Spacetime Stereo: Shape Recovery for Dynamic Scenes

Li Zhang   Brian Curless   Steven M. Seitz
Department of Computer Science and Engineering
University of Washington, Seattle, WA, 98195
{lizhang, curless, seitz}@cs.washington.edu

Abstract

This paper extends the traditional binocular stereo problem into the spacetime domain, in which a pair of video streams is matched simultaneously instead of matching pairs of images frame by frame. Almost any existing stereo algorithm may be extended in this manner simply by replacing the image matching term with a spacetime term. By utilizing both spatial and temporal appearance variation, this modification reduces ambiguity and increases accuracy. Three major applications for spacetime stereo are proposed in this paper. First, spacetime stereo serves as a general framework for structured light scanning and generates high quality depth maps for static scenes. Second, spacetime stereo is effective for a class of natural scenes, such as waving trees and flowing water, which have repetitive textures and chaotic behaviors and are challenging for existing stereo algorithms. Third, the approach is one of very few existing methods that can robustly reconstruct objects that are moving and deforming over time, achieved by use of oriented spacetime *windows in the matching procedure. Promising experimental results in the above three scenarios are demonstrated.*

1. Introduction

Shape estimation from stereo images has been one of the core challenges in computer vision for decades. The vast majority of stereo research has focused on the problem of establishing spatial correspondences between pixels in a single pair of images for a static moment in time. The appearance of the real world varies over time, due to **lighting changes**, **motion**, and **changes in shading or texture** over time. Traditional stereo algorithms handle these variations by treating each time instant in isolation. In this paper we argue that better results may be obtained by considering how each pixel varies over time and using this variation as a cue for correspondence an approach we call *spacetime stereo*.

The basic principle of spacetime stereo is straightforward. First, consider conventional stereo algorithms, which generally represent correspondence in terms of disparity between a pixel in one image and the corresponding pixel in another image. The matching function used to compute disparities typically compares spatial neighborhoods around candidate pairs of pixels. Spacetime stereo simply adds a temporal dimension to the neighborhoods used in the matching function. The spacetime window can be a rectangular 3D volume of pixels, which is useful for reconstructing scenes in which changes in lighting and appearance, rather than shape changes, dominate. When the object is moving significantly, the disparity must be treated in a time-dependent fashion. In this case, we compute matches based on oriented spacetime windows that allow the matching pixels to shift linearly over time. In either case, the match scores based on spacetime windows are easily incorporated into existing stereo algorithms.

The spacetime stereo approach has several advantages. First, it serves as a simple yet general framework for computing shape when only appearance or lighting changes. These changes may be natural, e.g., imparted by the weathering of materials or the motion of the sun, or they may be artificially induced, as in the case of structured light scanning. We present results specifically for the latter. Second, when shape changes are small and erratic and the appearance has complex semi-repetitive texture, such as one might find when observing a waterfall or a collection of leaves on a tree blowing in the wind, spacetime stereo allows robust computation of average disparities or "mean shapes." Finally, for objects in motion, possibly deforming, the oriented spacetime window matching provides a way to compute accurate disparity maps when standard stereo methods fail. This last case is shown to be particularly effective for structured light scanning of moving scenes.

The rest of the paper is organized as follows. Section 2 discusses prior work both in stereo and in structured light scanning. Section 3 presents spacetime metrics used to evaluate candidate correspondences. Section 4 describes how to adapt existing stereo algorithms to perform spacetime stereo matching. Section 5 presents experimental results. Section 6 concludes with a discussion of the strengths and weakness of the approach.

2. Previous work

Our work builds upon a large literature on stereo correspondence and structured light scanning. Here we summarize the most relevant aspects of this body of work.

Stereo matching has been studied for decades in the computer vision community. We refer readers to Scharstein and Szeliski's recent survey paper [15] for a good overview and comparison of the current state of art. Current stereo algorithms are able to simultaneously solve for smooth depth maps for texture-less regions and model depth discontinuity and occlusion in a principled and efficient way. Current stereo algorithms match two or more images acquired at a single time instant and do not exploit motion cues that exist over time.

Recently, a number of researchers have proposed integrating stereo matching and motion analysis cues, an approach called *motion stereo*. For example, Mandelbaum *et al.* [12] and Strecha and van Gool [16] recover static scenes with a moving stereo rig. Tao *et al.* [17] represent a scene with piecewise planar patches and assume a constant velocity for each plane to constrain dynamic depth map estimation. Vedula *et al.* [18] present a linear algorithm to compute 3D scene flow based on 2D optical flow and estimate 3D structures from the scene flow. Zhang *et al.* [20] compute 3D scene flow and structure in an integrated manner, in which a 3D affine motion model is fit to each local image region and an adaptive global smoothness regularization is applied to the whole image. They later improve their results by fitting parametric motion to each local image region obtained by color segmentation, so that discontinuities are preserved [21]. Carceroni and Kutulakos [4] present a method to recover piecewise continuous geometry and parametric reflectance under non-rigid motion with known lighting positions. Unlike this previous motion stereo work, spacetime stereo does *not* estimate inter-frame motion, but rather linearizes local temporal disparity variation. The local temporal linearization is generally valid for continuous visible surfaces so long as the cameras have a sufficiently high frame rate with respect to the 3D motion. A key advantage of our approach is that it does not require brightness constancy over time. Caspi and Irani [5] use a related idea to align image sequences assuming a global affine transformation between sequences. Our approach can be viewed as extending this method to compute a full stereo correspondence instead of an affine transformation.

