

Consistent Stereo-Assisted Absolute Phase Unwrapping Methods for Structured Light Systems

Ricardo R. Garcia, *Student Member, IEEE*, and Avideh Zakhor, *Fellow, IEEE*

Abstract—Phase shifted sinusoidal patterns have proven to be effective in structured light systems, which typically consist of a camera and projector. They offer low decoding complexity, require as few as three projection frames per reconstruction, and are well suited for capturing dynamic scenes. In these systems, depth is reconstructed by determining the phase projected onto each pixel in the camera and establishing correspondences between camera and projector pixels. Typically, multiple periods are projected within the set of sinusoidal patterns, thus requiring phase unwrapping on the phase image before correspondences can be established. A second camera can be added to the structured light system to help with phase unwrapping. In this work, we present two consistent phase unwrapping methods for two-camera stereo structured light systems. The first method enforces viewpoint consistency by phase unwrapping in the projector domain. Loopy belief propagation is run over the graph of projector pixels to select pixel correspondences between the left and right camera that align in 3-D space and are spatially smooth in each 2-D image. The second method enforces temporal consistency by unwrapping across space and time. We combine a quality guided phase unwrapping approach with absolute phase estimates from the stereo cameras to solve for the absolute phase of connected regions. We present results for both methods to show their effectiveness on real world scenes.

Index Terms—Depth reconstruction, dynamic scene capture, phase unwrapping, stereo, structured light (SL).

I. INTRODUCTION

A significant amount of work has been focused on systems that create accurate 3-D models of real world scenes; these include laser scanners, stereo-camera systems, and structured light systems [1]. Some of the methods are not well suited for the task of capturing 3-D models of dynamic scenes. Laser scanners, for example, have proven to be expensive and do not have the spatial or temporal resolution to capture the 3-D depth of a dynamic scene. Stereo-camera based approaches generate 3-D models of a scene by stereo matching across pairs, or multiple pairs, of cameras [2].

An alternative method for 3-D reconstruction of scenes is structured light (SL). SL systems replace one of the cameras in the traditional two-camera stereo method with a projector which projects patterns that uniquely identify points in the scene. The remaining camera identifies the pattern present at each observed

pixel from which correspondences between the camera and projector pixels are established. From these correspondences, a 3-D model is generated through triangulation [3].

There has been extensive research in designing effective SL illumination patterns that uniquely identify points within the illuminated scene [4]–[6]. Examples of pattern generation methods include temporal and spatial coding of projection patterns or “viewpoint-coded structured light” [7]–[9]. For capturing a dynamic scene using structured light, it is desirable to use illumination patterns that maximize the quality and resolution of each reconstruction while simultaneously minimizing the number of projected frames in the pattern, in order to maximize the temporal update rate of the scene [6].

An important class of projection patterns in SL systems is phase-shifted sinusoids. These patterns are robust to depth of field effects and are simple to decode. In the common setup, three sinusoidal images, each phase shifted by $2\pi/3$, are sequentially projected onto the scene. The three observed intensities of a given pixel during the pattern projection are used to determine the wrapped phase of the corresponding pixel in the projector. Performing this calculation on all points in the camera image allows a phase image to be generated from which the scene can be reconstructed [3], [10], [11]. In these systems, it is common to project multiple periods of the sinusoids across the screen. This reduces the number of unique phase values that must be identified, thereby making the decoding process less susceptible to noise. At the same time, for an M period projection there is an M -fold ambiguity in finding the corresponding point in the projector’s image because only the wrapped phase of each pixel can be determined. This ambiguity is removed through the image phase unwrapping process [12].

In this work, we present two phase unwrapping methods in a stereo structured light system to ensure viewpoint or temporal consistency. The first method ensures viewpoint consistency when unwrapping images in the stereo SL system [13], and the second method ensures temporal consistency while unwrapping each camera over time [14].

In our first method, the phase images for both the left and right cameras in the stereo SL system are simultaneously unwrapped. Rather than unwrapping over the phase image of each camera individually, we unwrap by processing the pixels in the projector. There are several potential advantages to such a projector domain phase unwrapping approach. First, operating on the projector pixels allows us to assign an absolute phase to corresponding pairs of pixels in the cameras. This ensures that the phase images are unwrapped consistently between camera views, and that the triangulated points from each camera are in agreement. Second, in a projector centric approach, the phase

Manuscript received August 01, 2011; revised January 13, 2012; accepted March 25, 2012. Date of publication April 18, 2012; date of current version August 10, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Stefano Mattoccia.

The authors are with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, CA 94720 USA (e-mail: rrgarcia@eecs.berkeley.edu; avz@eecs.berkeley.edu).

Digital Object Identifier 10.1109/JSTSP.2012.2195157

unwrapping algorithm only needs to be run once regardless of the number of cameras in the system. As such, it is likely to 1) be more computationally efficient, and 2) result in more consistent solutions with fewer discrepancies across cameras. Third, the computational complexity for a projector based solution is proportional to the number of projector pixels rather than the number of camera pixels, as is true in traditional camera centric reconstruction. Since from a technological point of view, camera resolution is likely to increase at a much faster rate than projector resolution, projector centric solutions are likely to be more computationally efficient.

In our second method, we unwrap temporally across multiple consecutive phase images from a single camera. Most existing SL systems for dynamic scenes process the captured data one frame at a time with no guarantee of temporal consistency in the unwrapped images. This could result in temporal discontinuities and artifacts when viewing the reconstructed point cloud over time. It is conceivable to both remove these temporal artifacts and improve phase unwrapping accuracy by unwrapping across time as well as space [15]. Thus, we propose merging stereo phase unwrapping with 3-D phase unwrapping to arrive at temporally consistent absolute phase volumes. By 3-D, we refer to (X, Y, T) rather than the commonly used (X, Y, Z) , where T represents time.

The outline of this paper is as follows. Section II provides background for our phase unwrapping methods. Section III reviews the basic stereo phase unwrapping process. In Section IV, we present our proposed viewpoint consistent phase unwrapping method. Section V provides an overview of the 3-D phase unwrapping method [15], and Section VI presents our proposed temporally consistent phase unwrapping method. In Section VII, we describe the experimental setup along with results for both methods. Section VIII provides a comparison of our methods to related works, and Section IX is conclusions.

II. BACKGROUND

Two-dimensional phase unwrapping has been extensively studied in the signal processing literature [16]–[19]. In particular, the problem of phase unwrapping in sinusoidal projection systems has been extensively explored [12], [17], [20]–[22]. The approaches to phase unwrapping in sinusoidal systems can be classified into three categories: temporal, period coded, and spatial phase unwrapping methods. Temporal unwrapping methods project additional patterns to remove the absolute phase ambiguity from the phase shifted sinusoidal patterns [23], [24]. In [23], gray coded patterns are projected in addition to the phase shifted sinusoidal patterns. The gray code patterns encode the period of each projected pixel so that the absolute phase of each pixel can be directly determined. In another approach by Huntley and Saldner, the frequency of the projected sinusoidal pattern is increased as each pattern is projected [24]. The absolute phase of each pixel is then determined by counting the number of 2π period crossings in each pixel's temporal sequence. In their experiments, 17 patterns are projected for a single reconstruction. Temporal methods allow for the absolute phase of each pixel to be determined at the cost of significantly increasing the number of projected patterns. Unless very high-speed cameras are used, temporal-based

projection methods are not suitable for dynamic scene capture. Period coded methods embed the period information within the same patterns as the phase shifted sinusoids. Liu *et al.* add a single period sinusoid to the high-frequency sinusoidal pattern [25]. By projecting five phase shifted versions of the summed sinusoidal patterns, the phase of both the high- and low-frequency sinusoid can be determined at each point. The two phase estimates can then be used to generate the absolute phase of each pixel. Once again, this comes at the expense of using additional patterns. The final approach to phase unwrapping is spatial unwrapping. In spatial unwrapping, the unwrapped phase of each pixel is determined by locally summing phase values along paths within the 2-D image. Unlike temporal and phase coded methods, spatial unwrapping methods do not always provide absolute phase results. It is often the case that the unwrapped results have an additional unknown $2\pi k$ offset, where k is an integer value. It is necessary to determine the absolute phase in order to calculate the true depth of an object.

To generate accurate unwrapped results, a number of assumptions are made about the content of the scene during the spatial phase unwrapping process in sinusoidal SL systems [20], [21]. First, the scene is assumed to contain a single continuous object in space, i.e., all objects in the scene are connected together. Second, phase unwrapping via traditional spatial methods requires most neighboring points to have phase differences that are smaller than $|\pi|$. Without this assumption, the true phase differences cannot easily be determined because of the periodic nature of the phase values. This typically restricts the types of scenes that can successfully be reconstructed. For instance, scenes with multiple disconnected objects or large depth discontinuities cannot be unwrapped with spatial phase unwrapping techniques alone.

As mentioned, spatial phase unwrapping allows for the recovery of the relative phase between pixels, but it does not directly anchor the phase of each pixel to an absolute value. Without absolute phase, there remains a constant unknown offset in the correspondence between camera pixels and projector columns. A variety of techniques have been developed to determine this constant in order to recover absolute depth via triangulation. In [11], Zhang and Yau embed a small cross in the projected pattern to serve as a reference position with a known absolute phase. Once detected, all the remaining points are unwrapped with respect to the marker in order to determine the absolute phase of the entire image. The embedded marker serves as a valid reference point, but it can be difficult to determine the position of the marker if the projected pattern is out of focus or if there is significant texture within the scene.

