Enhancing and Experiencing Spacetime Resolution with Videos and Stills

Ankit Gupta
University of Washington

Pravin Bhat
University of Washington

Mira Dontcheva
Adobe Systems

Oliver Deussen
University of Konstanz

Brian Curless
University of Washington

Michael Cohen
Microsoft Research

Abstract

We present solutions for enhancing the spatial and/or temporal resolution of videos. Our algorithm targets the emerging consumer-level hybrid cameras that can simultaneously capture video and high-resolution stills. Our technique produces a high spacetime resolution video using the high-resolution stills for rendering and the low-resolution video to guide the reconstruction and the rendering process. Our framework integrates and extends two existing algorithms, namely a high-quality optical flow algorithm and a high-quality image-based-rendering algorithm. The framework enables a variety of applications that were previously unavailable to the amateur user, such as the ability to (1) automatically create videos with high spatiotemporal resolution, and (2) shift a high-resolution still to nearby points in time to better capture a missed event.

1. Introduction

Still cameras are capturing increasingly high-resolution images. In contrast, video resolution has not increased at nearly the same rate. This disparity in the two mediums is not surprising as capturing high-resolution images at a high frame rate is a difficult and expensive hardware problem. Video produces enormous amounts of data, which must be captured quickly using smaller exposure times.

While newer digital SLR cameras can also capture high quality video, these cameras are still priced at the higher end of consumer level cameras and leave a significant resolution gap between photographs and videos. For example, the Nikon D90 can capture 12 MP images at 4 fps and 720P HD video (0.9 MP per frame) at 24 fps. The problem of capturing videos with high spatiotemporal resolution is further compounded by the constant push in the consumer market for miniaturization and integration of cameras with other products (e.g., cell phones, PDAs). Hence, high spatial resolution imagery is often incompatible with high frame rate imagery, especially in the case of consumer level cameras, due to bandwidth and storage constraints.

In the face of these realities, we investigate software solutions that can increase the spatial and/or temporal resolution of imagery recorded by *hybrid cameras* capable of capturing a combination of low-resolution video at medium frame rates (15-30 fps) and high-resolution stills at low frame rates (1-5 fps). Such hybrid cameras have been previously proposed [10] and several have been built as research prototypes [18, 1, 27, 16]. Commercial hybrid cameras are currently available (e.g., Sony HDR-HC7, Canon HV10, and Canon MVX330), and while these cameras have some limitations,[1] newer models hold substantial promise; e.g., Fujifilm has announced the development of the *Finepix 3D System*,[2] which has two synchronized lenses and sensors capable of capturing stills and video simultaneously.

We propose a framework for combining the output of hybrid cameras. Our framework combines and extends two existing algorithms, namely a high-quality optical flow algorithm [22] and a high-quality image-based-rendering algorithm [2] to enable a variety of applications that were previously unavailable to the amateur user, including:

- automatically producing high spatiotemporal resolution videos using low-resolution, medium-frame-rate videos and intermittent high-resolution stills,
- time-shifting high-resolution stills to nearby points in time to better capture a missed event.

We also describe a simple two-step flow algorithm that improves the quality of long-range optical flow in our setting (i.e., low-resolution video plus a few high-resolution stills). We demonstrate results using a simulated hybrid camera – simulated by downsampling existing video spatially and separately sub-sampling it temporally – and using our own

[1] The number of stills that these cameras can capture during a video session is currently limited to a maximum of three at one fps. In addition, we found the high-resolution stills produced by these cameras to not be significantly higher in quality than the video frames.

[2] http://www.dpreview.com/news/0809/08092209fujifilm3D.asp

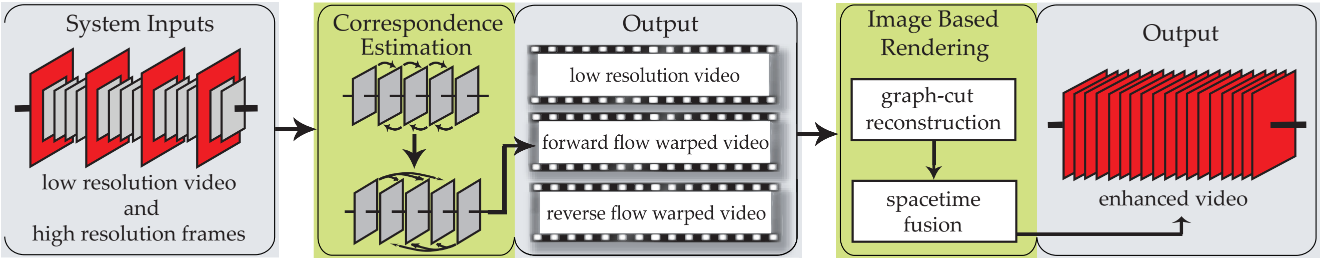


Figure 1. Our system consists of two main components. We use a two-step optical flow process to compute correspondences between pixels in a low-resolution frame and nearby high-resolution stills. We employ the image based rendering algorithm of Bhat *et al.* [2007] to render the final result.

prototype still+video camera. Results of a user study on time-shifting suggest that people would be quite interested in using this technology to capture better imagery. In the remaining sections, we describe related work (Section 2), present our framework and algorithms (Section 3), explore applications (Section 4), and conclude with a summary and discussion (Section 5).