One important application where brightness constancy is especially violated is structured light scanning, where controlled lighting is used to induce temporal appearance variation and used to reconstruct accurate shape models. We refer the readers to the related work sections in [8, 19] for a more complete review and summarize the most relevant research here. Kanade *et al.* [9] and Curless and Levoy [6] used temporal intensity variation (the latter called it "spacetime analysis") to resolve correspondence between sweeping laser stripes and sensor pixels. Pulli *et al* [13] adapted spacetime analysis to match a sweeping projector stripe observed by multiple video cameras. Bouguet and Perona [3] applied spacetime analysis to shadow profiles simultaneously cast onto a scene and a calibrated plane as observed by a single video camera. These single-stripe spacetime analysis methods are limited to static scenes and require hundreds of images for reconstructing an object.

In the direction of using fewer images, Sato and Inokuchi [14] describe a set of hierarchical stripe patterns to give range images with $\log N$ images, where N is the number of resolvable stripes. In particular, each camera pixel observes a bit code over time that uniquely determined its correspondence. The method is limited to static scenes. Hall-Holt and Rusinkiewicz [8] describe a method that consists of projected *boundary-coded* stripe patterns that vary over time. By finding nearest stripe patterns over time, a unique code can be determined for any stripe at any time. The constraint in this case is that the object move slowly to avoid erroneous temporal correlations, and only depths at stripe boundaries are measured. Note that each of these methods is specially designed for a certain class of patterns and is not applicable under more general illumination changes.

Zhang *et al.* [19] proposed an approach in which a pattern consisting of multiple color stripes is swept through the scene and multi-pass dynamic programming is used to match the video sequence and the pattern sequences. The work can be viewed as extending spacetime analysis to simultaneously resolve multiple stripes at the same time. Their spacetime results show clear improvements in terms of matching accuracy over pure frame by frame spatial matching. However, the method is limited to static objects. In this paper, we extend spacetime analysis to handle moving scenes.

Finally, in these same proceedings, Davis et al. [7] develop a similar spacetime stereo framework as the one presented here. However, their work is focused on analyzing and presenting results for geometrically static scenes imaged under varying illumination. In this paper, we develop ideas that handle illumination variation, as well as geometrically "quasi-static" and moving scenes.

3. Spacetime stereo metrics

In this section, we formulate the spacetime stereo problem, and define the metrics that are used to compute correspondences.

Consider a Lambertian scene observed by two synchronized and pre-calibrated video cameras. Spacetime stereo takes as input two rectified image streams $I_l(x, y, t)$ and $I_r(x, y, t)$. To recover the time-varying 3D structure of the scene, we wish to estimate the disparity function $d(x, y, t)$ for each pixel (x, y) at each time t. Most existing stereo algorithms solve for $d(x, y, t)$ at some position and moment (x_0, y_0, t_0) by minimizing the following error function

$$E(d_0) = \sum_{(x,y) \in W_0} e(I_l(x, y, t_0), I_r(x - d_0, y, t_0)) \tag{1}$$

where d_0 is shorthand notation for $d(x_0, y_0, t_0)$, W_0 is a spatial neighborhood window around (x_0, y_0), $e(p, q)$ is a similarity measure between pixels from two cameras, and we are