While traditional SL systems consist of a single camera and projector, several SL systems with multiple cameras recently have been proposed [11], [26], [27]. In addition to using a marker to solve for absolute phase, Zhang and Yau also use two cameras in order to increase the reconstruction coverage of a scene [11]. With the absolute phase recovered, the reconstructions from the two camera views can be accurately merged. The work by Han and Huang uses a pair of stereo cameras with a sinusoidal SL system to achieve accurate stereo matches between the cameras [26]. They search for correspondences between the cameras by using absolute unwrapped phase images. By not

triangulating depths with respect to the projector, their method does not require projector calibration and is robust to distortions of the sinusoidal patterns caused by the projector. Their method only works on scenes where there are no large depth discontinuities, since spatial unwrapping is required. A second camera can also be added to the single-camera, single-projector SL system [27] to overcome the phase discontinuity problem and allow for the reconstruction of scenes with multiple objects. In doing so, the absolute phase for many points in the scene can be determined. Specifically, the second camera resolves the absolute phase ambiguity as follows [27]: for each pixel in the first camera, the wrapped phase is calculated, and the possible corresponding locations in the projector are determined. Given M periods in the projected sinusoidal patterns, there are M columns in the projector that could possibly correspond to the identified camera pixel. These M points are triangulated in 3-D space, and each 3-D point is projected onto the image plane of the second camera. Each projected pixel's phase is then compared to the phase of the pixel in the first camera. The projected pixel with the closest phase to the reference pixel is identified as the correspondence. This approach can result in occasional absolute phase errors due to occlusions, calibration errors, "false" matches, and scene motion. In [27], the errors are cleaned up through an energy minimization framework that uses both data and smoothing cost components.

In a stereo SL system capturing multiple poses of a dynamic scene, it is important for the unwrapped images of each camera at each time to be consistent. We refer to "viewpoint consistency" as unwrapping left and right camera phase images such that the corresponding scene points in each phase image have the same absolute phase. Similarly, we refer to "temporal consistency" as having consecutively unwrapped phase images with similar phase values for all corresponding points between consecutive frames.

Three-dimensional phase unwrapping methods have been explored in the literature [15], [22], [28]–[46]. Specifically, Su and Zhang examine the 3-D unwrapping problem for a structured light application [22]. Despite generating phase images through Fourier transformation profilometry rather than through shifted sinusoids, they present several methods for unwrapping a 3-D phase volume that is generated by stacking consecutive phase images. In Huntley's work, a 3-D volume is unwrapped by removing edges between pixels that likely have a 2π discontinuity [28]. This method is capable of dealing with 2π discontinuities from noise, but cannot deal with true phase discontinuities.

Our proposed method is a variation of the 3-D phase unwrapping algorithm proposed in [15]. Originally developed for MRI applications, the method assigns a quality measure to each pixel in the wrapped phase volume based on its local spatio-temporal second difference values. The edges between neighboring spatio-temporal pixels are also assigned quality measures based on the quality of the connected pixels. The phase volume is then unwrapped based on rank ordered edge qualities starting from the highest to lowest [15]. Since in our setup we assume a stereo camera pair rather than a single camera, we develop a framework to integrate stereo phase unwrapping with the basic 3-D phase unwrapping method in [15] in order to determine absolute phase. Besides providing temporal consistency, this approach is

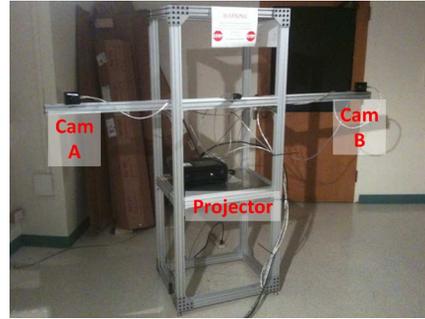


Fig. 1. Configuration of stereo SL system.

also effective in unwrapping scenes with large depth discontinuities or multiple spatially disjoint objects.

III. OVERVIEW OF STEREO PHASE UNWRAPPING

In stereo phase unwrapping, a second camera is added to the traditional SL system made of a projector and a single camera. To maximize the camera coverage of the illuminated scene, the projector is positioned in between the two cameras, as shown in Fig. 1. Before processing captured data, the projector and the pair of cameras must all be calibrated [47].

During capture, the scene is illuminated by three consecutive phase shifted sinusoidal images, as in [3]. After each camera captures the scene from its view, a wrapped phase image is computed as

$$\phi(x, y) = \tan^{-1} \left(\frac{\sqrt{3}(I_1(x, y) - I_3(x, y))}{2I_2(x, y) - I_1(x, y) - I_3(x, y)} \right) \quad (1)$$

where $I_i(x, y)$ represents the intensity of image i at image coordinates (x, y) , and $\phi(x, y)$ represents the wrapped phase at (x, y) . If a scene point is visible in both cameras, the phase measurement is independent of the viewing direction, except for saturated image regions corresponding to specular surfaces. Computing phase via (1) requires the projector to accurately project true sinusoidal patterns. Since the color processing algorithms of the projector modify the intended projection patterns, we pre-distort the sinusoidal patterns so that the outgoing patterns are truly sinusoidal. Similar to [3], [10], [11], [27], we embed our three sinusoidal patterns into the RGB channels of a DLP projector.

Even though the wrapped phase can be determined at each pixel of the camera, given M periods of sinusoids in the projection pattern, there is an M position ambiguity in each wrapped phase measurement [27]. To determine the absolute phase of a pixel, we search for the single absolute phase offset $2\pi m$ with $m \in \{0, 1, \dots, M - 1\}$, which must be added to its wrapped phase to obtain its absolute phase. We refer to m as the offset index. Since the M possible corresponding positions in the projector are known, the location of the M points can be triangulated, as shown in Fig. 2 [27]. These triangulated points all lie along the ray coming out of the camera, illustrated as camera A in Fig. 2. Fig. 3(a) shows an example of the wrapped phase image of camera A, from Fig. 2, with a pixel of interest P identified by a red dot. The extrinsic relationship between the stereo

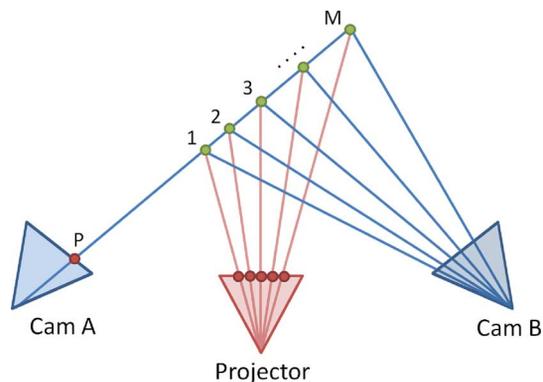


Fig. 2. Triangulation of the M possible positions of pixel P in camera A.

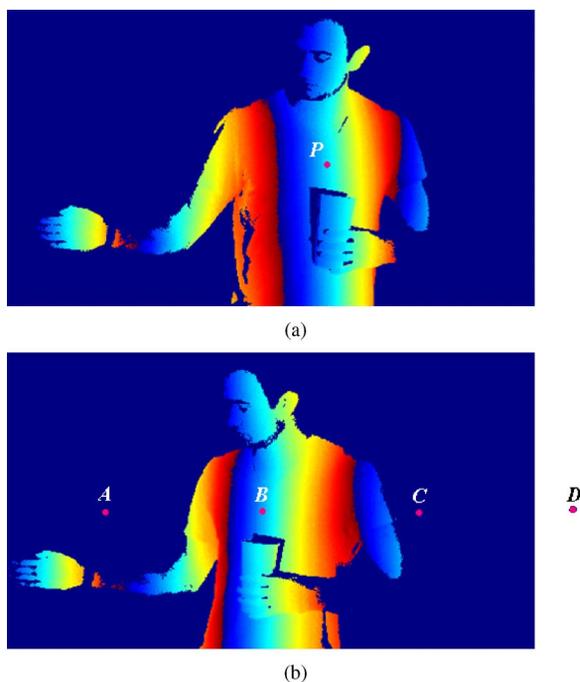


Fig. 3. (a) Wrapped phase image of the left camera A with the pixel of interest identified by the red dot P . (b) Wrapped phase image for the right camera B in a system with the M possible points projected onto the image.

cameras in Fig. 2, as well as each camera's intrinsic parameters can be used to project the M 3-D positions onto the second camera's image plane, as shown in Fig. 3(b). By comparing the wrapped phase values at the M pixel locations to the first camera's phase value, the absolute phase of P can be estimated. In Fig. 3(b), it is clear that point B is the corresponding match to the pixel P in Fig. 3(a). Points A and C map to blank areas of the scene, and point D is projected outside of the image frame.

Even though this approach works for most pixels, in practice it results in occasional errors. Common causes are occlusions, calibration errors, motion errors, or noise in the calculated phase. In some cases, the phase of the point in camera A is closer to an invalid point than to the correct one. In [27], the stereo unwrapped points are segmented into continuous regions and loopy belief propagation is used to perform an energy minimization on a cost function to locally smooth segmented regions.

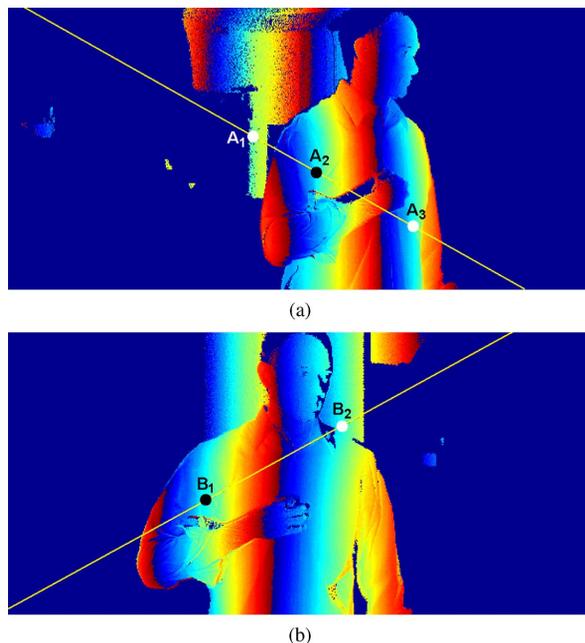


Fig. 4. Wrapped phase images with epipolar line plotted. Each circle identifies a pixel with the same wrapped phase as the projector pixel; image from (a) camera A and (b) camera B.