2. Related work

The problem of enhancing the resolution of images and videos has received much attention. In this section we review some of the previous approaches to this problem.

Spatial and temporal super-resolution using multiple low-resolution inputs. These techniques perform spatial super-resolution by first aligning multiple low-resolution images of a scene at sub-pixel accuracy to a reference view [13, 25, 11]. The aligned images can then be used to reconstruct the reference view at a higher spatial resolution. The same idea can be applied in the temporal domain by using sub-frame-rate shifts across multiple low temporal resolution videos of a scene to perform temporal super-resolution [26, 31, 32].

The techniques in this class rely on the assumption that the low-resolution inputs are undersampled and contain aliasing. However, most cameras usually bandlimit the high frequencies in a scene to minimize such aliasing, which severely limits the amount of resolution enhancement that can be achieved using such techniques. Lin *et al.* [17] have studied the fundamental limits of these techniques and found their magnification factor to be at most 1.6 in practical conditions. Also, the reliance of these methods on static scenes or multiple video cameras currently limits the practicality of these methods. We overcome these limitations by using data from a hybrid camera.

Temporal resolution enhancement. For temporal resolution enhancement, most techniques [8, 7, 33, 9] compute motion correspondences using optical flow and perform a weighted average of motion-compensated frames assuming linear motion between correspondences. However, since the intermediate frame is generated as a weighted average of the two warped images, any errors in the correspondences can result in artifacts such as ghosting, blurring and flickering in the final result. We use a similar technique for generating correspondences, but employ a graph-cut compositing and a spacetime gradient-domain compositing process to reduce these artifacts.

Learning-based techniques. These regression based techniques [12] learn a mapping between a patch of upsampled low-resolution pixels and the corresponding high-resolution pixel at the center of the patch to create a dictionary of patch-pairs. The high resolution image is synthesized using the patch of low-resolution pixels surrounding every pixel in the input to infer the corresponding high-resolution pixel from the dictionary. Often, the synthesis also incorporates some smoothness prior to avoid artifacts that commonly arise from relying on regression alone. Bishop *et al.* [3] proposed a similar technique for enhancing videos with the patch-pairs from previously enhanced video frame also added to the dictionary which helps in temporal coherence. While using dictionaries of examples is a general technique that can "hallucinate" high frequencies, it can also lead to artifacts that are largely avoidable with hybrid cameras, where the corresponding high frequency information is often captured in a different frame.

Combining stills with videos. Our technique can best be described as reconstructing the high-resolution spacetime video using a few high-resolution images for rendering and a low-resolution video to guide the reconstruction and the rendering. Bhat *et al.* [2] and Schubert *et al.* [24] proposed a similar approach to enhance low-resolution videos of a static scene by using multi-view stereo to compute correspondences between the low-resolution video and the high-resolution images. In contrast, our method uses optical flow to compute correspondences and can therefore handle dynamic scenes as well. Also, Schubert *et al.* [24] did not enforce any priors for temporal coherence across the generated video.

Sawhney *et al.* [23] use a stereo camera which captures two synchronized videos from different viewpoints, one video at the desired high resolution and another at a low resolution. They use stereo based image-based rendering techniques to increase the spatial resolution of the second sequence by combining the two streams. Compared to our method, this method is more restrictive as it needs to capture one video at high spacetime resolution and it requires multiple synchronized cameras.

MPEG video coding works on a similar domain by storing keyframes at full resolution and using block-based motion vectors to predict the intermediate frames. Our system produces results significantly better than simply using a MPEG-based frame prediction scheme since we use per-pixel flow estimation and a high quality rendering algorithm that suppresses artifacts in regions with faulty flow estimates.

The method proposed by Watanabe *et al.* [30] works on similar input data as ours (i.e., a low-resolution video with a few intermittent high-resolution frames). Each high-resolution frame is used to propagate the high frequency information to the low-resolution frames using a DCT fusion step. Nagahara *et al.* [19] also take a similar approach but use feature tracking instead of motion compensation. These methods generate each video frame independently and therefore are prone to temporal incoherence artifacts. See Section 4.1.1 for a comparison to our method.

3. System overview

Figure 1 gives a visual description of our system for performing spatial and/or temporal resolution enhancement of videos using high-resolution stills when available.

3.1. Spatial resolution enhancement

The input consists of a stream of low-resolution frames with intermittent high-resolution stills. We upsample the low-resolution frames using bicubic interpolation to match the size of the high-resolution stills and denote them by f_i. For each f_i, the nearest two high-resolution stills are denoted as S_{left} and S_{right}.

Computing motion correspondences. The system estimates motion between every f_i and corresponding S_{left} & S_{right}. Unfortunately, computing correspondences between temporally distant images of a dynamic scene is a hard problem. Most optical flow algorithms can compute correspondences for motion involving only tens of pixels. In our case the system needs to compute correspondences between a high-resolution still and a low resolution frame that might contain objects displaced over hundreds of pixels.