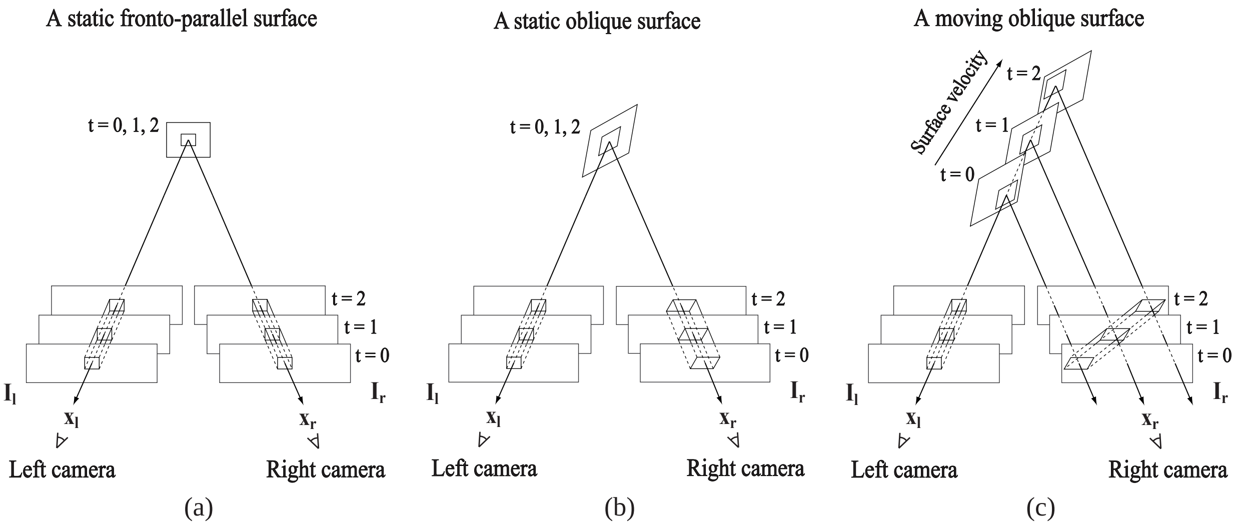


Figure 1. Illustration of spacetime stereo. Two stereo image streams are captured from stationary sensors. The images are shown spatially offset at three different times, for illustration purposes. For a static or quasi-static surface (a,b), the spacetime windows are "straight", aligned along the line of sight. For an oblique surface (b), the spacetime window is horizontally stretched and vertically sheared. For a moving surface (c), the spacetime window is also temporally sheared, i.e., "slanted". The best affine warp of each spacetime window along epipolar lines is computed for stereo correspondence.

measuring disparity in the right image with respect to the left image. Depending on the specific algorithm, the size of W_0 can vary from being a single pixel to, say, a 10x10 neighborhood. $e(a, b)$ can simply be

$$e(p, q) = (p - q)^2 \tag{2}$$

in which case Eq. (1) becomes the standard sum of squared difference (SSD). We have obtained better results in practice by defining $e(a, b)$ to compensate for radiometric differences between the cameras:

$$e(p, q) = (s{\cdot}p + o - q)^2 \tag{3}$$

where s and o are window dependent scale and offset constants to be estimated. Other forms of $e(a, b)$ are summarized in [1]. Note that Eq. (3) is similar enough to a squared difference metric that, after substituting into Eq. (1), we still refer to the result as an SSD formulation.

We seek to incorporate *temporal appearance variation* to improve stereo matching and generate more accurate and reliable depth maps. In the next two subsections, we'll consider how multiple frames can help to recover static and nearly static shapes, and then extend the idea for moving scenes.

3.1. Static scenes

Scenes that are geometrically static may still give rise to images that change over time. For example, the motion of the sun causes shading variations over the course of a day. In a laboratory setting, projected light patterns can create similar but more controlled changes in appearance.

Suppose that the scene is static for a period of time $T_0 = [t_0 - \Delta t, t_0 + \Delta t]$. As illustrated in Figure 1(a), we can extend the spatial window to a spatiotemporal window and solve for d_0 by minimizing the following sum of SSD (SSSD) cost function:

$$E(d_0) = \sum_{t \in T_0} \sum_{(x,y) \in W_0} e(I_l(x, y, t), I_r(x - d_0, y, t)) \tag{4}$$

This error function reduces matching ambiguity in any single frame by simultaneously matching intensities in multiple frames. Another advantage of the spacetime window is that the spatial window can be shrunk and the temporal window can be enlarged to increase matching accuracy. This principle was originally formulated as spacetime analysis in [6, 9] for laser scanning and was applied by several researchers [3, 13, 19] for structured light scanning. Here we are casting it in a general spacetime stereo framework.

We should point out that both Eq. (1) and Eq. (4) treat disparity as being constant within the window W_0, which assumes the corresponding surface is fronto-parallel. For a static but oblique surface, as shown in Figure 1(b), a more accurate (first order) local approximation of the disparity function is

$$d(x, y, t) \approx \hat{d}_0(x, y, t) \overset{\text{def}}{=} d_0 + d_{x_0} \cdot (x - x_0) + d_{y_0} \cdot (y - y_0) \tag{5}$$

where d_{x_0} and d_{y_0} are the partial derivatives of the disparity function with respect to spatial coordinates x and y at

(x_0, y_0, t_0). This local spatial linearization results in the following SSSD cost function to be minimized:

$$E(d_0, d_{x_0}, d_{y_0}) = \sum_{t \in T_0} \sum_{(x,y) \in W_0} e(I_l(x,y,t), I_r(x - \hat{d}_0, y, t)) \tag{6}$$

where $\hat{d}_0$ is a shorthand notation for $\hat{d}_0(x, y, t)$, which is defined in Eq. (5) in terms of (d_0, d_{x_0}, d_{y_0}) and is estimated for each pixel. Non-zero d_{x_0} and d_{y_0} will cause a horizontal stretch or shrink and vertical shear of the spacetime window respectively, as illustrated in Figure 1(b).

