Fast 3D Scanning with Automatic Motion Compensation

Thibaut Weise, Bastian Leibe and Luc Van Gool
Swiss Federal Institute of Technology (ETH Zürich)
Zürich, Switzerland
{weise,leibe,vangool}@vision.ee.ethz.ch

Abstract

We present a novel 3D scanning system combining stereo and active illumination based on phase-shift for robust and accurate scene reconstruction. Stereo overcomes the traditional phase discontinuity problem and allows for the reconstruction of complex scenes containing multiple objects. Due to the sequential recording of three patterns, motion will introduce artifacts in the reconstruction. We develop a closed-form expression for the motion error in order to apply motion compensation on a pixel level. The resulting scanning system can capture accurate depth maps of complex dynamic scenes at 17 fps and can cope with both rigid and deformable objects.

1. Introduction

The development of non-contact 3D sensors has made considerable progress over the past decades [2]. Laser point and slit scanners allow very accurate reconstructions but are unable to capture dense dynamic scenes [3]. Time-of-flight systems overcome this problem, but low resolution and high costs are still major drawbacks [13]. Structured-light systems are generally low-cost, and many such systems with different characteristics have been developed in the past [1]. In contrast to the above methods, multi-camera systems work with normal ambient light and reconstruct the 3D geometry using triangulation between cameras [15, 4].

Our goal is to create a robust real-time 3D scanning system which creates highly accurate dense reconstructions including surface color, without posing restrictions on surface texture, scene complexity and especially object motion and deformation. To this end, we propose a *stereo phase-shift method*, based on a combination of structured light and stereo, and thereby joining their benefits: it is more accurate than passive stereo because of active projection using phase-shift [18], but it is also robust to discontinuities due to stereo. The system provides denser reconstructions than laser scanners or time-of-flight systems and it is real-time due to implementation on the GPU.

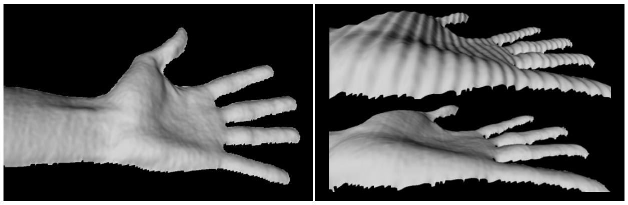


Figure 1. 3D reconstructions of a static (left) and a moving (right) hand. Motion compensation (bottom right) removes the ripples from the reconstructed surface (top right).

The phase-shift method requires three images recorded in quick succession. The inter-frame delay can be minimized by using the RGB channels of the projector for the three phases [22] and capturing the image with a high-speed monochrome camera. A color camera simultaneously captures the texture image with a longer exposure time. Despite the short acquisition period, motion will however lead to distortion artifacts. This is an intrinsic problem of phase-shift, since 3D acquisition is spread over multiple frames. When the scanned object is moving, it violates the basic phase-shift assumption that corresponding pixels in the three phase images depict the same surface point. This has only a small effect for tangential motion, since the resulting surface depth will change only slightly at the observed position in the camera. However, it leads to visible artifacts for motion directed along the surface normal (see Fig. 1). We address this problem by compensating for the motion artifacts and simultaneously estimating the underlying motion. The resulting reconstruction can also be seen in Fig. 1.

Our paper contains the following main contributions:
1) We present a low-cost, real-time 3D scanning system that operates by stereo phase-shift. It is composed of an off-the-shelf video projector with its color wheel removed to project three separate phase images per frame; two synchronized monochrome cameras to capture the phase images; and a third color camera camera operating at a longer exposure setting which simultaneously obtains a colored texture map of the reconstructed scene. Contrary to previous systems, our system performs phase unwrapping based on dense stereo methods, hence the need for the second monochrome camera. This makes it easily scalable to multiple scene objects. The whole reconstruction pipeline is efficiently implemented on the GPU.

2) We analytically derive a closed-form expression for the motion error incurred by this system. From this, we devise a numerical approximation of the error that can be efficiently estimated from the original 3-phase images in order to compensate for a large part of its effects. Simultaneously, motion along the surface normal is estimated, which could further be used to enhance tracking or registration algorithms.
3) The resulting system can capture high-quality textured depth maps at interactive frame rates of 17 fps, even when the scanned objects are moving rigidly, non-rigidly, or when they are being deformed. Interest in fast scanning and scanning dynamic scenes go hand in hand. However, for phase-shift the latter remains elusive without motion compensation as proposed here.

