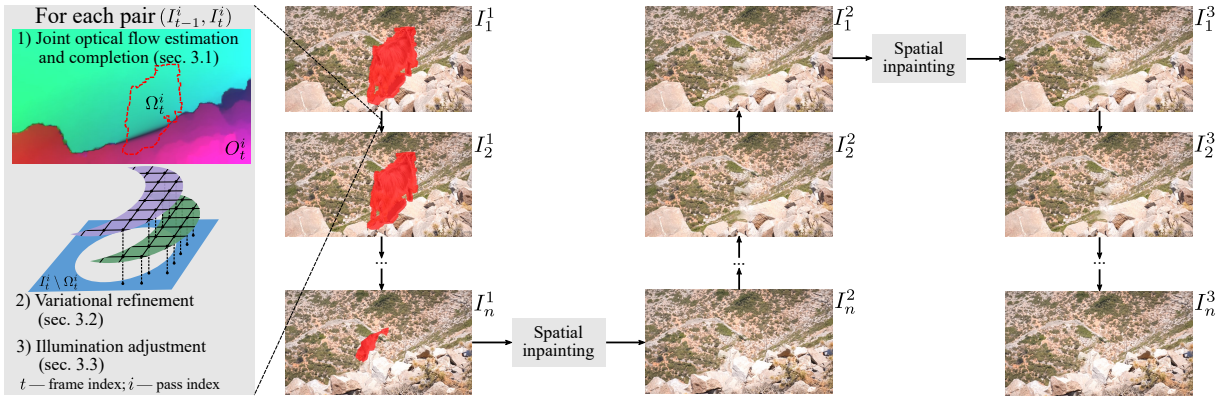


Fig. 2. Proposed video-completion pipeline. We use a three-pass scheme for temporally consistent propagation of known input-video fragments and spatial-inpainting results from the first and last frames. The missing region appears in red.

data set [3], which Huang et al. [2] introduced in the video-completion literature, and then computed subjective quality scores using the Subjectify.us web service. As Figure 1 shows, our proposed method yields completion quality that is close to the state of the art for this data set.

2. RELATED WORK

Numerous video-completion algorithms have been proposed over the years; for a more complete overview, consult the survey by Ilan and Shamir [4]. We mostly focus on newer methods that are relevant to our approach. Since the seminal work by Wexler et al. [5], spatiotemporal-patch-based techniques have appeared extensively in the video-completion literature either for performing motion completion or for inpainting the missing region directly [1, 6, 7, 8]. Although they provide impressive results in many cases, arbitrary camera motion is difficult to handle when using spatiotemporal patch sampling. Le et al. [9] addressed this issue by employing skewed parallelepipeds, where the optical-flow field defines the degree of skew. Huang et al. [2] instead chose two-dimensional patches while relying only on the estimated flow field to ensure temporal consistency. Patch-based synthesis typically requires repeated computation of nearest-neighbor fields—a task that despite all the speedup efforts [10, 11, 12] remains a major source of computational complexity in related video-completion methods. For this reason we avoid using patch-based techniques except when applying spatial inpainting to the input video's first and last frames. Although some approaches rely on a predefined parametric motion model such as affine [1], projective [13] or piecewise projective [14] transformations, we selected a more general nonparametric optical-flow-based model, as in [2, 15, 16].

Our work builds on a previous hole-filling method for virtual-view synthesis [17] and has several important distinctions. We propose a novel joint optical-flow estimation and completion approach that makes the method far more robust. We integrate our temporal reconstruction with spatial inpainting to fill in fragments that are missing from all input-video frames. Finally, we adapt the Poisson blending [18] formulation to perform illumination adjustment.