IV. VIEW CONSISTENT PHASE UNWRAPPING

Our proposed viewpoint consistent method consists of three steps. First, it is assumed that the cameras and projectors are calibrated with respect to each other. For each projector pixel, we determine its epipolar line in the two camera views, as shown in Fig. 4. By searching along the line in the images, we find correspondences between the camera views. The set of possible correspondences between the two views is treated as the set of labels for that pixel. Second, loopy belief propagation is applied to minimize an energy function that both quantifies the likelihood of each label and spatially smooths a set of labels [48], [49]. Finally, the remaining points that are not solved via the stereo approach are unwrapped with respect to their already unwrapped neighboring points. In what follows, we describe each step in detail.

A. Determining Possible Correspondences

To solve for the absolute phase of both cameras, we process each pixel in the projected pattern. Given a pixel P in the projected image, the epipolar line corresponding to P in each camera view can be determined, illustrated as the lines in Fig. 4(a) and (b). In each camera view, we identify the pixels along the epipolar line with the same phase as P , identified as $\{A_1, A_2, A_3\}$ and $\{B_1, B_2\}$ on the epipolar line in Fig. 4(a) and (b), respectively. Then, the 3-D position of each of these points is found by triangulating with point P , as illustrated in Fig. 5. All of the triangulated points lie along the ray out of the projector intersecting P .

Once the 3-D position of each possible point is determined, the distances between all pairs of candidate points in camera A and B are calculated. For a pair of points corresponding to the true correspondence, the triangulated 3-D positions are likely to be near one another, as illustrated by the black dots in Fig. 4.

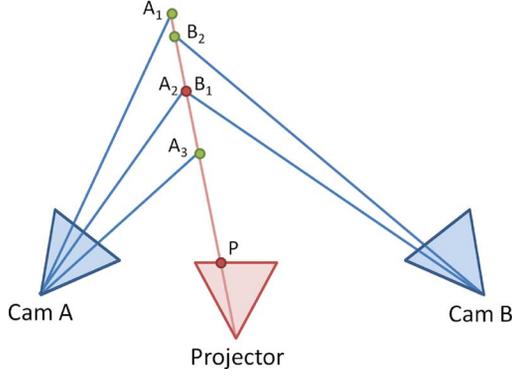


Fig. 5. Illustration of the stereo camera geometry used to identify correspondences across camera views.

Therefore, we can estimate the likelihood of a correspondence by the Euclidean distance between a pair of points.

B. Labeling Via Loopy Belief Propagation

After the set of possible correspondences and their associated likelihoods are determined for each pixel in the projector, an energy minimization framework is applied to determine the best label for each projector pixel. We only include projector pixels in the graphical model for which there is at least one pair of correspondences with a distance smaller than a set threshold. If there are no possible correspondences within this threshold, the projector pixel is not likely viewable from both cameras. The set of labels for each projector pixel is all possible pairwise correspondences between the detected points in each camera view. Specifically, in Fig. 4(a) and (b), the set of labels for pixel P are $\{(A_1, B_1), (A_1, B_2), (A_2, B_1), (A_2, B_2), (A_3, B_1), (A_3, B_2)\}$.

Similar to [27], we minimize a cost function consisting of a data and smoothing term:

$$E(f) = E_d(f) + E_s(f) \quad (2)$$

where f represents the labeling set for the pixels in the projector. The data component of the cost function comes from the likelihood of the correspondence as determined by the distance between the points in 3-D space:

$$E_d(f) = \sum_{p_m \in F, p_m = \{A_m, B_m\}} \min(\beta, \|l_{3D}(A_m) - l_{3D}(B_m)\|) \quad (3)$$

where p_m denotes the label of a pixel within the projector image F corresponding to a pixel A_m in camera A and B_m in camera B, β is a threshold to limit the penalty from occlusions, and $l_{3D}(A_m)$ denotes the 3-D location of the labeled point A_m . The smoothing component is chosen to enforce a solution that minimizes large discontinuities within the scene. In most SL systems, the depth of the scene is assumed to vary slowly. This means that neighboring pixels within the camera image should have similar depths, and thus similar phase values. Since our method operates on the projector pixels, the projector pixels are assumed to vary smoothly. The basic intuition is that if two neighboring pixels in the projector illuminate the smooth scene surface, the two pixels in the camera image corresponding to these points should also be neighbors. This assumption fails

when a scene discontinuity lies between the two examined pixels in the projector. Therefore, we enforce the following smoothness term:

$$E_s(f) = \sum_{\forall p_m, p_n \in F: p_m \in \mathcal{N}(p_n)} \min(\alpha, \|l_{2D}(A_m) - l_{2D}(A_n)\|) + \min(\alpha, \|l_{2D}(B_m) - l_{2D}(B_n)\|) \quad (4)$$

where p_m and p_n are neighboring points in the projector image, and A_m and B_m are the corresponding points for the pixel p_m in cameras A and B, respectively. The function l_{2D} is the 2-D image coordinates of the identified point, and $\mathcal{N}(p_n)$ is the set of four pixels that border p_n . The threshold α is set to minimize the penalty to occlusions.

To determine the label for the projector pixels that minimize (2), we apply loopy belief propagation [50]. The message passed between two neighboring pixels is constructed as

$$m_{p_m \rightarrow p_n}^i(f_n) = \arg \min_{f_m} \min(\beta, \|l_{3D}(A_m) - l_{3D}(B_m)\|) + \min(\alpha, \|l_{2D}(A_m) - l_{2D}(A_n)\|) + \min(\alpha, \|l_{2D}(B_m) - l_{2D}(B_n)\|) + \sum_{p_j \in \mathcal{N}(p_m) \setminus p_n} m_{p_j \rightarrow p_m}^{i-1}(f_m) \quad (5)$$

where f_n is the label assigned to pixel p_n and $m_{p_m \rightarrow p_n}^i(f_n)$ denotes the message passed from p_m to p_n in the i^{th} iteration. To minimize the energy cost function, messages are generated and passed to their neighboring pixels in each iteration. After a predetermined number of iterations is reached, the final label for each individual pixel is chosen by summing the final messages entering that pixel and choosing the label with the lowest cost. The final labeling is used to assign absolute phase offsets to the phase images in the two camera views. For each pixel in the projector, the energy minimization provides the most probable labeling. The final label of each projector pixel specifies the location where the projected pixel intersects the image plane of cameras A and B. Since the absolute phase for each point in the projected pattern is known, we can assign the identified image points with the proper absolute phase offset.

Noise in the captured images can lead to small errors in the decoded phase for each pixel. When searching for possible points of correspondence along the epipolar lines in the cameras, a small tolerance in the matched phase value is allowed to account for this noise. Because of this tolerance and the small errors in the system calibration, it is possible for multiple projector pixels to map to the same pixel in the camera, or for a pixel in the camera to have multiple neighboring correspondences in the other camera. For these cases, we keep the match with the lowest final computed cost. These small errors rarely lead to errors in our estimated absolute phase offsets since we only keep the absolute phase offset that is determined from the match rather than the absolute phase value of the projector's pixel.

C. Filling in Missing Phases

Unlike the previous two steps where the processing is done in the projector domain, in this step, missing phase values in camera images are filled in. As mentioned previously, our

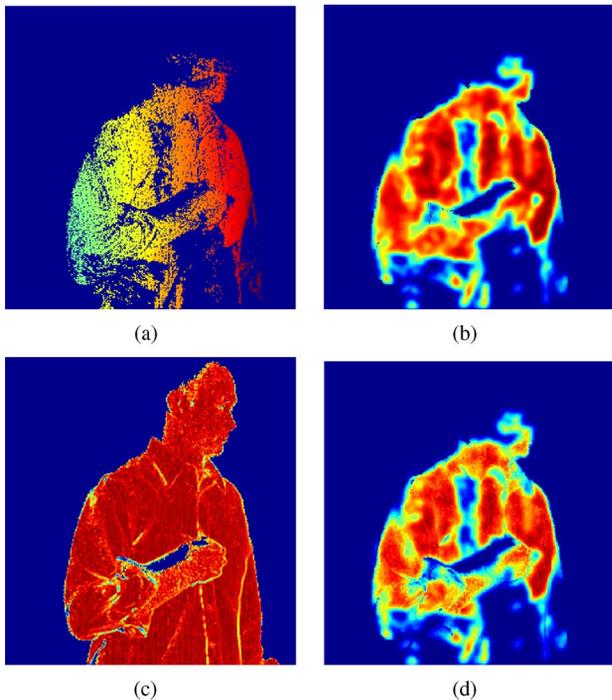


Fig. 6. Quality map components. (a) Pixels unwrapped using the approach of Sections IV-A and IV-B. (b) Density of stereo unwrapped points. (c) Local derivative measure. (d) Final quality map.

graphical model only provides absolute phase values for projector pixels that likely have a correspondence. Since not all projector pixels can be labeled, not all the pixels in the camera images can be labeled either. Nevertheless, it is possible to determine the absolute phase of these remaining pixels by local spatial phase unwrapping with respect to the already unwrapped points.

Before unwrapping the remaining points, a quality map is derived for each camera's phase image. In 2-D spatial phase unwrapping, quality maps quantify the confidence in unwrapping a local region of an image and are often based on local rates of change in the phase [19]. Regions of the phase image with slowly varying phase are unwrapped first since they result in fewer errors. In our implementation, the quality map is generated by computing local derivatives in the original wrapped phase image and also by calculating the density of points with assigned phases via stereo unwrapping. Fig. 6(a) illustrates the camera points with phases from the stereo observations described in this section. Fig. 6(b) illustrates the local density of absolute pixels from Fig. 6(a), Fig. 6(c) the local derivatives, and Fig. 6(d) the final quality map. The final quality map is generated by point wise multiplying the density image and the local derivative image. Once the quality map is generated, the points without an assigned absolute phase are unwrapped. Specifically, each pixel in the quality map is assigned a value from 0–1, and the quality values are placed into bins that span 0–1. Starting with the highest quality bin, we iterate through the non-unwrapped pixels in that bin and unwrap camera pixels that are neighbors to already absolute unwrapped pixels. After all pixels in a bin are unwrapped, the pixels in the next quality bin are unwrapped. A list of the remaining unwrapped pixels

is maintained, and the unwrapping process continues until the list of remaining pixels is empty.