The paper is organized as follows. In Section 2 we discuss related work. Section 3 proposes the stereo phase-shift method for robust scanning. In Section 4, we analyze the motion artifacts and describe a method for efficient compensation. Section 5 presents quantitative and qualitative results of the scanning system.

2. Related Work

3D reconstruction using stereo cameras is a popular research topic, and many algorithms have been developed in the past. A comprehensive overview can be found at [15]. The same authors also provide a benchmark on the Web [16] where developers can test and compare algorithms. The current top performer [10] relies on color segmentation and optimization on the segment level.

Structured light replaces one camera by a projector which illuminates the scene with one or more patterns [1, 14]. Phase-shift is one of the popular structured light techniques, as it allows for highly accurate and dense reconstructions [18, 22, 17]. The authors of [20] and [5] present recent systems combining structured light and stereo into a common framework.

Dynamic scenes are often handled by using one-shot techniques [11, 19]. However, highly accurate dense reconstructions that do not pose restrictions on texture generally require multiple frames. Motion then needs to be taken into account explicitly. [20, 21] use oriented spacetime windows to deal with motion. They provide high quality reconstructions, but are not suitable for real-time applications. [3] use a laser scanner and explicitly assume motion between each sample. Motion and model are recursively estimated and updated. [9] track stripe boundaries in the image to allow slow motion.

For phase-shift based methods, [12] mention the problem as (observed) sine wave modulation. They correct for it by estimating lines from the projected sinusoidal patterns and calculating motion as line translation. However, this method is not suitable for high-speed acquisition and can only compensate for rigid lateral motion. [23] reduce the motion error by using a phase-shift with three patterns, where the last pattern is flat. Thus, motion will still introduce distortions, though to a lesser extent, as only motion between the first two patterns is problematic. Still, the authors concede that this approach is not suited for fast motion, such as speaking.

3. Stereo Phase-shift

Our proposed scanning system is a combination of sinusoidal phase-shift structured light and stereo matching. It combines the benefits of the two methods, namely maximum accuracy through phase-shift and robustness to discontinuities through stereo. High-speed performance is achieved by implementing most of the algorithms on the GPU (currently NVidia GeForce 7900 GTX). Note that implementation details concerning fragment and vertex shaders on the GPU are omitted.

Hardware Setup. Our system is inspired by the phase-shift acquisition system by [22], but extends it with a second monochrome camera. It consists of a DLP projector, two high-speed monochrome cameras and a texture camera. The 4-segment color wheel (RGBW) of the projector has been removed, so that it projects three independent monochrome images at 120 Hz, which are sent to the projector as the R, G and B color channel. The two monochrome cameras are synchronized and record the three images. The delay between two image acquisitions is currently 4 ms, where each exposure takes 2 ms, resulting in a total recording time of 14 ms for the three phase-shifted images (2+4+2+4+2 ms). The texture camera is also synchronized, but uses a longer exposure to integrate over all three projected images.

3.1. Phase-Shift

Phase-shift is a well-known fringe projection method for the retrieval of 3D information. A set of phase-shifted sinusoidal patterns is projected, and the phase is calculated at each point. The minimum number of images is three, but more images will improve the accuracy of the reconstructed phase. However, we use three images, as this allows the images to be projected at high speed using the modified DLP.

The intensities for each pixel (x, y) of the three images can be described by the following formulas assuming a linear projector, a linear camera, constant lighting, and a static object during the 14 ms recording interval ((x, y) is omitted for the sake of brevity):

$$\begin{aligned} I_r &= I_{dc} + I_{mod} \cos(\phi - \theta) \\ I_g &= I_{dc} + I_{mod} \cos(\phi) \\ I_b &= I_{dc} + I_{mod} \cos(\phi + \theta) \end{aligned} \tag{1}$$

where I_r, I_g and I_b are the recorded intensities, I_{dc} is the DC component, I_{mod} is half the signal amplitude, ϕ is the phase, and θ the constant phase-shift. The phase ϕ corresponds to projector coordinates computed as

$$\phi = \frac{x_p}{w} 2\pi N, \qquad (2)$$

where x_p is the projector x coordinate, w the horizontal resolution of the projection pattern, and N the number of periods of the sinusoidal pattern. This means that if phase ϕ is known, the depth d can be calculated using point-surface triangulation between camera and projector. The wrapped phase ϕ' (0,2π) can be calculated as follows

$$\phi' = \arctan\left(\tan\left(\frac{\theta}{2}\right) \frac{(I_r - I_b)}{(2\, I_g - I_r - I_b)}\right). \qquad (3)$$