3.2 Quasi-static scenes

The simple SSSD method proposed above can also be applied to an interesting class of time-varying scenes. Some natural scenes, like water flow in Figure 5, have spatially varying texture and motion, but an overall shape that is roughly constant over time. Although these natural scenes move stochastically, people tend to fuse the image stream into a gross average shape over time. In this paper, this class of natural scenes is refered to as *quasi-static*. By applying the SSSD method from the previous section, we can compute a temporally averaged disparity map which corresponds roughly to the "mean shape" of the scene. In graphics applications where a coarse geometry is desired, one could, for instance, use the mean shape as static geometry with time-varying color texture mapped over the surface.

3.3. Moving scenes

Next, let's consider the case where the object is moving in the time interval $T_0 = [t_0 - \Delta t, t_0 + \Delta t]$, as illustrated in Figure 1(c). Because of the object motion, the window in the left video is deformed in the right sequence. The temporal trajectory of window deformation in the right video is determined by the object motion and could be arbitrarily complex. However, if the camera has a high enough frame rate relative to the object motion and there are no changes in visibility, we can also locally linearize the temporal disparity variation in much the same way we linearized spatial disparity in Eq. (5). Specifically, we take a first order approximation of disparity variation with respect to both spatial coordinate x and y and temporal coordinate t as

$$\begin{gathered} d(x,y,t) \approx \tilde{d}_0(x,y,t) \overset{\text{def}}{=} \\ d_0 + d_{x_0} \cdot (x - x_0) + d_{y_0} \cdot (y - y_0) + d_{t_0} \cdot (t - t_0) \end{gathered} \tag{7}$$

where d_{t_0} is the partial derivative of disparity function with respect to time at (x_0, y_0, t_0). This local spatial-temporal linearization results in the following SSSD cost function to be minimized:

$$\begin{gathered} E(d_0, d_{x_0}, d_{y_0}, d_{t_0}) = \\ \sum_{t \in T_0} \sum_{(x,y) \in W_0} e(I_l(x,y,t), I_r(x - \tilde{d}_0, y, t)) \end{gathered} \tag{8}$$

where $\tilde{d}_0$ is a shorthand notation for $\tilde{d}_0(x, y, t)$, which is defined in Eq. (7) in terms of $(d_0, d_{x_0}, d_{y_0}, d_{t_0})$and is estimated for each pixel at each time. Note that Eq. (7) assumes a linear model of disparity within the spacetime window, i.e., $(d_0, d_{x_0}, d_{y_0}, d_{t_0})$ is constant within $W_0 \times T_0$.

We use the term *straight window* to refer to a spacetime window whose position and shape is fixed over time, such as the windows shown in Figure 1(a,b). If the position of the window varies over time, we say that the spacetime window is *slanted*, such as the one in the right camera in Figure 1(c). In its current formulation, spacetime stereo requires matching a straight spacetime window around each pixel (x, y) at each time t in the left image stream to, in general, a slanted spacetime window in the right image stream. It would be straightforward, however, to make the metric more symmetric by adding another SSSD term similar to the one in Eq.(8) except that it would use a straight window in the right image and a slanted one in the left.

4. Spacetime stereo matching

A wide variety of existing stereo algorithms are easily adapted for spacetime stereo. Most stereo algorithms have a "data term" $C(x_l, x_r; y, t)$ that describes the similarity of the pixel or region around (x_l, y, t) in the left image and (x_r, y, t) in the right. To use these algorithms for spacetime stereo, one can simply replace the data term as

$$C(x_l, x_r; y, t) = \min_{d_x, d_y, d_t} E(x_l - x_r, d_x, d_y, d_t) \tag{9}$$

Eq. (9) can be incorporated as the cost function in most existing stereo algorithms, e.g., window-based correlation, dynamic programming, graph cuts, and so forth (see [15] for a description and evaluation of these methods). For our experiments, we used dynamic programming to compute a pixel-accurate correspondence followed by Lucas-Kanade flow to obtain sub-pixel disparities.

The minimization of Eq. (9) is constrained by $d_t = 0$ for geometrically static and quasi-static scenes and by $d_x = d_y = 0$ if local spatial disparity variation is ignored due to insufficient spatial intensity contrast.

5. Results

We have performed several experiments to evaluate the performance of spacetime analysis for static, quasi-static, and moving scenes. In each case, we performed camera calibration using Bouguet's calibration toolbox [2].