V. OVERVIEW OF THREE-DIMENSIONAL PHASE UNWRAPPING

Multidimensional phase unwrapping has been an area of active research for many years, with a substantial body of work dedicated to unwrapping in two dimensions [15], [19]. Applications for these methods include synthetic aperture radar, medical imaging, and SL systems [19]. In recent years, many of the two dimensional techniques have been extended to three and higher dimensions [15].

The basic idea behind most multidimensional phase unwrapping algorithms is to integrate along paths of neighboring phase values in order to determine the relative phase between all pixels. This is a trivial problem when the true phase difference between neighboring values is less than $|\pi|$ and noise does not push the resulting phase values outside this range. Since, in most applications, this assumption is invalid, phase unwrapping algorithms are designed to minimize the impact of these erroneous regions.

Most 3-D phase unwrapping algorithms can be classified into one of three groups: 1) global-error minimization; 2) residue detection; and 3) quality guided methods [19]. Of these basic approaches, quality guided methods have proven to be both computationally efficient and robust [15]. The basic idea behind them is to quantify the likelihood that two neighboring pixels can be correctly unwrapped with respect to each other. An edge is defined to exist between all neighboring pixels and a measure of quality is assigned to each edge [15]. If two neighboring pixels have similar phase values, the quality of that edge is considered to be high. In contrast, two pixels with greater phase difference would have a lower quality edge. Although, basic gradient calculations between pixels have been used to generate quality maps [19], other methods, such as a second difference quality map, have been found to be more robust [15].

The second difference value for each pixel (i, j, k) in the phase volume can be computed as

$$SD_{ijk} = \sqrt{H_{ijk}^2 + V_{ijk}^2 + N_{ijk}^2 + \sum_{n=1}^{10} D_{n,ijk}^2} \quad (6)$$

$$H_{ijk}^2 = \gamma[\phi_{i-1,j,k} - \phi_{i,j,k}] - \gamma[\phi_{i,j,k} - \phi_{i+1,j,k}] \quad (7)$$

$$V_{ijk}^2 = \gamma[\phi_{i,j-1,k} - \phi_{i,j,k}] - \gamma[\phi_{i,j,k} - \phi_{i,j+1,k}] \quad (8)$$

$$N_{ijk}^2 = \gamma[\phi_{i,j,k-1} - \phi_{i,j,k}] - \gamma[\phi_{i,j,k} - \phi_{i,j,k+1}] \quad (9)$$

where H , V , and N represent the horizontal, vertical, and normal, i.e., temporal, second differences via discrete Laplacians respectively. In addition, $\phi_{i,j,k}$ represents the wrapped phase at (i, j, k) and γ is a function that wraps its parameter to a value within $[-\pi, \pi)$. The ten diagonal second difference components $D_{n,ijk}^2$, $n \in \{1, \dots, 10\}$ in (6) are computed through all of the diagonal paths in the $3 \times 3 \times 3$ volume centered on the pixel of interest. The squared second derivative identifies pixels where a discontinuity occurs. If there are no discontinuities, the phase increase should be approximately linear, resulting in a squared second difference value near zero; on the other hand, if the squared second difference value is large, there is a strong likelihood of a discontinuity. Using

this quality map, it is possible to identify pixels with spatial or temporal discontinuities.

The quality of a pixel is inversely related to the sum of second difference values. Once a quality value is assigned to each pixel in the 3-D phase volume, the quality of each edge is simply defined as the sum of the quality of each pixel connected to the edge [15]. The quality of edges determines the order in which the edges in the volume are unwrapped by sorting them from highest to lowest.

When an edge is evaluated, the pixels connected to the edge are unwrapped with respect to one another, forming links in a chain. Each chain has a pixel that is designated the lead pixel. When the first link of a chain is created, the lead pixel is defined as the pixel with the highest quality. As each new pixel is added to a chain, the new pixel is unwrapped with respect to the connected pixel along the evaluated edge. Each newly added pixel is then given a phase value relative to the lead pixel in the chain. When two chains are merged, each pixel in the smaller chain is assigned a phase relative to the lead pixel in the larger chain [15]. Chains grow in connected 3-D volumes and their actual shape is unimportant as long as the phase of each pixel in the chain is defined with respect to the lead pixel.

While this 3-D algorithm is capable of accurately unwrapping the phases of all points within a volume, it does not necessarily result in absolute phase. Since each absolute phase is assigned to a column of pixels within the sinusoidal pattern, correct triangulation is only possible when the absolute phase is known; thus, without a method to recover the absolute phase, the 3-D unwrapping algorithm is useless in the SL system.

Additionally, the algorithm in [15] is only capable of unwrapping scenes with one contiguous, connected region. While this works in MRI, the application domain in which this algorithm was originally designed, it is not applicable to depth recovery for scenes with multiple disjoint objects as it could incorrectly assign relative phases to each object. This is because in [15] all edges between pixels with a wrapped phase are unwrapped with respect to one another regardless of the quality of the edges. For example, consider a scene with a ball traveling in front of a flat board. As the ball moves, the pixels that fall along the edge of the ball and the background receive low quality measures due to the motion and depth discontinuity. Therefore, the algorithm [15] is likely to assign erroneous phases to those pixels. However, within the ball, as the local depth is not changing significantly, we would expect these pixels to have high quality edges. Unwrapping the background and ball independently prevents the two regions from being incorrectly unwrapped. As long as the absolute phase of each region can be determined, e.g., through stereo phase unwrapping, there is no need to unwrap across low quality edges.

The shortcomings of the phase unwrapping algorithm in [15] are a direct consequence of the fact that it has been developed for applications in which only a single observation exists for each phase value. In an application such as MRI, since each point is only measured or observed once there is no choice but to unwrap all pixels with respect to one another. However, in an SL system with a second camera to disambiguate the absolute phase of each pixel, unwrapping can be restricted to high-quality edges, i.e., those edges connecting pixels with similar phase values, thus

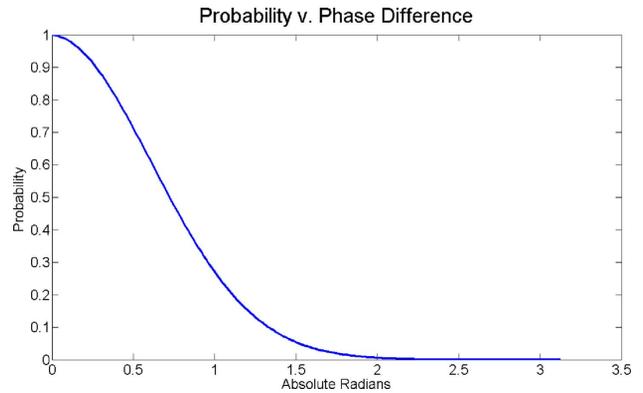


Fig. 7. Converting absolute phase differences to probabilities.

reducing the error. In the following section, we exploit this to develop a method for combining stereo phase unwrapping with 3-D phase unwrapping to overcome some of the shortcomings of [15].

VI. TEMPORALLY CONSISTENT PHASE UNWRAPPING

In this section, we develop a probabilistic framework for stereo phase unwrapping in order to augment the functionality of the 3-D phase unwrapping algorithm of [15] to make it capable of determining absolute phase.

A. Determining a Pixel's Absolute Phase Offset Probabilities

As mentioned earlier, the likelihood of a pixel having a specific absolute phase offset is determined by triangulating the M possible corresponding points between camera A and the projector, as shown in Fig. 2, projecting these M points onto the second camera's image, as shown in Fig. 3(b), and comparing the phases of the M projected points to the phase of the original pixel P in camera A. With pixel's taking on wrapped phase values from $-\pi$ to π , if the original pixel P and one of the corresponding M projected pixels have a phase difference of 0, they are likely to be a correct match; likewise, if the pair differ by $|\pi|$ then they are unlikely to be a correct match.¹ One way to compute the likelihood for each of the M possible absolute phase offsets is to divide the phase difference of the corresponding pixels by π . However, this likelihood measure does not sufficiently penalize large phase differences. Rather, we propose using a Gaussian function to assign likelihoods to phase differences, as shown in Fig. 7.

In recovering the absolute phase of each pixel, not all of the M corresponding points necessarily project onto valid regions in the second image as shown earlier in Fig. 3. Some could fall on portions of the image where there are no wrapped phase values, as indicated by dots A and C in Fig. 3(b), and some might fall outside the viewable region of the camera, as indicated by dot D in Fig. 3(b). We assign σ_B and σ_E as the probabilities for 3-D points projected outside the boundary and to an empty "phase-less" region of the image, respectively. Both σ_B and σ_E are set greater than zero to ensure we do not penalize non-overlapping views or occlusions between the cameras too harshly during unwrapping. We note that some pixels near the

¹When calculating the difference between the phase values, the wrapping function γ , from (7)–(9), is used to ensure the difference is in the range $[-\pi, \pi)$

edges of camera A's image may not be visible from the second view. For these cases, we assign a probability between σ_B and σ_E . Once the probabilities for all of the possible absolute phase offsets are determined, they are normalized to generate a probability distribution for the absolute phase offset of pixel P in camera A.

B. Using Pixel Probabilities to Build Chains

In [15], each time a pixel is connected to an existing chain, or when two chains are combined, the newly added pixels are unwrapped relative to their linked neighbors. The relative phase between all pixels in a chain is used to determine the phase difference between the lead pixel and each pixel in the chain. Since we only connect "high quality" pixels with reliable edges, all the pixels in a chain require the same phase offset to convert their relative unwrapped phase to the absolute phase.² To determine the absolute phase of the chain, we search for the single absolute phase offset $2\pi m$, where $m \in \{0, 1, \dots, M-1\}$, which must be added to the chain's lead pixel's wrapped phase to obtain its absolute phase. We refer to m as the offset index for that chain. If the absolute phase of the lead pixel and the relative phase difference between the lead pixel and each pixel in the chain are known, then the absolute phase for each pixel in the chain can be determined.

The key to ensuring the pixels in the chain have the correct absolute phase is to use the phase probability distributions of each pixel in the chain to determine the absolute phase of the lead pixel. As explained shortly, the basic idea is to use the phase probability distribution of each newly added pixel in the chain to update the probability distribution of the offset index for the lead pixel of the chain. Once the chain is completed, the lead pixel offset with the highest likelihood is chosen to be the phase offset for the entire chain.