In our system, we use a shift offset of $\theta = \frac{2\pi}{3}$ which gives the following final expressions for phase, DC component, and amplitude:

$$\begin{aligned} \phi' &= \arctan\left(\frac{\sqrt{3}\,(I_r - I_b)}{(2\, I_g - I_r - I_b)}\right), \\ I_{dc} &= \frac{I_r + I_b + I_g}{3}, \qquad (4) \\ I_{mod} &= \sqrt{\frac{(I_b - I_r)^2}{3} + \frac{(2I_g - I_r - I_b)^2}{9}}. \end{aligned}$$

A mask is calculated that removes all uncertain phase values from the image. The masking operation takes into account saturation, signal strength, and phase derivative variance.

3.2. Stereo Unwrapping

Two-dimensional phase unwrapping is the process that converts the wrapped phase $\phi'(x, y)$ to the absolute phase

$$\phi(x, y) = \phi'(x, y) + 2\pi k(x, y), \quad k(x, y) \in [0, N-1] \qquad (5)$$

where $k(x, y)$ represents the period, and N is the number of projected periods. It solves the inherent ambiguity of the phase calculation. Different methods for unwrapping have been proposed in the past. In general these methods unwrap the phase by integrating along reliable paths in the image or by taking a global minimum-norm approach [8]. One exception is [7] that uses belief propagation for unwrapping on the pixel level.

There are two major problems with such unwrapping algorithms: Firstly, the unwrapping methods only provide a relative unwrapping and do not solve for the absolute phase. Secondly, if two surfaces are scanned that have a discontinuity of more than 2π for all contact points, then no method based on unwrapping will correctly unwrap these two surfaces relative to each other. For 3D scanning this ambiguity has often been addressed using a combination of gray-code light and phase-shift [17]. However, this requires more recorded images, and does not lend itself to dynamic environments. We propose a stereo-based alternative, which does not pose this problem.

Stereo Matching. As mentioned above only the wrapped phase ϕ' can be calculated at each image pixel. Eq. (5) shows that for each pixel N possibilities exists. This means that the recorded phase can originate from exactly N different positions in the projector image (see eq. (2)), and thus N possible 3D positions.

For each pixel $p = (x, y)$ we can solve for the period $k(p)$ using stereo matching between the two cameras. In contrast to traditional dense stereo matching, the number of possibilities is limited to N, which allows for a very fast implementation on the GPU. For each possible period k, the 3D position is calculated using point-surface triangulation between the first camera and the projector. Note that both projector and cameras are calibrated internally as well as externally. The resulting 3D point is projected into the second camera, and the correlation value $d(k, p)$ is calculated as pixel-based sum-of-squared-differences (SSD) on the three phase images:

$$d\,(k, p) = SSD\,(I_1\,(p)\,, I_2\,(k, p)) \qquad (6)$$

For an efficient implementation on the GPU, the 3D position calculation can be omitted by using the trifocal tensor. This allows to calculate the coordinates in the second camera directly from the coordinates in the first camera and the projector x coordinate.

The period $k(p)$ with maximum correlation (minimum SSD) is assumed to be the true period of the recorded phase. This is in general a valid assumption, as the probability is low that a random pixel in the second camera will record a surface point with the same surface reflectance and a similar wrapped phase but different period. However, noise, specularities, and occlusions increase the probability that a wrong $k(p)$ is chosen. All further algorithmic steps are therefore aimed at decreasing the number of false $k(p)$ in the image.

In the following description, a continuous surface segment according to our definition is a segment of pixels in the absolute phase image, such that the absolute phase within that segment is continuous. This means that no two neighboring pixels have an absolute phase difference of more than π (similar to the continuity assumption in two-dimensional phase unwrapping). The number of segments is reduced by a series of morphological operations to mask out pixels close to phase discontinuities. Thus, small segments are removed and then incorporated into larger surface segments using local two-dimensional phase unwrapping.