5.1. Static Scenes

To test the performance with static scenes, we applied varying illumination to induce appearance changes. In the first experiment, we employed a stereo pair of synchronized Basler A301f monochrome video cameras and projected light onto a scene with a 60 Hz Compaq MP1800 digital projector. Spacetime stereo places no restriction on the type of

(a) Gray code

	stripe intensities															
t=0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
t=1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
t=2	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
t=3	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0

(b) Modified Gray code

	stripe intensities															
t=0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
t=1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
t=2	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
t=3	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1

Figure 2. Structured light illumination patterns. Each pattern constists of a set of black and white vertical stripes, defined by the sequence of zeros and ones shown above. At each time instant, a different pattern is displayed. (a) A Gray code pattern for 16 stripes. Each column is unique, but some rows contain only low spatial frequencies. (b) After shuffling columns of the Gray code, we obtain a modified pattern that still has unique columns, but also has high spatial frequencies in each row.

projected pattern. For instance, a hierarchical stripe (Gray code) pattern [14], as shown in Figure 2(a) should yield good depth maps. In practice, we obtained more accurate results by projecting patterns with high spatial frequencies in each frame. Intuitively, this happens because the coarse patterns, while providing some disambiguation cues for the finer patterns, give little detailed information for disparity matching. Our approach then is to take the Gray code and shuffle the temporal columns so that each is still unique, but each resulting horizontal pattern has high frequency spatial detail (Figure 2(b)).

Our final pattern sequence is comprised of 8 patterns, each of which has 256 4-pixel-wide stripes. Each pattern was low-pass filtered to obtain patterns with continuous intensity variation, resulting in more accurate per-pixel matching.

Figure 3 shows the results obtained by imaging a small sculpture (a plaster bust of Albert Einstein) and using 10 images taken from each camera matched with a spacetime window of 5x5x10 (5x5 pixels per frame by 10 frames). The shaded rendering reveals details comparable to those obtained with a laser range scanner. Eq. (3) is used as pixel similarity measure in the Lucas-Kanade refinement for this example (also in the moving hand example later).

Next, we tried a much simpler imaging system for static shape capture using much looser structured light. For the stereo pair, we attached a Pentax stereo adapter to a single SONY-TRV900 camcorder. For illumination, we shined an ordinary desk lamp through a transparency printed with a black and white square pattern onto the subject (a teddy bear) and moved the pattern by hand in a free-form fashion. In this case, we captured 125 frames and tried both single frame stereo for one of the image pairs using a 5x1 window and spacetime stereo over all frames using a 5x1x125 window. In both cases (also in the waterfall example later), spatial disparity variation is ignored within the windows, i.e. $d_x = d_y = 0$, and Eq. (2) is used as the pixel similarity measure. Figure 4 shows marked improvement of spacetime stereo over regular stereo, in addition to improvement due to the final Lucas-Kanade subpixel refinement step.

5.2. Quasi-static objects

For a quasi-static scene, we took a sequence of 45 images of a small but fast-moving waterfall using a Nikon Coolpix 900 still camera with the stereo adapter attached. Figure 5 shows a comparison of the results obtained with traditional stereo for one of the image pairs, followed by results obtained with the same spacetime stereo reconstruction technique used for the teddy bear example. Note how much more consistent and complete the spacetime result is.

5.3. Moving Scenes

For moving scenes, we tried two experiments. In the first, we projected the modified Gray code pattern onto a human hand that is moving fairly steadily away from the stereo rig. In this case, the disparity is changing over time, and the straight spacetime window approach fails to reconstruct a reasonable surface. By estimating the temporal derivative of the disparity using slanted windows, we obtain a much better reconstruction, as shown in Figure 6.

We also tried imaging moving and deforming textured objects under more natural lighting conditions. In practice, we found that spacetime stereo performed approximately as well as regular stereo with larger windows. In the next section, we offer an explanation of why we did not observe substantial improvement with spacetime stereo in this case.

6. Discussion

In this paper, we have described a simple extension to traditional stereo that incorporates temporal information into the stereo pair disparity estimation problem. Here we discuss the “operating range” of the technique and suggest ideas for future work.

For geometrically static scenes, the spacetime stereo framework has proved effective for reconstructing shape from changes in appearance. We’ve demonstrated results for tightly controlled illumination changes, as well as more loosely controlled variations, both in a laboratory setting. The method thus represents a generic way to explore structured lighting methods with the advantage over the standard methods being that the illumination need not be calibrated and interreflections on the surface are not problematic as long as the scene is diffuse. Further, the approach is general enough to handle more natural appearance variations including shadows sweeping across architecture and objects that change color naturally over time (e.g., metallic patinas, food that browns with age, ants crawling over surfaces, etc.). The