3. PROPOSED METHOD

Our method consists of three passes over the input sequence, as Figure 2 illustrates. First, a forward pass incrementally creates a cumulative mapping V_t from each current frame I_t to the previous frames $\{I_k\}_{k=1}^{t-1}$ and simultaneously performs reconstruction by copying corresponding fragments from the previous frames into the current frame. We use an existing spatial-inpainting algorithm [19] to fill in regions that are still missing from the last frame after reconstruction. Second, a backward pass enables temporally consistent propagation of the spatial-inpainting result and copying of known fragments from any future frame. It finishes by applying spatial inpainting to any regions in the first frame that are still missing. And third, we apply another forward pass to propagate the spatial-inpainting result from the first frame. This spatial-inpainting integration is less general than in the framework Huang et al. proposed [2], but it allows much faster processing. In rare cases where missing regions persist after three passes, we apply a diffusion-based image-inpainting method [20] to each frame independently as a final step. In all passes we process each frame by first computing an optical-flow field that is smoothly interpolated into the missing region, sum it with the cumulative mapping to previous frames and refine the new mapping through a variational approach. For our final step before reconstruction we apply Poisson blending, with additional temporal-consistency constraints, to each copied fragment. The following sections describe in more detail our flow computation, variational refinement and Poisson blending modification.

3.1. Fast joint optical-flow estimation and completion

Unlike existing two-frame motion-completion approaches [15, 21] that perform optical-flow estimation and completion separately, we perform these tasks jointly. Our approach

is inspired by a recent fast optical-flow algorithm [22], but instead of alternating patch-based gradient descent and variational refinement, we combine them into a single optimization problem to be solved on a sparse grid. In particular, we employ $s \times s$ patches with a stride of s pixels and compute one optical-flow vector per patch from the current frame I_0 to the previous frame I_1. We use $P(x)$ to denote a patch centered on pixel x. The data term is undefined inside the respective missing regions Ω_0 and Ω_1, so the smoothness term becomes the only constraint there. To estimate the sparse optical flow $O = (u, v)$ both inside and outside the missing region, we minimize the following energy function:

$$E(O) = \sum_x \Phi\Big(\sum_{\substack{p \in P(x) \\ p \notin \Omega_0 \\ p+O(x) \notin \Omega_1}} w_p \Big(\nabla_{O(x)} I(p) - \frac{\sum_p w_p \nabla_{O(x)} I(p)}{\sum_p w_p}\Big)^2\Big) + \alpha \Phi\big(||\nabla u(x)||^2 + ||\nabla v(x)||^2\big). \quad (1)$$

Similar to [22], $\Phi(s^2) = \sqrt{s^2 + 0.001^2}$ is a robust penalty and $w_p = (1 + ||\nabla I_0(p)||^2)^{-1}$ is a normalization weight for the brightness-constancy term. $\nabla_{O(x)} I(p)$ denotes the brightness difference $I_0(p) - I_1(p + O(x))$ and α is a smoothness-term weight. We omit the gradient-constancy term, as we already have patch mean-normalization in the data term to reduce the influence of illumination changes. We found that the high-frequency component of the optical-flow field has a negligible impact on completion quality, so computing flow vectors with a stride of $s = 8$ is sufficient. The final per-pixel optical-flow field is the result of bilinear interpolation.

We solve (1) iteratively in a Gaussian pyramid with the same patch size s for all scales. We use a downscale factor of 2 and select the coarsest scale so at least one $s \times s$ patch still fits in the downscaled frame. For each scale we perform N_{of} outermost iterations with a first-order Taylor expansion of I_1 in the neighborhood of the current optical-flow-vector approximation. Each outermost iteration includes N_{fp} fixed-point iterations, and each fixed-point iteration entails solving a linear system with N_{SOR} successive over-relaxation (SOR) iterations.