Similar to the unwrapping method in [15], we start by assigning quality values to all of the pixels and edges. The quality value of an edge is the sum of the quality of the two pixels connected to the edge. Once the quality of all edges has been determined, the pixels are ranked in quality from highest to lowest.

To start the unwrapping process, the most reliable edge is identified and the pixels connected to the edge are unwrapped with respect to each other to form a two pixel chain. We set the pixel with higher quality as the lead pixel of the chain. We need to combine the probability distributions of the absolute phase offset for these two pixels in order to determine the probability distribution for the absolute phase offset of the leading pixel, or equivalently the whole chain. One way to do this is to add the log probabilities of the two pixels. However, this can only be done after accounting for the phase difference between the two connected pixels.

To illustrate this, consider two possible chains, each with two elements. The first chain, shown in Fig. 8(a), has a lead pixel with a wrapped phase of $\pi - 2\delta$ and a second pixel with a phase of $\pi - \delta$, where $0 < \delta \ll \pi$. Local unwrapping would imply that the two points are from the same sinusoidal period. This

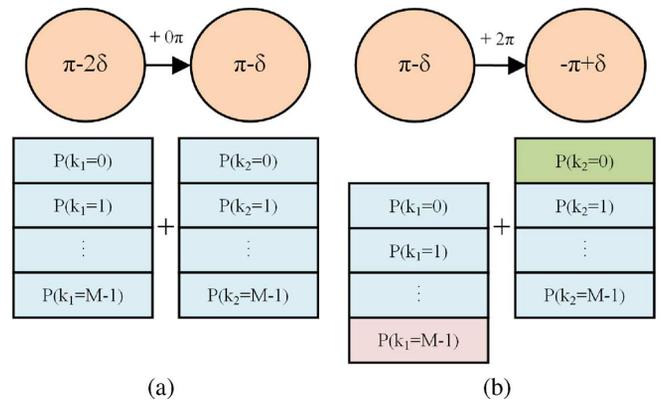


Fig. 8. Addition of log probabilities for two pixels (a) in the same period and (b) from different periods.

means that the \log^3 probability distributions could be directly added together to refine the offset index state of the lead pixel:

$$\log P(c_i = m) \propto \log P(p_j = m) + \log P(p_k = m) \quad (10)$$

where c_i is the offset index state of the lead pixel of the i^{th} chain given the influence from the connected pixels, p_j is the offset index state of the lead pixel and p_k is the offset index state of the second pixel, with offset index $m \in \{0, 1, \dots, M-1\}$.

Now consider the second case, shown in Fig. 8(b), where the lead pixel has a value of $\pi - \delta$ and the second pixel has a value of $-\pi + \delta$, where $0 < \delta \ll \pi$. For this case, the second pixel is unwrapped into a period that is outside the period corresponding to the lead pixel. Specifically, the second pixel has an absolute phase offset that is 2π greater than the lead pixel, or equivalently an offset index that is one greater than the lead pixel. Therefore, when we add the log probabilities to determine the absolute phase distribution for the whole chain, we must shift the individual pixel probability distributions, as shown in Fig. 8(b):

$$\log P(c_i = m) \propto \log P(p_j = m) + \log P(p_k = m + 1). \quad (11)$$

The same procedure is used when adding a single pixel to an already formed chain. As each new pixel is added to the chain, the absolute phase offset distribution of the new pixel is first properly shifted to reflect the relative phase difference between the lead pixel and the new pixel; then the log probabilities for the new pixel are added to a running sum for the chain. In essence, this represents the contribution of the phase probability distribution of the new pixel to the overall phase offset of the chain it joins.

In summing log probabilities as pixels are added, we keep a count of the number of pixel probabilities added to each possible offset index. Due to the shifting, each possible offset index might have a different number of log probabilities summed together. For example, in Fig. 8(b), the lead, left, pixel in the chain has two log probabilities summed together for $k_1 \in [0 \dots M-2]$, shown in blue, but only one for $k_1 = M-1$, shown in pink. We know the second pixel in the chain cannot have an offset index of 0, shown in green, since this would imply the lead pixel to have an offset index of -1 . This is not

²In essence, we are assuming all the pixels in a chain are part of a contiguous object in a scene without depth discontinuities.

³Due to numerical stability issues, we opt to add the log of probabilities rather than multiplying them.

possible since $m \in \{0, 1, \dots, M - 1\}$. To determine the final absolute phase offset for the lead pixel in the chain, the average log probability for each absolute phase offset is computed and the absolute phase offset with the maximum average log probability is selected. This absolute phase offset is then used to determine the absolute phase for all the pixels in the chain.

For the case where two chains are connected together, the smaller chain is unwrapped relative to the larger chain. That is, each pixel in the smaller chain is unwrapped relative to the lead pixel in the larger chain. In addition, the set of summed log probabilities for the smaller chain is shifted to reflect the relative phase offset between the lead pixels of the two chains. The shifted log probabilities are then added to the summed log probabilities of the larger chain. This is similar to the probability distribution shifting and summing explained in the two pixel chain examples in (10) and (11) except that it is being carried out with respect to the lead pixels in the two chains rather than individual pixels.

C. Removing Edges

The advantage of using stereo phase unwrapping to anchor each chain is that it is no longer necessary to unwrap across all edges, which is done in [15]. This allows us to determine the absolute phase of spatially disjoint objects in the scene; in addition it prevents us from unwrapping across unreliable low quality edges that may have true phase differences greater than $|\pi|$. Local unwrapping fails when the true phase differences between neighboring pixels are greater than $|\pi|$.

By limiting unwrapping to edges with quality greater than a minimum threshold, we reduce the probability of unwrapping across spatial or temporal discontinuities. By adding the time dimension to unwrapping, small regions are unwrapped more accurately since more observations are merged over time.

In our implementation, we choose a minimum quality value to be met in order for the edge to be considered valid. Even though this results in several smaller chains at the end of the unwrapping process, the points within each chain are more likely to be correctly unwrapped. In general, erroneous unwrapping only occurs along the edges between discontinuous regions.

VII. EXPERIMENTAL SETUP AND RESULTS

To test the presented algorithms, we have developed an SL system similar to the one in [27], but without using the color camera for texture capture. The system consists of two Point Grey Dragonfly Express cameras with a resolution of 640×480 and 1394b ports. We use an Optoma TX780 projector with a resolution of 1024×768 , operating at 60 Hz. Similar to many other SL systems utilizing a DLP projector, the color wheel has been removed to increase the rate at which grayscale patterns are projected to 180 Hz [10], [51].

To illustrate the effectiveness of our proposed algorithm, we have captured several data sets. Once the correspondences are found between the camera and projector, the depth of each pixel is determined via triangulation. We present results on the accuracy of the estimated absolute phase of the scene, or equivalently the accuracy of the reconstructed point clouds.

Our current implementations of the viewpoint and temporally consistent unwrapping algorithms do not operate in real time.

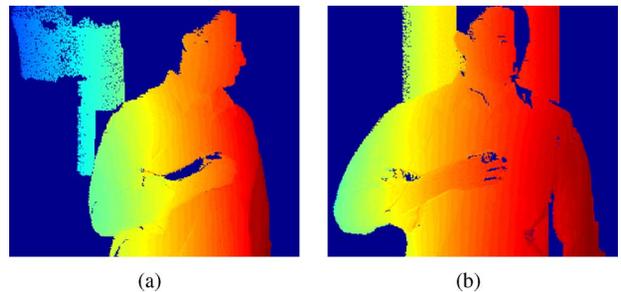


Fig. 9. Final unwrapped images using our algorithm of Section IV for (a) camera A, corresponding to Fig. 4(a); (b) camera B, corresponding to Fig. 4(b).

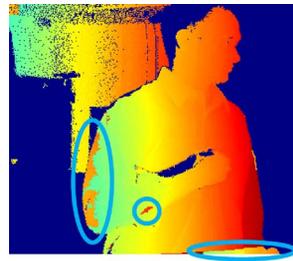


Fig. 10. Unwrapped image for camera A, corresponding to Fig. 4(a), using the method in [27]. Erroneous unwrapped regions are circled.

Generating the stereo matching data for both the viewpoint and temporal unwrapping methods requires several points to be triangulated for each pixel in the phase volume. The independent nature of these calculations allows for the stereo matches to be computed using a highly parallelized GPU algorithm. For the viewpoint algorithm, the greatest bottleneck is in solving for the correspondences between camera views using loopy belief propagation. To speed up processing, a more efficient parallel loopy belief propagation algorithm could be used. Once the set of available correspondences are solved, unwrapping the remaining pixels is completed in linear time. For the temporal unwrapping method, the bottleneck is due to iterating through all the pixels in the phase volume. This step cannot be easily parallelized since the chains are built up throughout the entire phase unwrapping volume. In our non-optimized implementations, the unwrapping process requires on average less than 10 seconds per frame.

A. Viewpoint Consistent Unwrapping Results

Fig. 9(a) and (b) show the unwrapped results using our proposed viewpoint consistent method of Section IV for the camera images shown in Fig. 4(a) and (b), respectively. The algorithm is clearly effective in properly unwrapping the illustrated phase images. For comparison purposes, Fig. 10 shows the unwrapping results for Fig. 4(a) using our implementation of the approach described in [27], but without using the left to right consistency check. The consistency check is likely able to remove some outlying pixels, but would not correct the mislabeled segments encircled in Fig. 10. Clearly, the reconstruction quality of Fig. 9(a) is higher than that of Fig. 10. In a sample sequence of 700 frames, our method provides more accurate results than [27] in 80% of frames, and performance comparable to or better than [27] in 96% of frames. More specifically, in a sample sequence of the 700 phase images, unwrapping via [27] results in an average error of 3.2% of incorrect foreground pixels compared to

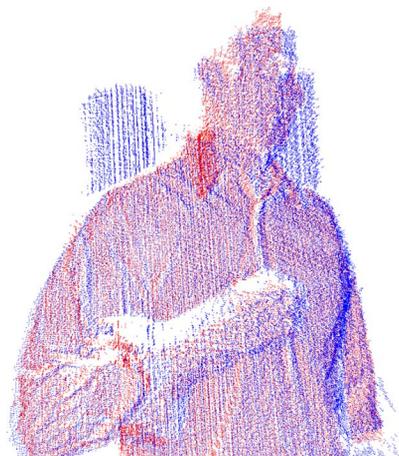


Fig. 11. Merged camera A (red) and B (blue) points clouds resulting from unwrapped phases of Fig. 9.