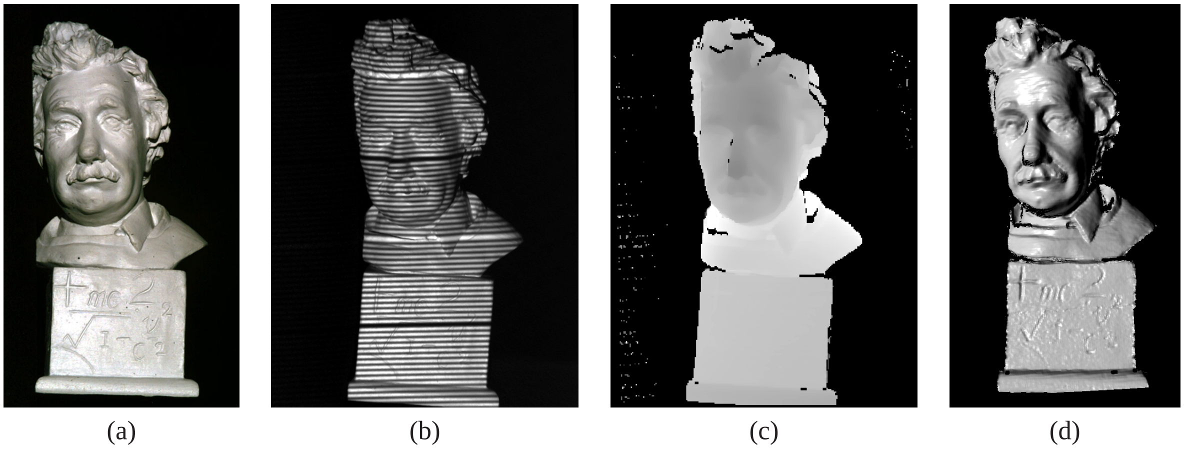


Figure 3. Structured lighting result with modified Gray code. (a) Einstein bust under natural lighting. (b) One image taken from the set of 10 stereo pair images. The images were taken with the bust laying on its side, but the one shown here has been rotated 90 degrees. (c) Reconstructed disparity map. (d) Shaded rendering of geometric reconstruction.

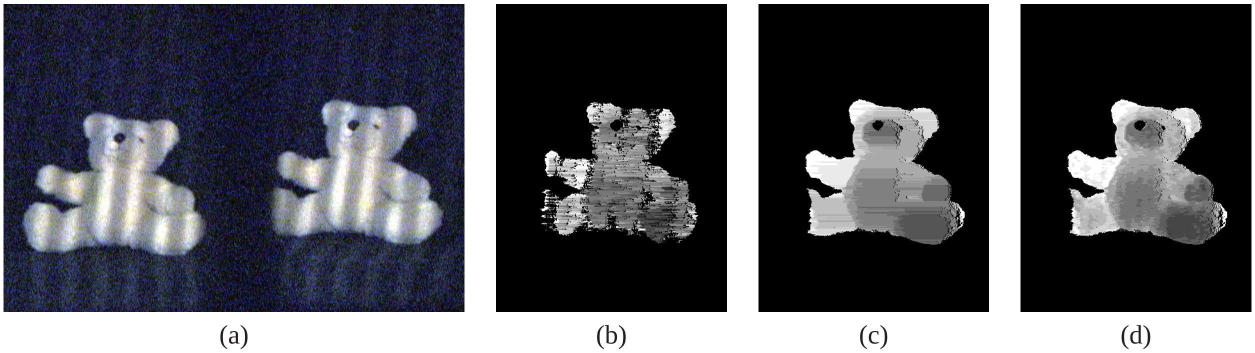


Figure 4. Loosely structured lighting using transparency and desk lamp. (a) One out of the 125 stereo pair images. (b) Disparity map for a traditional stereo reconstruction. (c) Disparity map for spacetime stereo after DP. (d) Same as (c) after Lucas-Kanade refinement.