One approach is to compute optical flow directly from the high-resolution stills, S_{left} or S_{right}, to the upsampled frames f_i. This approach, however, produces errors because of the long range motion and the differences in image resolution. The flow quality can be improved by first filtering S_{left} and S_{right} to match the low resolution of the video frames. We will denote these filtered images by f_{left} and f_{right} respectively. This improves matching, but the flow algorithm is still affected by errors from the long range motion. Instead of computing long range flow, one could compute pairwise optical flow between consecutive video frames and sum the flow vectors to estimate correspondences between distant frames. This approach performs better, but flow errors between consecutive frames tend to accumulate.

Our approach is to use a two-step optical flow process. First, we approximate the long range motion by summing the forward and backward optical flow between adjacent low-resolution frames. This chains together the flow from f_{left} forward to each frame until f_{right}, and similarly from f_{right} backward in time. Then, we use these summed flows to initialize a second optical flow computation from f_{left} to f_i and from f_{right} to f_i. The summed motion estimation serves as initialization to bring long range motion within the operating range of the optical flow algorithm and reduces the errors accumulated from the pairwise sums. See Figure 4 for a comparison between our method for computing correspondences and the alternatives mentioned above.

In our two-stage process, we employ the optical flow algorithm of Sand *et al.* [22]. Sand's optical flow algorithm combines the variational approach of Brox *et al.* [5] with Xiao *et al.*'s [34] method for occlusion handling. We used Sand's original implementation of the algorithm and its default parameter settings for all of our experiments.

Graph-cut compositing: Once the system has computed correspondences from S_{left} to f_i and S_{right} to f_i, it warps the high-resolution stills to bring them into alignment with f_i thus producing two warped images, w_{left} and w_{right}. Then it reconstructs a high-resolution version of f_i using patches from w_{left} and w_{right}. The reconstruction is computed using a multi-label graph-cut optimization with a metric energy function [4]. Each pixel in f_i is given label from three candidates: w_{left}, w_{right}, and f_i. We use the standard energy function used for graph-cut compositing with a data cost that is specialized for our problem and the smoothness cost proposed by Kwatra *et al.* [15]. Kwatra's smoothness cost encourages the reconstruction to use large coherent regions that transition seamlessly from one patch to another.

Our data cost encourages the reconstruction to prefer labels that are likely to produce a high-resolution reconstruction while trying to avoid artifacts caused by errors in the correspondences.

The formal definition of our data cost function D for computing the cost of assigning a given label l to a pixel p is as follows:

$$D(p,l) = \begin{cases} c & \text{if } l = f_i \\ \infty & \text{if } w_l(p) \\ & \text{ undefined} \\ D_c(p,l) + D_f(p,l) + D_d(l,f_i) & \text{otherwise} \end{cases}$$

$$D_c(p,l) = ||w_l(p) - f_i(p)||$$
$$D_f(p,l) = 1 - \text{motion\_confidence}(w_l(p))$$
$$D_d(l,f_i) = \frac{|\text{frame\_index}(w_l) - i|}{|\text{frame\_index}(w_{\text{right}}) - \text{frame\_index}(w_{\text{left}})|}$$

Here, c is the fixed cost for assigning a pixel to the low-resolution option (i.e, f_i); w_l is the warped image corresponding to the label l; D_c encourages color consistency between a pixel and its label; D_f factors in the confidence of the motion vector that was used to generate the pixel $w_l(p)$; and D_d favors labels that are closer in temporal distance to the current frame number i. All examples in this paper were generated by setting c to 0.3. The pixel channel values are in the range [0..1], and the confidence values (also in the range [0..1]) for the motion vectors are generated by Sand's optical flow algorithm in the process of computing the correspondences. The confidence value for a motion vector to a pixel can also be understood as a probabilistic occlusion map where low confidence value means that there is a high probability that the pixel is occluded. We use a threshold of 0.1 on this map to determine occluded pixels and w_l is considered undefined at those locations.

Spacetime fusion. When each individual frame in the video has been reconstructed using the graph-cut compositing step described above, the resulting video has high spatial resolution but it suffers from the types of artifacts common to videos reconstructed using pixel patches – that is, the spatial and temporal seams between the patches tend to be visible in the result. These spatial seams can often be mitigated using the 2D gradient domain compositing technique described by Pérez *et al.* [20]. However, the temporal seams that arise due to errors in the motion vectors and exposure/lighting differences in the high-resolution stills can be difficult to eliminate.

We use the spacetime fusion algorithm proposed by Bhat *et al.* [2], which is a 3D gradient domain compositing technique that can significantly reduce or eliminate both spatial and temporal seams. Spacetime fusion takes as input the spatial gradients of the high-resolution reconstruction and the temporal gradients of the low-resolution video (computed along flow lines) and tries to preserve them simultaneously. Thus, the temporal coherence captured in the low-resolution video is reproduced in the final high-resolution result, while preserving the high spatial resolution information as well as possible. We can assign relative weights to the spatial and temporal gradients. Using only spatial gradients leads to high spatial but poor temporal quality. Using only temporal gradients leads to too much blurring in the spatial domain. In all of our experiments for spatiotemporal resolution enhancement, we set the temporal gradient constraint weight to 0.85 and thus the spatial gradient constraint weight is 0.15. The reader is referred to Bhat's spacetime fusion paper [2] for further details.