0.5% for our method. In our experiments, we have found that nearly all unwrapped images experience at least a few incorrectly unwrapped pixels. Unlike [27], in our experiments we do not compensate for scene motion. The amount of motion in our experiments has been deliberately limited in order to reduce the errors in decoding the phase value at each scene point. Despite our efforts, we do find that some of the decoded phase values are incorrect, especially along the edges of the scene. For these cases, we find that our algorithm is still able to unwrap the edge pixels assuming the error is not too large. A video comparing the two methods can be found on our website at [52]. The merged point cloud generated from the two unwrapped phase images in Fig. 9(a) and (b) is shown in Fig. 11. As seen, the merged point cloud is denser than either component; furthermore, the two components are reasonably aligned, and provide increased scene coverage by alleviating occlusion.

B. Temporally Consistent Unwrapping Results

We show our algorithm's ability to handle scenes with disjoint objects. Fig. 12(a) shows the unwrapped phase of a scene with disjoint objects using our proposed algorithm in Section VI. Fig. 12(b) illustrates how the algorithm in [15] fails to correctly unwrap the disconnected regions inside the ellipse. In the phase image, a cone shaped object is lifted off of a box. The cone and box objects are positioned at approximately the same depth, so they should have nearly the same unwrapped phase values as shown in Fig. 12(a). Since there are no connections between the two regions, there is no way for the algorithm in [15] to determine the phase difference between the isolated segments. Our method handles this by using the added information from the stereo camera as long as the points are visible in both cameras. While existing SL approaches using stereo cameras enforce only spatial smoothness during the phase unwrapping process [27], our approach enforces both spatial and temporal smoothing. To characterize the advantage of temporal smoothing, we have compared our algorithm with the stereo phase unwrapping method presented in [27]. Since our method only determines the unwrapped phase of one camera at a time, to do a fair comparison we do not implement the left-to-right and right-to-left consistency check described

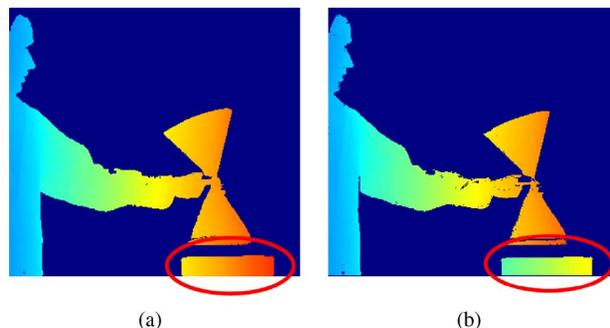


Fig. 12. (a) Correctly unwrapped scene using our proposed algorithm of Section VI. (b) Incorrect unwrapping for the algorithm in [15].

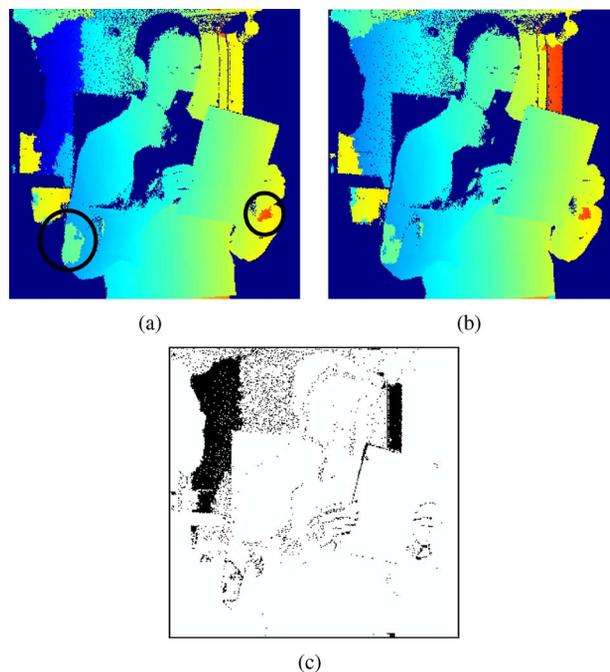


Fig. 13. (a), (b) Two successively unwrapped phase images using the method in [27]. (c) Regions of phase difference in consecutive images.

in [27]. This type of consistency check could easily be applied to our method to provide the same benefits as the approach in [27]. In the loopy belief propagation implementation, we process ten iterations before generating the final labeling for all segments. Figs. 13(a)-(b) and 14(a)-(b) show two successively unwrapped phase images resulting from the algorithm in [27] and our proposed algorithm, respectively. As seen, each image in Fig. 13(a)-(b) suffers from more artifacts than the corresponding one in Fig. 14(a)-(b). More importantly, the images in Fig. 14(a)-(b) are more temporally consistent than those in Fig. 13(a)-(b). The unwrapped phase difference between the successive images in Figs. 13(a)-(b) and 14(a)-(b) are shown in Figs. 13(c) and 14(c), respectively. As seen, the areas with phase difference in Fig. 13(c) are significantly smaller and less noticeable than in Fig. 14(c). Specifically, in Fig. 13(c), 15.3% of pixels are not consistent over time as compared to 3.4% in Fig. 14(c); some of the 3.4% corresponds to actual motion in the scene. Visual inspection of the two phase video sequences also confirms the temporal consistency of our approach as compared to spatial only smoothing which exhibits a significant amount

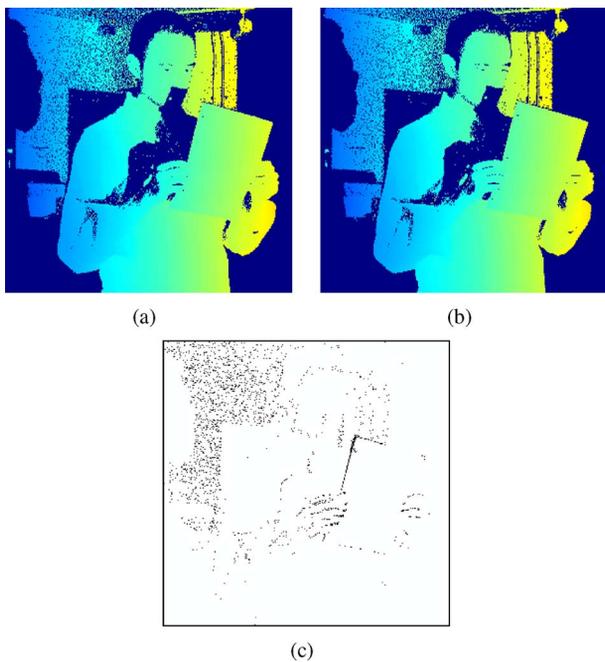


Fig. 14. (a), (b) Two successively unwrapped phase images with our proposed algorithm of Section VI. (c) Regions of phase difference in consecutive images.

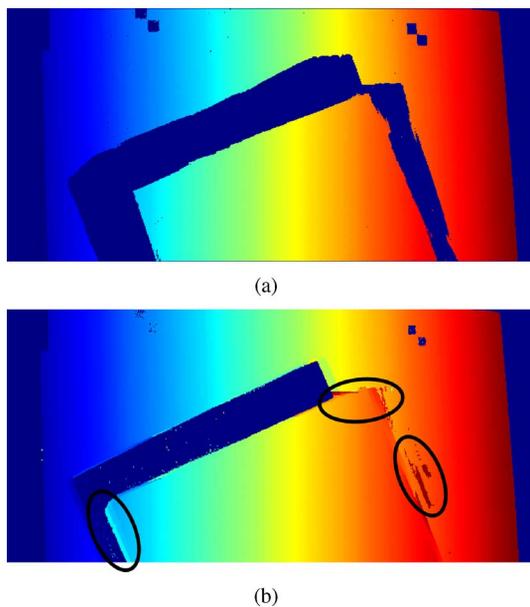


Fig. 15. Front board is moving fast in front of back flat surface; unwrapping by (a) our proposed algorithm of Section VI; (b) the algorithm in [15].

of flicker. The comparison video can be found on our website at [53].

In [15], the edges are sorted and unwrapped sequentially according to the quality of the connection between pixels. Since the lower quality pixels are unwrapped at the end, any potential errors only affect the few pixels that follow in unwrapping order. In our setup, it is not necessary for every pixel to be unwrapped relative to all of its neighboring pixels since we combine the absolute phase positioning from the stereo camera. Therefore, our proposed algorithm eliminates the need to unwrap between pixels with low quality edges. Fig. 15 shows the same scene unwrapped with our proposed algorithm and that of [15]. In

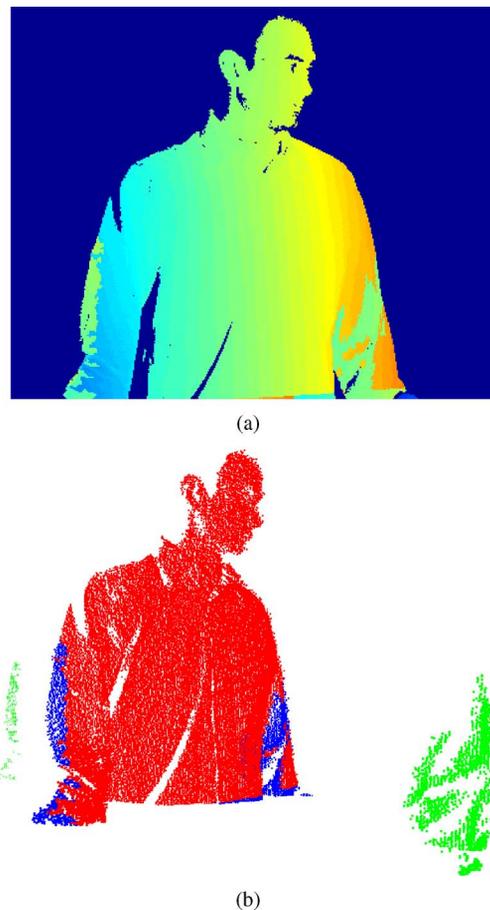


Fig. 16. Example of 3-D geometry from incorrectly unwrapped phase image generated via [27]. (a) Incorrectly unwrapped phase image with errors in the arms and bottom of image. (b) Resulting geometry; points in green are due to incorrect unwrapping of some regions in (a); points in blue result from our view-point consistent algorithm which correctly unwrapped those regions.