Unwrap Optimization. Let $s_j \in S$ be a surface segment consisting of a set of pixels $p_i \in s_j$. Every pixel p_i has an assigned period $k_a(p_i)$ which may or may not correspond to the true period $k_t(p_i)$. As all pixels in s_j are assumed to be connected, the period error $o_j = k_a(p_i) - k_t(p_i)$ will be constant for all pixels of that segment. This allows to optimize for a period offset o_j at segment instead of pixel level. The optimization attempts to find period offsets o_j for all segments, such that their connectivity is maximized. This is a labeling problem, which we formulate in an energy minimization framework and solve using Loopy Belief Propagation [6, 10].

A labeling f assigns to each segment s_j a (corrective) period offset $o_j = f(s_j)$, so that $k_{new}(p_i) = k_a(p_i) - o_j$. The energy for a labeling f is given by:

$$E(f) = E_{data}(f) + \beta E_{smooth}(f) \tag{7}$$

where

$$E_{data}(f) = \sum_{s_j \in s} \sum_{p_i \in s_j} min\left(\alpha, d\left(k_a(p_i) - f\left(s_j\right), p_i\right)\right) \tag{8}$$

and

$$E_{smooth}(f) = \sum_{\forall s_i, s_j \in S | f(s_i) - f(s_j) \neq o(s_i, s_j)} b\left(s_i, s_j\right)\left(1 - disc\left(s_i, s_j\right)\right) \tag{9}$$

α is a threshold parameter to ensure that occlusions are not too strongly penalized. In our GPU implementation, only the two smallest SSD candidates $d(k, p_i)$ (eq. (6)) are retained for each pixel, making α the default weight for all other assignments. β is a scale parameter that weighs the importance of smoothness between segments; $b\left(s_i, s_j\right)$ is the border length between two segments; $disc\left(s_i, s_j\right) = \max\left[1, w_1\left(I_x^2 + I_y^2\right) + w_2\left(\phi_{xx}'^2 + \phi_{yy}'^2\right)\right]$ is a discontinuity weight based on a weighted combination of the squared intensity gradient magnitude and the phase Laplacian along the segment border. Thus, it favors discontinuities to coincide with texture and phase borders. $o(s_i, s_j)$ is the required relative period offset between the segments so that they are connected. This can be calculated from the border pixels. The labeling f with minimum energy is then used to update the period of every pixel p_i:

$$k_{new}(p_i) = k_a(p_i) - f(s_j) \tag{10}$$

Consistency Check. After the period assignment optimization, there will still be errors in the phase map. We therefore perform a left-right consistency check to reduce the number of outliers, thus improving reconstruction accuracy. However, this comes at the price of increased run-time (since the algorithm has to run twice) and may remove pixels that are correctly assigned in one camera, but occluded in the other.

3.3. Texture

Texture acquisition is achieved using a dedicated color camera. Eq. (5) shows that for a phase-shift of $\theta = \frac{2\pi}{3}$, the DC component I_{dc} will be the average of the three projected sinusoidal patterns. This averaging is done implicitly by setting the camera's exposure time to one 3-phase projection period. The cameras are calibrated, and thus, texturing a 3D point is a simple projection and texture look-up operation, which is handled efficiently on the GPU.

4. Motion

The phase-shift algorithm requires three images to calculate the phase. In order to be as insensitive to motion as possible, these three images need to be acquired in quick succession. Our proposed system has an interframe delay of 4+2 milliseconds. Despite this fast acquisition speed, motion will still distort the reconstructed phase and thus the reconstructed 3D geometry. An example can be seen in Fig. 2, where a planar surface and a hand have been moved towards the camera. The distortion of the reconstructed 3D geometry can be clearly perceived as ripples on the surface. For accurate scene reconstruction, this artifact needs to be compensated for. In the following section, we present a method that allows not only to compensate for the ripples, but also to estimate the motion of the reconstructed surface.

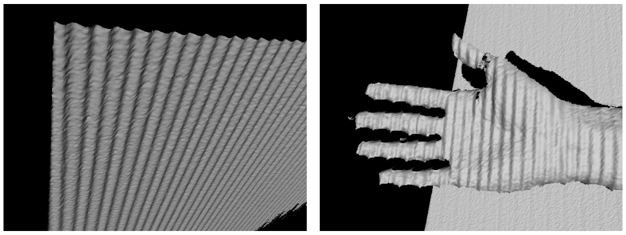


Figure 2. 3D reconstructions of a moving a) plane and b) hand. The motion artifacts can be seen as ripples on the surface.