3.2. Temporal resolution enhancement

The temporal resolution of any video can be increased given good flow estimation between frames. To increase the temporal resolution of a video by some factor, we insert the appropriate number of intermediate frames between existing frames. To estimate flow vectors between neighboring frames (we'll call these neighbors "boundary frames" for the rest of this section) and the new intermediate frame, we assume that the motion varies linearly between the frames. The system simply divides the flow across the boundary frames evenly between new intermediate frames (e.g., with three intermediate frames, the flow to the first intermediate frame is 1/4 of the flow between boundary frames, 1/2 to the second and 3/4 to the third intermediate frame). Then, the two boundary frames are warped to the appropriate point in time and composited using the graph-cut compositing step described in Section 3.1 to construct the intermediate frame. Note that in this compositing step, the data cost D is defined as the sum of D_f and D_d only. D_c is not used since it depends on the original low-resolution frame which does not exist.

Occlusions in the scene cause holes in the reconstruction. Previously, we used the low-resolution frames, f_i, to fill these holes. Now, we use spacetime fusion to remove these holes (and other artifacts) by assuming all spatial gradients inside holes are null, which allows surrounding information to be smoothly interpolated within holes. For the spacetime fusion step, the system needs temporal gradients between the new intermediate frames. Just as we assumed motion varies linearly between video frames to compute the intermediate flow vectors, we assume temporal color changes vary linearly along flow lines. To estimate the temporal gradient between two intermediate frames we divide the temporal gradient between the boundary frames by one more than the number of new intermediate frames.

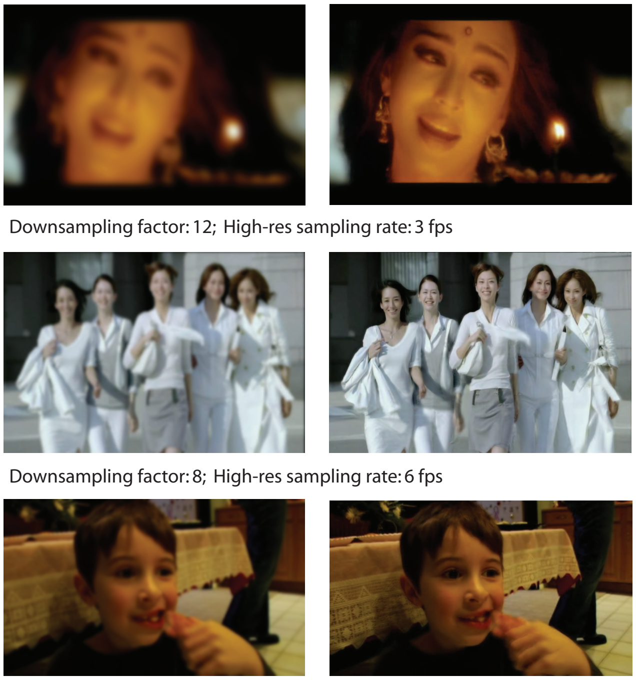
4. Applications and analysis

In this section we explore several consumer-level applications that could benefit from the ability to combine low-resolution video with high-resolution stills to produce videos with high spatiotemporal resolution.

4.1. Spatial resolution enhancement

We first explore enhancing the spatial resolution of videos containing complex scenes, non-rigid deformations, and dynamic illumination-effects. As mentioned in the introduction, commercially available hybrid cameras do not yet capture stills at the spatial and temporal resolution required by our system. To evaluate our approach we use two types of datasets – simulated and real.

We simulated the output of a hybrid camera by downsampling high-resolution videos and using high-resolution frames from the original video at integer intervals. Figure 2 shows results on three such datasets. Please see the supplementary video for video results.


Downsampling factor: 12; High-res sampling rate: 3 fps

Downsampling factor: 8; High-res sampling rate: 6 fps

Downsampling factor: 8; High-res sampling rate: 3 fps

Figure 2. The left column shows the low-resolution input video frame. The right column shows our result. We suggest zooming in to see improvements at the actual scale of the images.

We also created a few more datasets by simultaneously capturing a scene using a camcorder and a digital SLR placed in close proximity. The camcorder captured the scene at 0.3 megapixel resolution at 30 fps, while the SLR captured the scene at six megapixel resolution at three fps. For our experiments, we need to align these two streams in color, space and time. First, we match the color statistics of the two streams in LAB color space [21]. Then we compensate for differences in camera positions and fields of view by computing a homography between a photograph and a corresponding video frame. We apply this homography to all frames in order to spatially align them with the photographs. Finally for temporal alignment of data streams, we computed SIFT features for photographs and frames and formulated a simple dynamic programming problem to match the photographs with the frames using the SIFT features, while maintaining their temporal ordering. Figure 3 shows results on these datasets. We provide the corresponding video results in supplementary material.