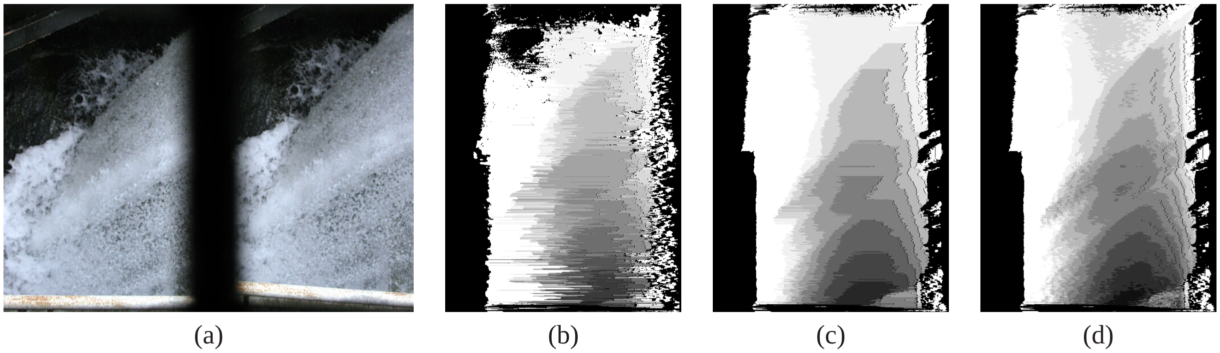
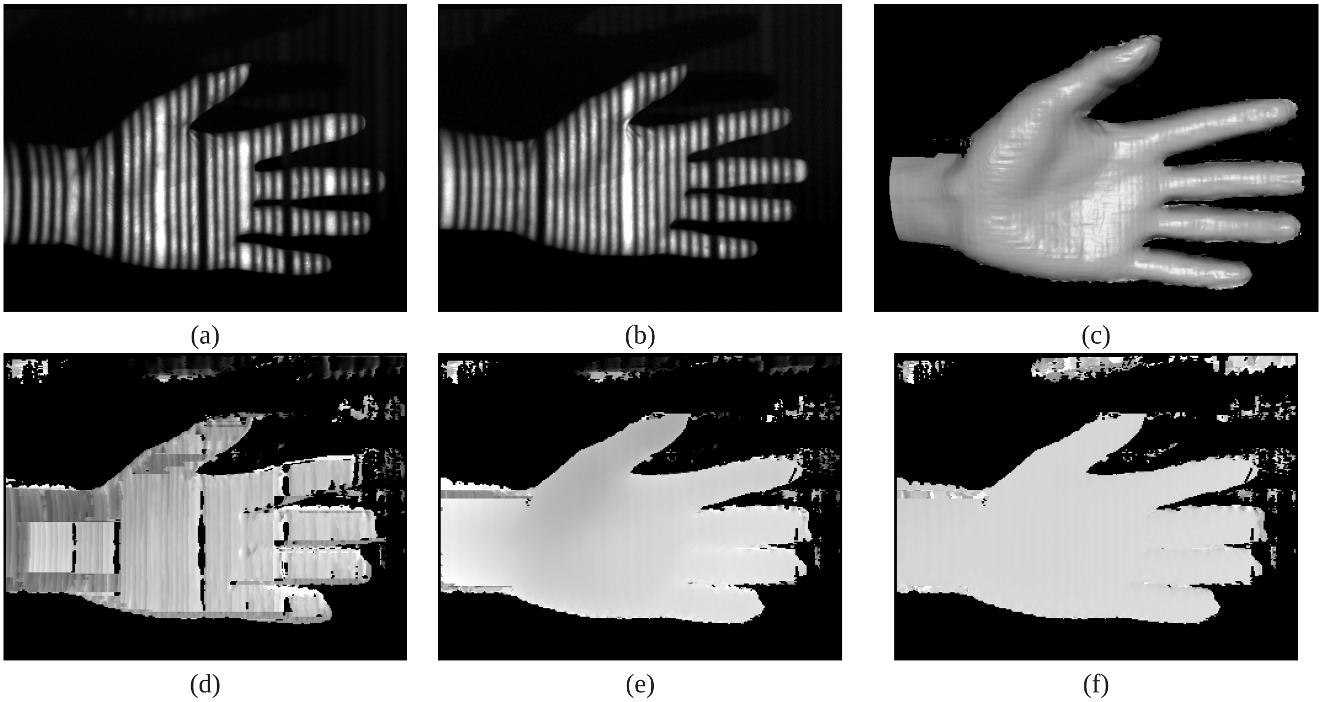


Figure 5. Quasi-static rushing water experiment. (a) One out of the 45 stereo pair images. (b) Disparity map for a traditional stereo reconstruction. (c) Disparity map for spacetime stereo after DP. (d) Same as (c) after Lucas-Kanade refinement.



(a) (b) (c)

(d) (e) (f)

Figure 6. Moving hand under structured light. (a), (b) Two images taken from one of the cameras. The hand is moving away from the stereo rig, which is why it is getting smaller. (c) Shaded rendering of the geometrically reconstructed model using slanted window spacetime analysis. (d) Disparity map with straight spacetime windows. (e) Disparity map with slanted spacetime windows. (f) Temporal derivative of disparity. Since the hand is translating at roughly a constant speed, the disparity velocity is fairly constant over the hand.

approach should also demonstrate some small amount of improvement over per-frame stereo for image streams of geometrically static scenes with no temporal variations, because the influence of noise should be averaged out over time.

For quasi-static scenes, we have also demonstrated improvement over per-frame stereo. Instead of using our spacetime approach, one could in principle compute the disparities per-frame and then average them together, though we would expect a number of outliers and non-matches to complicate this process, and it would actually be slower to execute because of the need to perform stereo matching n times for n frames instead of only once. An area for future work would be to try to model the statistical variation of quasi-static and moving scenes, e.g., to model the stochastic changes in disparity for a waterfall.

For dynamic scenes, our most compelling results have been for structured light systems. In this setting, the fact that we do not perform motion stereo analysis (i.e., 2D optical flow in tandem with stereo) is essential, since the brightness constancy constraint does not apply. Our results indicate that dense structured light stereo is possible even as the subject moves.