Let us consider the motion of a planar surface and its effect on phase calculation, as displayed in Fig. 3. The true location that the camera observes at a certain pixel is P_0 at time t_0 and P_{-1} at t_{-1}. From our system we know that $\Delta t = t_0 - t_{-1} = t_1 - t_0 = 6ms$. If we know P_0 and P_{-1}, we can calculate the distance vector Δc, and thus, the normal motion displacement Δs as the projection of Δc onto the surface normal n. This allows the calculation of the velocity $\frac{\Delta s}{\Delta t}$ of the surface along its normal. This calculation is in fact tightly coupled with the problem of compensating the motion ripples.

Assume p_0, p_{-1}, and p_1 are the corresponding projector pixel coordinates of P_0, P_{-1}, and P_1, respectively. Only the x coordinates are of importance, as the projection pattern is invariant vertically. The difference between the points in the projection pattern is $\Delta x = p_{-1}^x - p_0^x \approx p_0^x - p_1^x$. As given in eq. (1), the intensity of an image pixel in each of the three images depends on the DC component I_{dc}, amplitude I_{mod}, phase ϕ, and shift offset θ. If we assume that the surface is planar, uniform, and diffuse, then I_{dc} and I_{mod} will be locally constant on the observed surface. The shift offset θ is constant at $\frac{2\pi}{3}$. However, ϕ changes between the three images as the camera observes the surface at three different moments in time. At time t_{-1},t_0, and t_1 it observes the intensity as projected by p_{-1}, p_0, and p_1, respectively. By converting Δx into the phase difference $\Delta\theta$

$$\Delta\theta = 2\pi N \frac{\Delta x}{w} \tag{11}$$

(where w is the width of the projection pattern and N is the number of projected phase wraps), the truly observed intensities can be described as follows

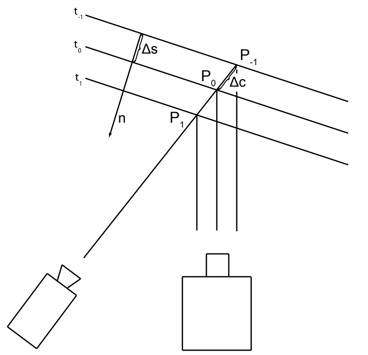


Figure 3. Diagrammatic view of a plane moving towards the camera and its resulting recordings at 3 timesteps.

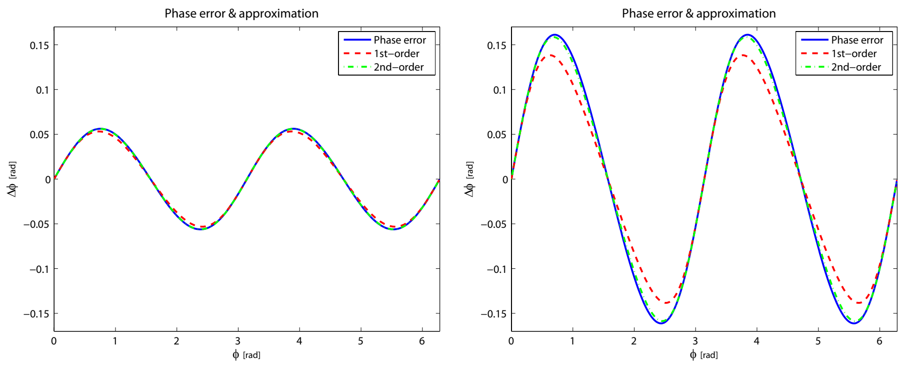


Figure 4. True phase error against first and second-order approximation for $\theta = \frac{2\pi}{3}$: $\Delta\theta_1 = 0.1$ (left), $\Delta\theta_2 = 0.3$ (right).

$$\begin{aligned} I_r &= I_1 + I_2 \, \cos\left(\phi - \theta + \Delta\theta\right) \\ I_g &= I_1 + I_2 \, \cos\left(\phi\right) \\ I_b &= I_1 + I_2 \, \cos\left(\phi + \theta - \Delta\theta\right). \end{aligned} \qquad (12)$$

As can be seen, these intensities are the same as if we had used a phase-shift with a shift offset $\theta - \Delta\theta$ instead of θ. It follows that if we can estimate $\Delta\theta$ for each pixel, then we will be able to calculate the true undistorted phase at each pixel, as well as estimate the surface motion.