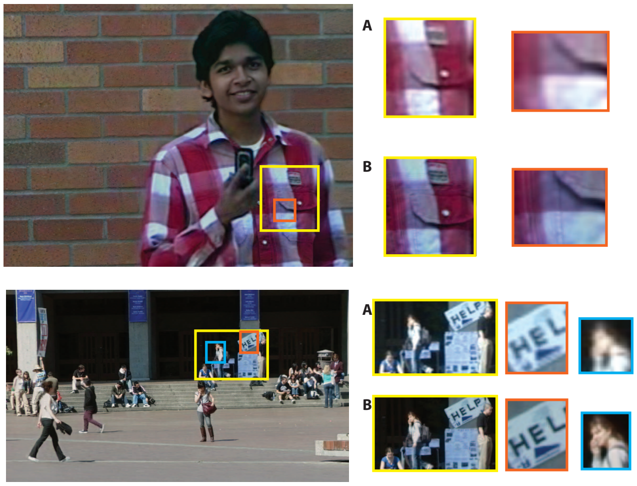


Figure 3. The figure shows spatial resolution enhancement results for hybrid data captured using a two-camera setup. The left column shows a result frame for each of the two data sets, (A) shows zoomed in parts from low-resolution input frame, and (B) shows the corresponding parts of result frame. Zoom in to see the resolution difference more clearly.

4.1.1 Qualitative and quantitative analysis

In this subsection we provide some qualitative and quantitative analysis of our method for spatial resolution enhancement.

As discussed in Section 3.1 there are a number of techniques for computing the motion correspondences between the boundary and intermediate frames. In Figure 4 we compare these techniques using a challenging scenario that includes water spray and fast motion. We improve the spatial

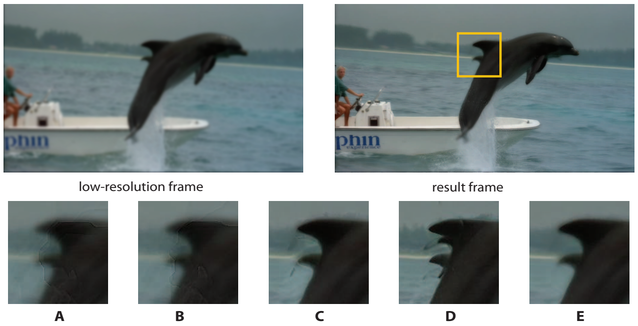


Figure 4. The figure provides qualitative comparisons of our system to alternate approaches. (A) Computing flow directly between high-resolution boundary stills and low-resolution intermediate frames produces ghosting and blurring artifacts due to the resolution difference and long range motion. (B) Computing flow between low-resolution version of the boundary stills and low-resolution intermediate frames results in similar artifacts due to long range motion. (C) Summed up pairwise optical flow to estimate the motion between distant frames performs better but still suffers from motion trails. (D) and (E) use our two-step optical flow approach but use different rendering styles. (D) Taking a weighted average of the warped boundary-frames results in tearing artifacts. (E) Our rendering approach produces a video that is visually consistent and has relatively few artifacts. We suggest zooming in to see improvement at the actual scale of the images. Note that (A)-(C) and (E) use our graph-cut spacetime fusion rendering; And (D)-(E) use our two-step optical flow process.

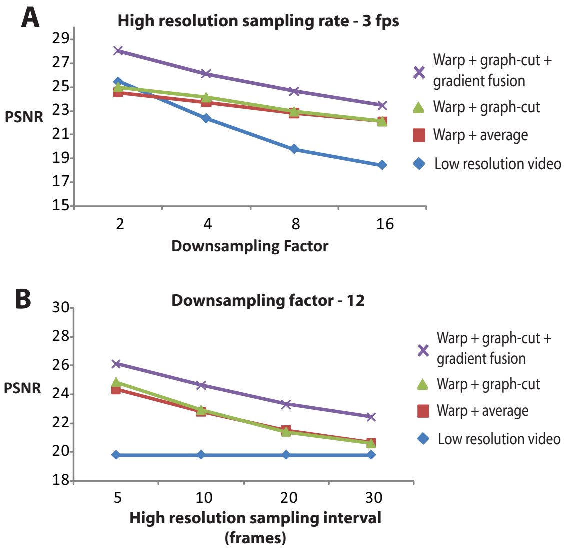


Figure 5. (A) Variation of PSNR with respect to the spatial downsampling factor of the low-resolution frames (B) Variation of PSNR with respect to the sampling rate of high-resolution keyframes

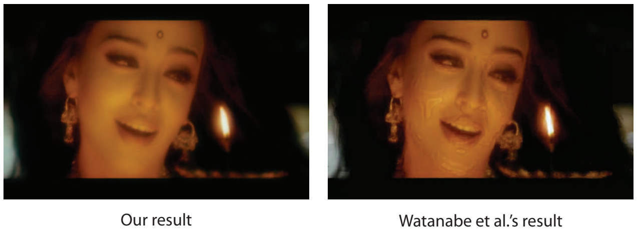


Figure 6. Comparison with DCT based frequency fusion method of Watanabe *et al*. [2006]. One can clearly see the ringing and blocking artifacts in regions where motion correspondences are incorrect. Zoom in to see the differences more clearly.