For dynamic scenes under more natural (smoother) and constant illumination, we have observed less benefit with the spacetime stereo method. Let's consider in particular a

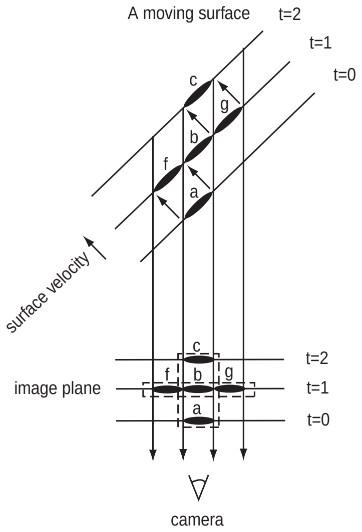


Figure 7. Trade-off between spatially narrow windows for spacetime stereo and wide windows for traditional stereo.

scene with constant ambient lighting and with Lambertian reflectance at each surface point. In this case, the spacetime window appears to be a trade-off between wider spatial window and a narrower (but temporally longer) spacetime window, as illustrated in Figure 7. The spacetime window $\{a, b, c\}$ over time $t = 0, 1, 2$ contains the same information as the spatial window $\{f, b, g\}$ at time $t = 1$ because the window f at $t = 1$ is the same as window a at $t = 0$ and window c at $t = 2$ is the same as window g at $t = 1$. Therefore, using a spacetime window is not always more powerful than a purely spatial window. An existing frame by frame stereo matching algorithm with a spatial window size $\{f, b, g\}$ is expected to have the same performance as spacetime stereo with spacetime window $\{a, b, c\}$. An area for future work is to formalize this reasoning and extend the adaptive window method [10] to search for optimal spacetime windows. Finally, another interesting direction would be to adapt global optimization methods like graph cuts [11] to compute depth maps that are piece-wise smooth both in space and time.

Acknowledgments

This work was supported in part by National Science Foundation grants CCR-0098005 and IIS-0049095, an Office of Naval Research YIP award, the Animation Research Labs, and Microsoft Corporation. The authors would like to thank P. Anandan and R. Szeliski for their insightful comments and N. Li for her help in data collection.

References

[1] S. Baker, R. Gross, I. Matthews, and T. Ishikawa. Lucas-kanade 20 years on: A unifying framework: Part 2. Technical Report CMU-RI-TR-03-01, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, February 2003.

[2] J.-Y. Bouguet. *Camera Calibration Toolbox for Matlab*. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html, 2001.

[3] J.-Y. Bouguet and P. Perona. 3D photography on your desk. In *Int. Conf. on Computer Vision*, 1998.

[4] R. L. Carceroni and K. N. Kutulakos. Scene capture by surfel sampling: from multi-view streams to non-rigid 3D motion, shape and reflectance. In *Int. Conf. on Computer Vision*, 2001.

[5] Y. Caspi and M. Irani. A step towards sequence to sequence alignment. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2000.

[6] B. Curless and M. Levoy. Better optical triangulation through spacetime analysis. In *Int. Conf. on Computer Vision*, pages 987–994, June 1995.

[7] J. Davis, R. Ramamoorthi, and S. Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2003.

[8] O. Hall-Holt and S. Rusinkiewicz. Stripe boundary codes for real-time structured-light range scanning of moving objects. In *Int. Conf. on Computer Vision*, pages 359–366, 2001.

[9] T. Kanade, A. Gruss, and L. Carley. A very fast vlsi rangefinder. In *Int. Conf. on Robotics and Automation*, volume 39, pages 1322–1329, April 1991.

[10] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(9), 1994.

[11] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *Eur. Conf. on Computer Vision*, 2002.

[12] R. Mandelbaum, G. Salgian, and H. Sawhney. Correlation based estimatin of ego-motion and structure from motion and stereo. In *Int. Conf. on Computer Vision*, pages 544–550, 1999.

[13] K. Pulli, H. Abi-Rached, T. Duchamp, L. Shapiro, and W. Stuetzle. Acquisition and visualization of colored 3D objects. In *Int. Conf. on Pattern Recognition*, 1998.

[14] K. Sato and S. Inokuchi. Range-imaging system utilizing nematic liquid crystal mask. In *Int. Conf. on Computer Vision*, pages 657–661, 1987.

[15] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. on Computer Vision*, 47(1):7–42, 2002.

[16] C. Strecha and L. V. Gool. Motion - stereo integration for depth estimation. In *Eur. Conf. on Computer Vision*, 2002.

[17] H. Tao, H. S. Sawhney, and R. Kumar. Dynamic depth recovery from multiple synchronized video streams. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.

[18] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. In *Int. Conf. on Computer Vision*, 1999.

[19] L. Zhang, B. Curless, and S. M. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *Int. Symp. on 3D Data Processing, Visualization, and Transmission*, 2002.

[20] Y. Zhang and C. Kambhamettu. Integrated 3D scene flow and structure recovery from multiview image sequences. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2000.

[21] Y. Zhang and C. Kambhamettu. On 3D scene flow and structure estimation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.