The relative phase error $\Delta\phi$ between distorted phase ϕ_f and correct phase ϕ_t can be calculated as follows

$$\begin{aligned} \phi_t &= \arctan\left(\tan\left(\frac{\theta - \Delta\theta}{2}\right) g\right) \\ \phi_f &= \arctan\left(\tan\left(\frac{\theta}{2}\right) g\right) \\ \Delta\phi &= \phi_f - \phi_t \\ g &= \frac{(I_r - I_b)}{(2\, I_g - I_r - I_b)} \end{aligned} \qquad (13)$$

where g is the ratio as calculated with the true shift offset $\theta - \Delta\theta$. Fig. 4 shows the relative phase error when using a shift offset $\theta = \frac{2\pi}{3}$ for calculation instead of the motion-induced true shift offsets $\theta_1 = \frac{2\pi}{3} - 0.1$ and $\theta_2 = \frac{2\pi}{3} - 0.3$. The relative phase error would result in motion ripples if used for geometric reconstruction, similar to those in Fig. 2.

Using eqs. (13) we can find an expression for the distorted phase ϕ_f in terms of the true phase ϕ_t:

$$\phi_f = \arctan\left(\tan\left(\phi_t\right) \frac{\tan\left(\frac{\theta}{2}\right)}{\tan\left(\frac{\theta-\Delta\theta}{2}\right)}\right). \qquad (14)$$

Locally, the undistorted phase values can be approximated to evolve linearly along a scanline of the camera:

$$\phi_t(m) = \phi_c + \phi_m m \qquad (15)$$

where m is the x coordinate of the pixel. Thus,

$$\phi_f(m) = \arctan\left(\tan\left(\phi_c + \phi_m m\right) \frac{\tan\left(\frac{\theta}{2}\right)}{\tan\left(\frac{\theta-\Delta\theta}{2}\right)}\right) \qquad (16)$$

The unknowns in the above equation are ϕ_c, ϕ_m, and $\Delta\theta$. Instead of minimizing a non-linear function, we linearize the problem by reformulating ϕ_t as Taylor expansion of ϕ_f:

$$\begin{aligned} \phi_t = \phi_f \quad &+ \quad \frac{1}{1+\tan^2\left(\phi_f\right)} \left(\tan\left(\phi_t\right) - \tan\left(\phi_f\right)\right) \\ &- \quad \frac{\tan\left(\phi_f\right)}{\left(1+\tan^2\left(\phi_f\right)\right)^2} \left(\tan\left(\phi_t\right) - \tan\left(\phi_f\right)\right)^2 \\ &+ \quad O\left(\left(\tan\left(\phi_t\right) - \tan\left(\phi_f\right)\right)^3\right). \end{aligned} \qquad (17)$$

Rewriting eq. (14) as

$$\tan\left(\phi_t\right) = \frac{\tan\left(\frac{\theta-\Delta\theta}{2}\right)}{\tan\left(\frac{\theta}{2}\right)} \tan\left(\phi_f\right) \qquad (18)$$

and using the fact that

$$\frac{1}{1+\tan^2\left(\phi_f\right)} = \cos^2\left(\phi_f\right), \qquad (19)$$

we can express ϕ_t as

$$\phi_t = \phi_f + \sin\left(2\phi_f\right) y - \left(\frac{1}{2}\sin\left(2\phi_f\right) - \frac{1}{4}\sin\left(4\phi_f\right)\right) y^2 \qquad (20)$$

where

$$y = \frac{1}{2}\left(\frac{\tan\left(\frac{\theta-\Delta\theta}{2}\right)}{\tan\left(\frac{\theta}{2}\right)} - 1\right) \qquad (21)$$

$$\Delta\theta = \theta - 2\arctan\left(\tan\left(\frac{\theta}{2}\right)(2y+1)\right). \qquad (22)$$

If only the first-order term of the Taylor expansion is used, then a linear least-square fit can be performed in the local neighborhood of each pixel solving for ϕ_c,ϕ_m, and y:

$$\min_{\phi_c, \phi_m, y} \sum \left(\phi_f(m) - \left(\phi_c + m\phi_m - \sin\left(2\phi_f(m)\right) y\right)\right)^2 \qquad (23)$$

In case the second-order term is also incorporated in the minimization, the roots of a third-order polynomial need to be found.