resolution by a factor of 12 in each dimension and show a qualitative comparison between our two-step optical flow approach (Figure 4E) and three alternatives (Figure 4A-C). We also compare our rendering technique (Figure 4E) to a naive morphing based composite (Figure 4D). Our approach, while not perfect, has fewer artifacts than any of the alternatives. The artifacts are seen in regions of the video where optical flow fails. Optical flow assumes brightness constancy, and errors can arise when this assumption is violated. A notable example is the singer's face in Figure 2, where the flickering flame illuminates her face; the resulting flow errors result in a smooth warping artifact in the rendered result. Occlusions and large motion also cause error in optical flow. In these regions our algorithm copies information from the low-resolution frame during the graph-cut composition. Any residual error is distributed across the video using spacetime fusion in order to further reduce the artifacts.

Figure 5 shows a quantitative analysis of our spatially enhanced results by measuring the overall peak signal-to-noise ratio (PSNR) of the result video with respect to the original high-resolution video. PSNR is widely used as a compression quality metric for images and videos. In this analysis we use the dolphin sequence shown in Figure 4. We explore PSNR variation with respect to the downsampling factor (Figure 5A) and the sampling interval of the high-resolution stills (Figure 5B) used to simulate the hybrid camera data. Figure 5A shows that as the resolution of the input video decreases, the PSNR also decreases. This correlation is not surprising, as the resolution of the input video will invariably affect the quality of the motion correspondences. Figure 5B shows that as the sampling interval of the high-resolution stills increases, i.e., there are fewer high resolution stills, the PSNR also decreases. These figures also compare the performance of warping the boundary frames and then (1) performing a weighted blend of the warped frames, (2) creating a graph-cut composite, or (3) creating a graph-cut composite followed by gradient domain integration (our method). The figures show that our method outperforms the other methods.

We now compare our system to Watanabe *et al.*'s system [30] for increasing the spatial resolution of a low-resolution stream using periodic high-resolution frames through frequency spectrum fusion. We implemented their approach and show the comparison in Figure 6 and in the supplementary video. Their method processes each intermediate frame separately and does not ensure temporal coherence. By comparison, the spacetime fusion step in our framework adds coherence and mitigates artifacts due to bad motion correspondence. Our graph-cut based image compositing uses the flow confidence for assigning appropriate labels to pixels. Watanabe's frequency spectrum fusion technique fuses the high frequency content everywhere leading to artifacts where motion correspondences are bad. Further, their use of block DCTs introduces ringing and blocking artifacts.

4.2. Time shift imaging

Although advances in digital cameras have made it easier to capture aesthetic images, photographers still need to know when to press the button. Capturing the right instant is often elusive, especially when the subject is an exuberant child, a fast-action sporting event, or an unpredictable animal. Shutter delay in cameras only exacerbates the problem. As a result, users often capture many photographs or use the motor drive setting of their cameras when capturing dynamic events.

Given the input from a hybrid camera, our system can help alleviate this timing problem. We assume the hybrid camera is continuously capturing a low-resolution video and periodically capturing high-resolution stills. When the user "takes a picture," the camera stores the video interval between the last and next periodic high-resolution stills and also three high-resolution stills (two periodic captures and the one clicked by the user which is positioned at some random time moment between the periodic stills). Using our spatial enhancement approach, we can propagate the high-resolution information from the high-resolution still to the surrounding low-resolution frames, thus producing a very short high-resolution video around the high-resolution still captured by the user. This high-resolution image collection enables users to choose a different high-resolution frame than the one originally captured. We envision that the ability to shift a still backward or forward in time will make it easier to capture that "blowing out the candles" moment, or that "perfect jumpshot." Figure 7 shows a result for this application.

To assess the utility of time-shifting, we performed a user study. In the study, 17 users were shown 22 videos and, for each video, were asked to indicate with a keystroke when they would like to take a snapshot of the action. Afterwards, they were given the opportunity to replay each video with a time slider to indicate which frame they had intended to capture. On average, our participants intended to select a frame different from their snapshots 85.7% of the time. Further, informal comments from these users revealed that they are looking forward to a time-shifting feature in their camera to help them capture the moment of their choice.

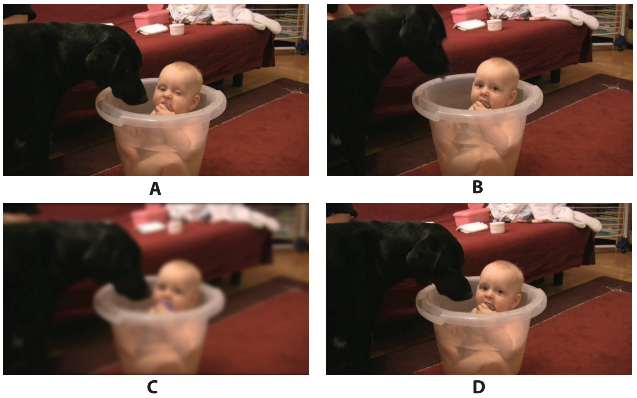


Figure 7. Images (A) and (B) show high-resolution frames captured by the hybrid camera. Image (C) shows an intermediate low-resolution video frame. Note that the eyes of the baby are closed in (A) and the dog's snout is far away in (B). A more photographic moment occurs in (C), where the eyes are open and the snout is close to the face. Our system generates a high spatial resolution frame for (C) as shown in (D) by flowing and compositing high-resolution information from (A) and (B).