3.2. Frame-by-frame variational refinement

By accumulating interframe optical-flow fields we obtain an initial approximation of the mapping V_t that defines the correspondence between certain parts of the missing region Ω_t in the current frame and known regions in other frames. This mapping can be used as is for reconstruction, but the variational-refinement step can increase its quality. The goal of this additional step is to improve the alignment between fragments copied from different source frames as well as the alignment with the known fragments around the missing region Ω_t, thereby avoiding visible seams and compensating for optical-flow-error accumulation. To enable optimization for alignment of different fragments, we maintain an overlap of $d = 6$ pixels between mappings to different source frames when accumulating V_t. Doing so allows us to define an optical-flow-like data term $E_D(V_t)$ that penalizes misalignment and to define a smoothness term $E_S(V_t)$ that maintains the continuity of each mapping to a given source frame:

$$V^t = \arg\min_{V^t} \big(E_D(V^t) + \lambda E_S(V^t)\big). \quad (2)$$

Solving this optimization problem involves N_{ref} fixed-point iterations and N_{CG} iterations of the conjugate-gradient method. A more formal description, along with the definition of E_D and E_S, appears in [17].

3.3. Illumination adjustment

Compared with existing Poisson-blending extensions in the video-completion literature [14], we propose a different way to handle temporal consistency. Instead of uniformly penalizing the deviation from the motion-compensated previous frame, we use adaptive weights $w_p^{PB} = (1+\sigma^{PB}||\nabla I^{PB}(p) - G_t(p)||^2)^{-1}$ to enforce temporal consistency more in regions where the gradient field of the base Poisson-blending result I^{PB} (i.e., solution of (3) with $w_p^{PB} \equiv 1$) deviates the most from the target gradient field $G_t(p)$. This approach enables better processing of scenes with global uniformly changing brightness (since w_p^{PB} will be close to one) while enforcing temporal stability around local inconsistencies. So, to perform the final reconstruction we copy the gradient-field fragments from corresponding source frames to form the target gradient field G_t, and we use the surrounding region in the current frame I_t along with the reconstruction results from the previous frame I_{t_0} to formulate boundary conditions weighted separately according to w_p^{PB}:

$$B(I) = \sum_{p \in \Omega_t} ||\nabla I(p) - G_t(p)||^2 + \sum_{p \in \delta\Omega_t} w_p^{PB} ||I(p) - I_t(p)||^2 + \sum_{p \in \Omega_t} (1 - w_p^{PB}) ||I(p) - I_{t_0}(p + O_t(p))||^2. \quad (3)$$

Here, $\delta\Omega_t$ denotes the outer-boundary pixels of the missing region Ω_t, and σ^{PB} in w_p^{PB} is a constant that defines the strength of temporal-consistency enforcement.

We found that minimizing (3) directly is too computationally expensive; it becomes a bottleneck in our proposed video-completion pipeline. Instead we use a separable approximation by decomposing $B(I)$ into row- and column-wise energy functions B_H and B_V that result from simply ignoring all the vertical and horizontal dependencies, respectively. The authors of [23] previously applied the same idea to fast global filtering. The solution then comes from alternately minimizing B_H and B_V, using copied source-frame fragments as the

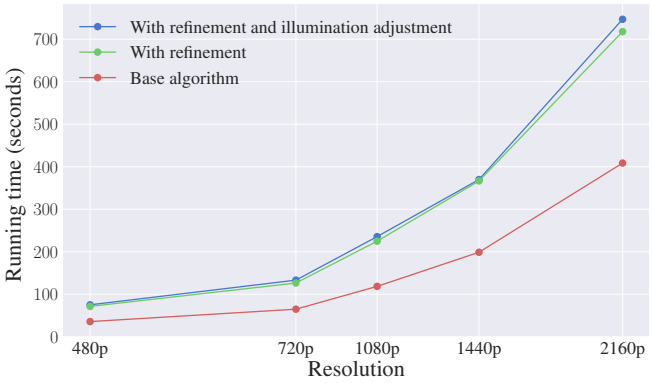


Fig. 3. Running time of the proposed method on the "Camel" sequence upscaled to various resolutions (90 frames, DAVIS data set). The base algorithm uses the full video-completion pipeline but omits the optional variational-refinement and illumination-adjustment steps.