Fig. 15(b) resulting from [15], the black circled regions contain points that are unwrapped incorrectly. The phase in these regions changes rapidly both spatially and temporally due to the discontinuity between the two planes and the high motion of the plane. As seen, the low-quality edges do not allow for accurate unwrapping.

Both our methods are effective for scenes in which the majority of the points in the scene can be viewed by both cameras. In situations where the two camera views are significantly different, neither of our stereo assisted methods is able to correctly unwrap many points in the scene.

VIII. DISCUSSION

Even though the main focus of this paper has been phase unwrapping, it is informative to examine the effect of unwrapping on the quality of the resulting 3-D point cloud. The errors due to incorrect phase unwrapping are readily noticeable in the resulting 3-D point clouds. Fig. 16(a) shows an incorrectly unwrapped phase image generated via [27] with errors in the arms and along the bottom of the image. The resulting 3-D geometry for this image is shown in Fig. 16(b). The red points represent the correctly unwrapped points, while the points in green represent the incorrectly unwrapped segments. Applying our view-point consistent method of Section IV, we obtain blue points in

Fig. 16(b) which are a result of correct unwrapping. As shown in this example, it is common to have unwrapping errors occur over local segments.

One of the fundamental causes of error in the stereo SL method is the false matches in the projected 3-D points [27]. When the M points are projected onto the second camera view it is possible for one or more of the non-corresponding points to fall onto an image point that has a similar phase value to the one being searched for. The likelihood of this depends on the scene structure as well as the number of periods in the projection pattern. When the number of periods M in the projected pattern increases, so does the probability of a false match.

In our viewpoint consistent unwrapping method, it is also possible to incorrectly choose the correspondence by relying on the stereo match data alone. Specifically, possible corresponding 3-D locations from the left and right cameras all lie along the ray out of the projector that corresponds to the projector pixel being evaluated. In this setup, the correctness of points is based on the closeness in Euclidean space of the corresponding 3-D points in the left and right view rather than closeness in phase. Switching the distance metric from a Euclidean distance to a phase difference does not necessarily guarantee better results. However, by enforcing the 3-D and 2-D cost functions in the energy minimization, our viewpoint consistent method is able to generate accurate results.

The viewpoint consistent method offers an advantage over [27] in that it determines the absolute phase of corresponding pixels in the two camera views, thus ensuring consistency in the final unwrapped phase between the two views. In [27], an estimate of the absolute phase for each pixel is fed into a loopy belief propagation algorithm. Despite using stereo information to determine the absolute phase estimate of all the pixels in the two camera views, there is no attempt to directly enforce the corresponding pixels in the two camera views to the same absolute phase value. For example, a pixel in the left camera may find that its true corresponding pixel in the right camera has the lowest correspondence cost, and thus the absolute phase is correctly set for the left camera. However, the actual corresponding pixel in the right camera may not find the original pixel in the left to have the lowest cost. Although the loopy belief propagation step in [27] can clear up some of these errors, it does not ensure consistency across the two cameras.

In [27], the stereo matching information is used to select an initial offset for each pixel. It is acknowledged that the stereo information for a single pixel does not always indicate the correct absolute phase for a pixel. However, hard decisions are made with these stereo measurements and the resulting errors are later corrected by segmenting the initial unwrapped phase image and by applying an energy minimization on the mislabeled unwrapped image. In contrast, rather than selecting an individual offset for each pixel, our temporal method selects an offset for a group of pixels that are first locally unwrapped according to an edge quality measure. By restricting local unwrapping to edges with high measures of reliability, the correct relative phase between all pixels in a chain can be determined with high confidence. Once the relative phase of all pixels within the chain is known, only a single offset needs to be chosen in order to find the absolute phase for all pixels within the chain. The

final probability distribution for the chain is generated by combining the “noisy” probability distributions of each pixel. In this detection/estimation problem, many random variables are combined to generate a new random variable with a distribution that has a lower likelihood of error.

In [27], the estimated absolute phase is segmented by combining pixels with a phase difference of less than $|\pi|$. Phase unwrapping based on local gradients has been shown to be less accurate than methods using other edge quality measures such as discrete Laplacians [15]. In addition, once segmentation is completed, the local smoothing component of the energy cost function in [27] only operates along the edges of segments. If incorrect labeling occurs within the interior of a segment, the local smoothing cost component cannot detect the error. In contrast, all edges between pixels in our temporal method are examined, and are only connected if the edge is reliable.

The most significant difference between our temporal method and existing phase unwrapping methods for SL is the enforcement of temporal consistency in the unwrapped phase maps [27]. In doing so, we take advantage of the inherent temporal correlation between successively captured phase images. If the capture rate of a system is high enough to ensure that the phase at a single point does not change by more than $|\pi|$ in successive frames,⁴ then the same spatial local smoothing assumptions during phase unwrapping can also be applied in the temporal domain. In our case, consistent labeling over time is ensured by building larger chains that span multiple frames. The offset for each of these larger chains is determined more accurately as long as only high quality edges are used during local unwrapping.

Finally, our temporal method improves the phase unwrapping results obtained by applying the technique in [15] to a structured light system. The primary motivation of the phase unwrapping approach in [15] is magnetic resonance imaging applications. Unlike [15], by using the stereo observations from two cameras, our algorithm can determine absolute rather than relative phase values.

IX. CONCLUSION

We have proposed two schemes for consistent phase unwrapping in a stereo SL system. The stereo cameras in our system not only aid in unwrapping phase, but also allow for reconstruction of a class of disjoint scenes not possible in the single camera case. Our viewpoint and temporally consistent phase unwrapping methods are more robust than simple stereo assisted phase unwrapping alone. Enforcing viewpoint and temporal consistency results in fewer errors in unwrapped images and more accurate point clouds.

In future work, the basic temporal unwrapping idea could be combined with the viewpoint consistent approach. Rather than running the energy minimization across the pixels of a single projected image, a larger graph could be constructed with multiple projector frames. At least one new data term would have to be added to enforce a smoothing in the labeling that occurs across time. Another improvement is to apply a short sliding temporal window in order to keep the computational complexity manageable.

⁴For scenes with high speed motion, phase recovery at each pixel is erroneous if not compensated [27].