4.3. Temporal resolution enhancement

As mentioned in Section 3.2, by assuming linear motion between consecutive video frames, our system can insert an arbitrary number of intermediate frames to enhance the temporal resolution of a video. We show some results in the supplementary video. Most previous methods for temporal resolution enhancement [28] take this same approach. Our method differs in the rendering phase.

Existing techniques use weighted averaging of warped frames to hide artifacts resulting from bad correspondences. In comparison, our rendering method (i.e., graph-cut compositing plus spacetime fusion) focuses on preserving the overall shape of objects and hence leads to stroboscopic motion in regions where the motion correspondences are extremely bad. This can be observed for the motion of the player's hand in the cricket clip (in supplementary video). Therefore, the artifacts of our technique are different from those of previous techniques in regions where flow fails. Our technique may be preferred for enhancing videos that involve objects like human faces where distortion and tearing artifacts would be jarring. On the other hand, the distortion and tearing artifacts common in previous methods look like motion blur for certain objects, which in turn makes their results look temporally smoother for regions with bad flow. Like most previous methods, our method does not remove motion blur present in the input video for fast moving objects, which can seem perceptually odd when these fast moving objects are seen in slow motion.

We can also combine the spatial and temporal steps of our approach to produce spatiotemporally enhanced videos. We show several examples of temporal and spatiotemporal resolution enhancement in the supplementary video. The ability to insert intermediate frames also allows the users to speed-up or slow-down different parts of a video using a simple curve-based interface; a demo of which is also shown in the supplementary video.

4.4. Computational cost

We compute pairwise optical flow for a 640x480 frame in 3 minutes with a 3.4 GHz processor and 4GB RAM. Doing forward and backward computation in each of the two steps requires 4 optical flow computations per frame. In practice, we compute optical flows in parallel on multiple machines. The graph-cut compositing for each frame takes around 30 seconds. The spacetime fusion is performed on the whole video using a simple conjugate gradient solver and the average time per frame is also around 30 seconds. We expect that this performance could be dramatically improved with multi-grid solvers and GPU acceleration [6, 14, 29].

5. Conclusion

We have demonstrated the power of combining an optical flow algorithm with an image-based-rendering algorithm to achieve numerous enhancements to the spatial and temporal resolution of consumer-level imagery. We hope that our work will encourage camera manufacturers to provide more choices of hybrid cameras that capture videos and stills simultaneously. The combination of good hybrid cameras and software solutions like ours can further bridge the quality gap between amateur and professional imagery.

Currently the capabilities of our framework depend on the quality of motion correspondences produced by the optical flow algorithm. Unlike previous work, our method fails gracefully in regions where flow fails, by defaulting to the low-resolution video in those regions. One improvement would be to fill in detail with an learning-based super-resolution approach [12] by using the intermittent high-resolution stills as training data. Improving the flow algorithm would also help, of course; as motion correspondence algorithms improve, we will be able to apply our framework to a broader set of scenarios, such as videos with larger and faster motion. We also envision using our framework to increase the temporal resolution of high-resolution stills captured in motor-drive on an SLR. Additionally, the capability to generate high spacetime resolution videos from hybrid input could possibly be used as a form of video compression (i.e., storing just the hybrid video and doing decompression by synthesis only when needed).

Acknowledgments

We thank Peter Sand for providing the implementation of his optical flow algorithm. We would also like to thank Eli Shechtman and the reviewers for their insightful comments. This work was supported by NVIDIA and Microsoft Research fellowships and by funding from the University of Washington Animation Research Labs, Microsoft, Adobe, and Pixar.

References

[1] M. Ben-Ezra and S. K. Nayar. Motion-based motion deblurring. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):689–698, 2004. 1

[2] P. Bhat, C. L. Zitnick, N. Snavely, A. Agarwala, M. Agrawala, B. Curless, M. Cohen, and S. B. Kang. Using photographs to enhance videos of a static scene. In *EGSR '07: Proc. of Eurographics Symposium on Rendering*, pages 327–338. Eurographics, June 2007. 1, 2, 4

[3] C. Bishop, A. Blake, and B. Marthi. Super-resolution enhancement of video. *Proc. of the 9th Conference on Artificial Intelligence and Statistics (AISTATS)*, 2003. 2

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001. 3

[5] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *ECCV '04: Proc. of the 8th European Conference on Computer Vision*, volume 4, pages 25–36, 2004. 3

[6] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnorr. Variational optical flow computation in real time. *IEEE Transactions on Image Processing*, 14(5):608–615, May 2005. 8

[7] C. Cafforio, F. Rocca, and S. Tubaro. Motion compensated image interpolation. *IEEE Trans. on Communications*, 38(2):215–222, Feb 1990. 2