initial approximation:

$$I_k^{'} = \arg\min_I(\epsilon_k \sum_p ||I(p) - I_{k-1}(p)||^2 + B_H(I)), \quad (4)$$

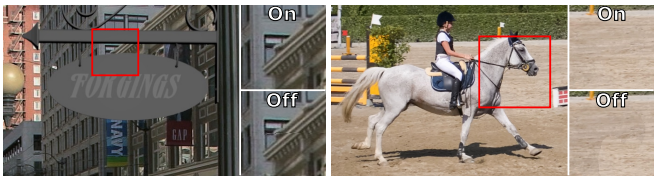
$$I_k = \arg\min_I(\epsilon_k \sum_p ||I(p) - I_k^{'}(p)||^2 + B_V(I)), k = 1 \dots K$$

We use $K = 5$ iterations and $\epsilon_k = 10^{-4} \cdot 8^{k-1}$ as weights controlling the contribution of the previous iteration.

4. EXPERIMENTAL RESULTS

We implemented the proposed video-completion algorithm in C++ using multicore parallelization of all major components and using SIMD optimizations in the optical-flow solver. Our approach employs the parallel conjugate-gradient solver from the Eigen library to solve the variational-refinement problem (2). We used the following algorithm parameters in all of our experiments: $\alpha = 0.5$, $\lambda = 200$, $\sigma^{PB} = 0.002$, $N_{of} = 10$, $N_{fp} = 5$, $N_{SOR} = 25$, $N_{ref} = 1$ and $N_{CG} = 50$. The running time was measured on a laptop with a quad-core 2.9GHz Intel i7-7820HQ CPU and 16GB of memory.

We used the DAVIS data set [3] to estimate the video-completion performance under challenging conditions, as well as to compare our results with those of Huang et al. [2] using the provided object masks (which also include shadows). We used the published video-completion results of [1] and [2] on 26 test sequences at 480p resolution and applied our method to the same input data. Le et al. [9] have only published results for their method using raw masks that exclude object shadows, making these results hard to compare with others. To numerically assess the completion quality we performed a subjective pairwise comparison using the Subjectify.us web service. In this comparison, 63 participants were shown pairs of video-completion results produced by different methods; for each pair we asked them



(a) "Sign" (VideoCompletion.org) (b) "Horsejump-low" (DAVIS data set)

Fig. 4. Effect of variational refinement (a) and illumination adjustment (b) in our method.

(a) Source (b) Le et al. [9] (c) Ours

Fig. 5. Our algorithm may fail to reconstruct complex background-motion fields with several moving objects. Global spatiotemporal-optimization approaches like those in [2] and [9] are advantageous in such cases.

to select the result with the best visual quality. We transformed the pairwise comparison results into subjective scores using the Crowd Bradley-Terry model. According to the subjective-comparison results, our method provides completion quality similar to that of [2] (see Figure 1). We also published complete sequences illustrating our method's results at `http://videocompletion.org/fast_video_completion`. Compared with existing video-completion approaches that take hours to process a few seconds of 480p video, our approach is easily scalable to 4K (see Figure 3). Moreover, variational refinement accounts for a large fraction of the computation time but in many cases provides only a minor quality improvement. Figure 4 illustrates the effect of different algorithm components on completion quality.

4.1. Limitations

Although variational refinement helps to correct small optical-flow-completion errors, our method is unable to recover from major errors in motion-field reconstruction (see Figure 5). Spatiotemporal-optimization approaches can employ information from many surrounding frames to get a better reconstruction compared with our two-frame optical flow, but it comes at the cost of much higher computational complexity, as we discussed earlier.

5. CONCLUSION

We proposed a computationally efficient video-completion algorithm that provides competitive results for a wide range of input sequences involving a freely moving camera, although methods based on global spatiotemporal optimization have an

advantage in cases of complex background motion and multiple objects moving in the missing region. The high speed of our approach is enabled by its novel algorithm for fast joint optical-flow estimation and completion, as well as the separable approximation we used in the proposed Poisson-blending modification.