REFERENCES

- [1] F. Blais, "Review of 20 years of range sensor development," *J. Electron. Imag.*, vol. 13, no. 1, p. 231, 2004.
- [2] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, pp. 7–42, 2002.
- [3] S. Zhang and P. Huang, "High-resolution, real-time 3D shape acquisition," in *Proc. Comput. Vis. Pattern Recogn. Workshop*, 2004, vol. 3, p. 28.
- [4] H. Kawasaki, R. Furukawa, R. Sagawa, and Y. Yagi, "Dynamic scene shape reconstruction using a single structured light pattern," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recogn.*, 2008, vol. 0, pp. 1–8.
- [5] O. Hall-Holt and S. Rusinkiewicz, "Stripe boundary codes for real-time structured-light range scanning of moving objects," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2001, vol. 2, pp. 359–366.
- [6] J. Salvi, J. Pagés, and J. Batlle, "Pattern codification strategies in structured light systems," *Pattern Recogn.*, vol. 37, no. 4, pp. 827–849, 2004.
- [7] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz, "Space-time stereo: a unifying framework for depth from triangulation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 296–302, Feb. 2005.
- [8] L. Zhang, B. Curless, and S. Seitz, "Spacetime stereo: Shape recovery for dynamic scenes," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recogn.*, Jun. 2003, vol. 2, pp. II-367–74.
- [9] M. Young, E. Beeson, J. Davis, S. Rusinkiewicz, and R. Ramamoorthi, "Viewpoint-coded structured light," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. CVPR '07.*, Jun. 2007, pp. 1–8.
- [10] P. S. Huang and S. Zhang, "Fast three-step phase-shifting algorithm," *Appl. Opt.*, vol. 45, no. 21, pp. 5086–5091, Jul. 2006.
- [11] S. Zhang and S.-T. Yau, "Absolute phase-assisted three-dimensional data registration for a dual-camera structured light system," *Appl. Opt.*, vol. 47, no. 17, pp. 3134–3142, Jun. 2008.
- [12] S. Zhang, X. Li, and S.-T. Yau, "Multilevel quality-guided phase unwrapping algorithm for real-time three-dimensional shape reconstruction," *Appl. Opt.*, vol. 46, no. 1, pp. 50–57, Jan. 2007.
- [13] R. R. Garcia and A. Zakhor, "Projector domain phase unwrapping in a structured light system with stereo cameras," in *Proc. 3DTV Conf.: The True Vision – Capture, Transmiss., Display of 3D Video (3DTV-CON)*, May 2011, pp. 1–4.
- [14] R. R. Garcia and A. Zakhor, "Temporally-consistent phase unwrapping for a stereo-assisted structured light system," in *Proc. Int. Conf. 3D Imaging, Modeling, Processing, Visualiz. Transmiss. (3DIMPVT)*, May 2011, pp. 389–396.
- [15] H. S. Abdul-Rahman, M. A. Gdeisat, D. R. Burton, M. J. Lalor, F. Lilley, and C. J. Moore, "Fast and robust three-dimensional best path phase unwrapping algorithm," *Appl. Opt.*, vol. 46, no. 26, pp. 6623–6635, Sep. 2007.
- [16] R. M. Goldstein, H. A. Zebker, and C. L. Werner, "Satellite radar interferometry – two-dimensional phase unwrapping," *Radio Science*, vol. 23, no. 4, pp. 713–720, 1988.
- [17] D. J. Bone, "Fourier fringe analysis: the two-dimensional phase unwrapping problem," *Appl. Opt.*, vol. 30, no. 25, pp. 3627–3632, Sep. 1991.
- [18] R. Gens, "Two-dimensional phase unwrapping for radar interferometry: Developments and new challenges," *Int. J. Remote Sens.*, vol. 24, no. 4, pp. 703–710, 2003.
- [19] D. Ghiglia and M. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software*. New York: Wiley, 1998.
- [20] T. R. Judge and P. J. Bryanston-Cross, "A review of phase unwrapping techniques in fringe analysis," *Opt. Lasers Eng.*, vol. 21, no. 4, pp. 199–239, 1994.
- [21] W.-S. Li and X.-Y. Su, "Phase unwrapping algorithm based on phase fitting reliability in structured light projection," *Opt. Eng.*, vol. 41, no. 6, pp. 1365–1372, 2002.
- [22] X. Su and Q. Zhang, "Phase unwrapping in the dynamic 3D measurement," in *Proc. AIP Conf.*, 2010, vol. 1236, pp. 467–471, 1.
- [23] D. Bergmann, *New Approach for Automatic Surface Reconstruction with Coded Light*, T. F. Schenk, Ed. Philadelphia, PA: SPIE, 1995, vol. 2572, pp. 2–9.
- [24] J. M. Huntley and H. Saldner, "Temporal phase-unwrapping algorithm for automated interferogram analysis," *Appl. Opt.*, vol. 32, no. 17, pp. 3047–3052, Jun. 1993.
- [25] K. Liu, Y. Wang, D. L. Lau, Q. Hao, and L. G. Hassebrook, "Dual-frequency pattern scheme for high-speed 3-D shape measurement," *Opt. Express*, vol. 18, no. 5, pp. 5229–5244, Mar. 2010.
- [26] X. Han and P. Huang, *Combined Stereovision and Phase Shifting Method: A New Approach for 3D Shape Measurement*, P. H. Lehmann, Ed. Philadelphia, PA: SPIE, 2009, vol. 7389, p. 73893C.
- [27] T. Weise, B. Leibe, and L. Van Gool, "Fast 3D scanning with automatic motion compensation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. CVPR '07.*, Jun. 2007, pp. 1–8.
- [28] J. M. Huntley, "Three-dimensional noise-immune phase unwrapping algorithm," *Appl. Opt.*, vol. 40, no. 23, pp. 3901–3908, Aug. 2001.
- [29] R. Cusack and N. Papadakis, "New robust 3-D phase unwrapping algorithms: Application to magnetic field mapping and undistorting echoplanar images," *NeuroImage*, vol. 16, no. 3, pt. A, pp. 754–764, 2002.
- [30] M. Costantini, F. Malvarosa, F. Minati, L. Pietranera, and G. Milillo, "A three-dimensional phase unwrapping algorithm for processing of multitemporal SAR interferometric measurements," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS '02*, Jun. 2002, vol. 3, pp. 1741–1743.
- [31] M. Jenkinson, "Fast, automated, N-dimensional phase-unwrapping algorithm," *Magn. Reson. Med.*, vol. 49, no. 1, pp. 193–197, 2003.
- [32] H. Abdul-Rahman, M. Gdeisat, D. Burton, and M. Lalor, W. Osten, C. Gorecki, and E. L. Novak, Eds., "Fast three-dimensional phase-unwrapping algorithm based on sorting by reliability following a non-continuous path," in *Proc. Soc. Photo-Optical Instrum. Eng. (SPIE) Conf. Series*, Jun. 2005, vol. 5856, pp. 32–40.
- [33] N. Saad and S. Peled, "Easy 3D phase unwrapping," *Proc. Int. Soc. Magn. Reson. Med.*, vol. 13, pp. 2251–2251, 2005.
- [34] F. J. van Leijen, R. F. Hanssen, P. S. Marinkovic, and B. M. Kampes, "Spatio-temporal phase unwrapping using integer least-squares," in *Proc. Fringe 2005 Workshop*, Feb. 2006, vol. 610, ESA Special Publication.
- [35] A. Hooper and H. A. Zebker, "Phase unwrapping in three dimensions with application to InSAR time series," *J. Opt. Soc. Amer. A*, vol. 24, no. 9, pp. 2737–2747, Sep. 2007.
- [36] O. Marklund, J. M. Huntley, and R. Cusack, "Robust unwrapping algorithm for three-dimensional phase volumes of arbitrary shape containing knotted phase singularity loops," *Opt. Eng.*, vol. 46, no. 8, p. 085601, 2007.
- [37] H. S. Abdul-Rahman, M. A. Gdeisat, D. R. Burton, M. J. Lalor, F. Lilley, and A. Abid, "Three-dimensional Fourier fringe analysis," *Opt. Lasers in Eng.*, vol. 46, no. 6, pp. 446–455, 2008.
- [38] M. Gdeisat, M. Arevalillo-Herráez, D. Burton, and F. Lilley, "Three-dimensional phase unwrapping using the Hungarian algorithm," *Opt. Lett.*, vol. 34, no. 19, pp. 2994–2996, Oct. 2009.
- [39] H. Abdul-Rahman, M. Arevalillo-Herráez, M. Gdeisat, D. Burton, M. Lalor, F. Lilley, C. Moore, D. Sheltraw, and M. Qudeisat, "Robust three-dimensional best-path phase-unwrapping algorithm that avoids singularity loops," *Appl. Opt.*, vol. 48, no. 23, pp. 4582–4596, Aug. 2009.
- [40] J. Langley and Q. Zhao, "A model-based 3D phase unwrapping algorithm using Gegenbauer polynomials," *Phys. Med. Biol.*, vol. 54, no. 17, p. 5237, 2009.
- [41] M. Arevalillo-Herráez, M. A. Gdeisat, and D. R. Burton, "Hybrid robust and fast algorithm for three-dimensional phase unwrapping," *Appl. Opt.*, vol. 48, no. 32, pp. 6313–6323, Nov. 2009.
- [42] F. M. M. Constantini and F. Minati, "A general formulation for robust and efficient integration of finite differences and phase unwrapping on sparse multidimensional domains," in *Proc. ESA Fringe Workshop*, Nov. – Dec. 2009.
- [43] S. S. Gorthi and P. Rastogi, "Fringe projection techniques: Whither we are?," *Opt. Lasers in Eng.*, vol. 48, pp. 133–140, Feb. 2010.
- [44] M. Arevalillo-Herráez, D. R. Burton, and M. J. Lalor, "Clustering-based robust three-dimensional phase unwrapping algorithm," *Appl. Opt.*, vol. 49, no. 10, pp. 1780–1788, Apr. 2010.
- [45] M. Costantini, F. Malvarosa, and F. Minati, "A novel approach for redundant integration of finite differences and phase unwrapping on a sparse multidimensional domain," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2010, pp. 1565–1568.
- [46] A. P. Shanker and H. Zebker, "Edgelist phase unwrapping algorithm for time series InSAR analysis," *J. Opt. Soc. Amer. A*, vol. 27, no. 3, pp. 605–612, Mar. 2010.
- [47] J. Bouquet, Camera Calibration Toolbox for Matlab, [Online]. Available: http://www.vision.caltech.edu/bouquet/calib_doc/
- [48] P. Felzenszwalb and D. Huttenlocher, "Efficient belief propagation for early vision," *Int. J. Comput. Vis.*, vol. 70, pp. 41–54, 2006.
- [49] B. J. Frey, R. Koetter, and N. Petrovic, "Very loopy belief propagation for unwrapping phase images," in *Advances in Neural Information Processing System*. Cambridge, MA: MIT Press, 2001, pp. 737–743.

- [50] K. Murphy, Y. Weiss, and M. I. Jordan, "Loopy Belief Propagation for Approximate Inference: An Empirical Study," in *Proc. 15th Conf. Uncertainty in Artif. Intell.*, 1999, pp. 467–475.
- [51] P. S. Huang, C. Zhang, and F.-P. Chiang, "High-speed 3-D shape measurement based on digital fringe projection," *Opti. Eng.*, vol. 42, no. 1, pp. 163–168, 2003.
- [52] [Online]. Available: http://www-video.eecs.berkeley.edu/research/4D_SL/viewpointConsistentUnwrap.wmv
- [53] [Online]. Available: http://www-video.eecs.berkeley.edu/research/4D_SL/temporalConsistentUnwrap.wmv



Ricardo R. Garcia (S'12) received the B.S. degree in electrical engineering from the University of Texas at Austin in 2007 and the M.S. degree in electrical engineering from the University of California (UC), Berkeley, in 2009. He is currently pursuing the Ph.D. degree at UC Berkeley.

His research interests are in computer vision with an emphasis in structured light systems and dynamic geometry capture.



Avidesh Zakhor (M'87–SM'01–F'02) received the B.S. degree from the California Institute of Technology, Pasadena, in 1983 and the S.M. and Ph.D. degrees from Massachusetts Institute of Technology, Cambridge, 1985, and 1987, respectively, all in electrical engineering.

She joined the faculty at the University of California (UC) Berkeley in 1988 where she currently holds the Qualcomm Professor of Electrical Engineering and Computer Sciences. Her areas of interest include theories and applications of signal, image and video processing, 3-D computer vision, and multimedia networking. She holds six U.S. patents and is the coauthor of three books with her students. She cofounded OPC Technology in 1996, which was later acquired by Mentor Graphics (Nasdaq: MENT) in 1998, and UrbanScan Inc. in 2005 which was acquired by Google in 2007.

Prof. Zakhor won a number of best paper awards, including the IEEE Signal Processing Society in 1997 and 2009, IEEE Circuits and Systems Society in 1997 and 1999, IEEE Solid Circuits Society in 2008, IEEE international conference on image processing in 1999, Packet Video Workshop in 2002, and IEEE Workshop on Multimodal Sentient Computing in 2007. She was a General Motors scholar from 1982 to 1983, was a Hertz fellow from 1984 to 1988, received the Presidential Young Investigators (PYI) award, and Office of Naval Research (ONR) young investigator award in 1992. In 2001, she was elected as IEEE fellow and received the Okawa Prize in 2004.