[8] R. Castagno, P. Haavisto, and G. Ramponi. A method for motion-adaptive frame rate up-conversion. *IEEE Trans. Circuits and Systems for Video Technology*, 6(5):436–446, October 1996. 2

[9] B. Choi, S. Lee, and S. Ko. New frame rate up-conversion using bi-directional motion estimation. *IEEE Trans. on Consumer Electronics*, 46(3):603–9, 2000. 2

[10] M. F. Cohen and R. Szeliski. The moment camera. *IEEE Computer*, 39(8):40–45, 2006. 1

[11] M. Elad and A. Feuer. Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. *IP '97: IEEE Transactions on Image Processing*, 6(12):1646–1658, Dec. 1997. 2

[12] W. T. Freeman, T. R. Jones, and E. C. Pasztor. Example-based super-resolution. *IEEE Computer Graphics and Applications*, 22(2):56–65, 2002. 2, 8

[13] M. Irani and S. Peleg. Improving resolution by image registration. *CVGIP: Graphical Models and Image Processing*, 53(3):231–239, 1991. 2

[14] M. Kazhdan and H. Hoppe. Streaming multigrid for gradient-domain operations on large images. *SIGGRAPH '08, ACM Transactions on Graphics*, 27(3):1–10, 2008. 8

[15] V. Kwatra, A. Schődl, I. Essa, G. Turk, and A. Bobick. Graphcut textures: Image and video synthesis using graph cuts. *SIGGRAPH '03, ACM Transactions on Graphics*, 22(3):277–286, July 2003. 3

[16] F. Li, J. Yu, and J. Chai. A hybrid camera for motion deblurring and depth map super-resolution. In *CVPR '08: Proc. of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008. 1

[17] Z. Lin and H.-Y. Shum. Fundamental limits of reconstruction-based superresolution algorithms under local translation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):83–97, 2004. 2

[18] H. Nagahara, A. Hoshikawa, T. Shigemoto, Y. Iwai, M. Yachida, and H. Tanaka. Dual-sensor camera for acquiring image sequences with different spatio-temporal resolution. In *AVSBS '05: Advanced Video and Signal Based Surveillance*, pages 450–455, 2005. 1

[19] H. Nagahara, T. Matsunobu, Y. Iwai, M. Yachida, and T. Suzuki. High-resolution video generation using morphing. In *ICPR '06: Proc. of the 18th International Conference on Pattern Recognition*, pages 338–341, Washington, DC, USA, 2006. IEEE Computer Society. 3

[20] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *SIGGRAPH '03: ACM Transactions on Graphics*, pages 313–318, New York, NY, USA, 2003. ACM Press. 4

[21] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):34–41, 2001. 5

[22] P. Sand and S. Teller. Particle video: Long-range motion estimation using point trajectories. In *CVPR '06: Proc. of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2195–2202, Washington, DC, USA, 2006. IEEE Computer Society. 1, 3

[23] H. S. Sawhney, Y. Guo, K. Hanna, R. Kumar, S. Adkins, and S. Zhou. Hybrid stereo camera: an ibr approach for synthesis of very high resolution stereoscopic image sequences. In *SIGGRAPH '01: ACM Transactions on Graphics*, pages 451–460, New York, NY, USA, 2001. ACM. 3

[24] A. Schubert and K. Mikolajczyk. Combining high-resolution images with low-quality videos. In *Proceedings of the British Machine Vision Conference*, 2008. 2

[25] R. Schultz and R. Stevenson. Extraction of high-resolution frames from video sequences. *IP '96: IEEE Transactions on Image Processing*, 5(6):996–1011, June 1996. 2

[26] E. Shechtman, Y. Caspi, and M. Irani. Space-time super-resolution. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(4):531–545, 2005. 2

[27] Y. Tai, H. Du, M. Brown, and S. Lin. Image/video deblurring using a hybrid camera. In *CVPR '08: Proc. of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008. 1

[28] D. Vatolin and S. Grishin. N-times video frame-rate up-conversion algorithm based on pixel motion compensation with occlusions processing. In *Graphicon: International Conference on Computer Graphics and Vision*, pages 112–119, 2006. 7

[29] V. Vineet and P. Narayanan. Cuda cuts: Fast graph cuts on the gpu. *Computer Visions and Pattern recognition Workshops, 2008*, pages 1–8, 2008. 8

[30] K. Watanabe, Y. Iwai, H. Nagahara, M. Yachida, and T. Suzuki. Video synthesis with high spatio-temporal resolution using motion compensation and spectral fusion. *IEICE Transactions on Information and Systems*, E89-D(7):2186–2196, 2006. 3, 7

[31] Y. Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):463–476, 2007. 2

[32] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. *SIGGRAPH '05: ACM Transactions on Graphics*, 24(3):765–776, 2005. 2

[33] W. F. Wong and E. Goto. Fast evaluation of the elementary functions in single precision. *IEEE Trans. on Computers*, 44(3):453–457, 1995. 2

[34] J. Xiao, H. Cheng, H. S. Sawhney, C. Rao, and M. A. Isnardi. Bilateral filtering-based optical flow estimation with occlusion detection. In *ECCV '06: Proc. of the European Conference on Computer Vision*, pages 211–224, 2006. 3