6. REFERENCES

[1] A. Newson, A. Almansa, M. Fradet, Y. Gousseau, and P. Pérez, "Video inpainting of complex scenes," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 1993–2019, 2014.

[2] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf, "Temporally coherent completion of dynamic video," *ACM Transactions on Graphics*, vol. 35, no. 6, pp. 196:1–196:11, 2016.

[3] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 724–732.

[4] S. Ilan and A. Shamir, "A survey on data-driven video completion," in *Computer Graphics Forum*, 2015, vol. 34, pp. 60–85.

[5] Y. Wexler, E. Shechtman, and M. Irani, "Space-time video completion," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2004, vol. 1, pp. I–120–I–127.

[6] T. Shiratori, Y. Matsushita, X. Tang, and S.-B Kang, "Video completion by motion field transfer," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006, vol. 1, pp. 411–418.

[7] M. Liu, S. Chen, J. Liu, and X. Tang, "Video completion via motion guided spatial-temporal global optimization," in *ACM International Conference on Multimedia*, 2009, pp. 537–540.

[8] Z. Xu, Q. Zhang, Z. Cao, and C. Xiao, "Video background completion using motion-guided pixel assignment optimization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 8, pp. 1393–1406, 2016.

[9] T. T. Le, A. Almansa, Y. Gousseau, and S. Masnou, "Motion-consistent video inpainting," in *IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 2094–2098.

[10] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Transactions on Graphics*, vol. 28, no. 3, pp. 24:1–24:11, 2009.

[11] K. He and J. Sun, "Computing nearest-neighbor fields via propagation-assisted KD-trees," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 111–118.

[12] C. Barnes, F.-L. Zhang, L. Lou, X. Wu, and S.-M. Hu, "Patchtable: Efficient patch queries for large datasets and applications," *ACM Transactions on Graphics*, vol. 34, no. 4, pp. 97, 2015.

[13] Y. Shen, F. Lu, X. Cao, and H. Foroosh, "Video completion for perspective camera under constrained motion," in *International Conference on Pattern Recognition*, 2006, vol. 3, pp. 63–66.

[14] M. Granados, K. I. Kim, J. Tompkin, J. Kautz, and C. Theobalt, "Background inpainting for videos with dynamic objects and a free-moving camera," in *European Conference on Computer Vision*, 2012, pp. 682–695.

[15] M. Strobel, J. Diebold, and D. Cremers, "Flow and color inpainting for video completion," in *German Conference on Pattern Recognition*, 2014, pp. 293–304.

[16] M. Roxas, T. Shiratori, and K. Ikeuchi, "Video completion via spatio-temporally consistent motion inpainting," *IPSJ Transactions on Computer Vision and Applications*, vol. 6, pp. 98–102, 2014.

[17] A. Bokov and D. Vatolin, "Toward efficient background reconstruction for 3D-view synthesis in dynamic scenes," in *IEEE International Conference on Multimedia & Expo Workshops*, 2017, pp. 37–42.

[18] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," in *ACM Transactions on Graphics*, 2003, vol. 22, pp. 313–318.

[19] K. He and J. Sun, "Statistics of patch offsets for image completion," in *European Conference on Computer Vision*, pp. 16–29. 2012.

[20] A. Telea, "An image inpainting technique based on the fast marching method," *Journal of Graphics Tools*, vol. 9, no. 1, pp. 23–34, 2004.

[21] S. Liu, L. Yuan, P. Tan, and J. Sun, "Steadyflow: Spatially smooth optical flow for video stabilization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 4209–4216.

[22] T. Kroeger, R. Timofte, D. Dai, and L. Van Gool, "Fast optical flow using dense inverse search," in *European Conference on Computer Vision*, 2016, pp. 471–488.

[23] D. Min, S. Choi, J. Lu, B. Ham, K. Sohn, and M. N. Do, "Fast global image smoothing based on weighted least squares," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5638–5653, 2014.