Fig. 4 displays the true relative phase error against its first and second-order approximations for $\theta = \frac{2\pi}{3}$, where first $\Delta\theta_1 = 0.1$ and then $\Delta\theta_2 = 0.3$. It can be seen that both approximations are close to the true curve, but degrade with increasing $\Delta\theta$. The second-order curve is a better approximation and is less affected by $\Delta\theta$. The figure also shows that the quality of the approximation is not uniformly distributed: the curves coincide well for phase angles around 0 and π, but are worse for $[\frac{1}{4}\pi, \frac{3}{4}\pi]$ and $[\frac{5}{4}\pi, \frac{7}{4}\pi]$.

For large $\Delta\theta$, the first-order Taylor expansion degrades, and a bias is introduced in the estimation of y. Instead of using the second-order approximation, we propose a faster solution. We use a simulation that estimates y for different values of $\Delta\theta$ to create a lookup-table (LUT), which can then be used to retrieve the true $\Delta\theta$ from an estimated biased y.

5. Experimental Results

Our current scanning setup allows reconstructions in a frustum of 60 cm depth with a 40×30 cm^2 front end and a 60×50 cm^2 back end. Both stereo phase-shift and motion estimation are efficiently implemented on the GPU, though segment optimization runs on the CPU. Motion estimation uses a local 1D-neighborhood of seven pixels to estimate y for each pixel. Median filtering is applied for robustness. Then y is either used to update the phase ϕ_f, or the shift offset $\Delta\theta$ is retrieved from the LUT and the phase ϕ_f recalculated. The depth at each pixel is calculated using point-surface triangulation, and optionally 5×5 spatial Gaussian smoothing is applied to the resulting geometry.

5.1. Quantitative Evaluation

Experimental Setup. For quantitative evaluation, we measure the accuracy of the reconstruction of a 20×20 cm^2 planar surface. The surface is swept across the working volume at discrete steps, and for each step ten (static) reconstructions are performed.

Reconstruction Error. Fig. 5(a) displays the root mean square (RMS) reconstruction error for the planar surface, averaged over ten frames at a distance of 800 mm to 1200 mm. At each point, a plane is fitted to the 3D points, and the RMS of the residuals is calculated with or without motion compensation and smoothing. The figure shows that the reconstruction accuracy of the system is high with only 0.125 mm error at 1100 mm distance. It can further be seen that Gaussian smoothing improves the reconstruction considerably, while motion compensation does not degrade the reconstruction quality for these static scenes. Fig. 5(b) shows the magnified residuals for a fitted plane. Gray represents no error, white a positive residual, and black a negative residual. Three sources of error can be identified: First, a calibration error as shown by the slight distortion. Second, a periodical signal which is due to non-constant lighting of the projector and third, noise from the camera. Gaussian smoothing partially removes the third source of error from the geometry, which is significant at farther distances from the camera and projector. Closer to the camera, the two other sources of error dominate, as the system has been calibrated for a distance of about 1100 mm.

Motion Compensation. A second experiment evaluates the reconstruction accuracy during motion. The planar surface is moved at 0.5 mm intervals, and the resulting images are combined so as to simulate motion. Fig. 5(c) shows the results of this experiment. It can clearly be seen that motion compensation is indeed beneficial to remove the motion artifacts. For motion speeds up to $500\frac{mm}{s}$, our method can factor out the effects of motion entirely. At higher speeds, the first-order Taylor approximation is no longer sufficient, and motion artifacts become visible, although their strength is greatly reduced. Using the LUT further improves the results, as the bias of the estimated y is removed. At very high speeds ($\geq 1300\frac{mm}{s}$ and $\leq -900\frac{mm}{s}$), reconstruction is no longer possible, as other sources of error dominate.

The asymmetry of the reconstruction error can be explained by the fact that the scale of the phase modulation in eq. 21 is asymmetric with respect to $\Delta\theta$ (this relationship is shown in Fig. 6(a)). y increases more dramatically for negative $\Delta\theta$, and thus, motion away from the camera induces larger ripples than motion towards the camera.

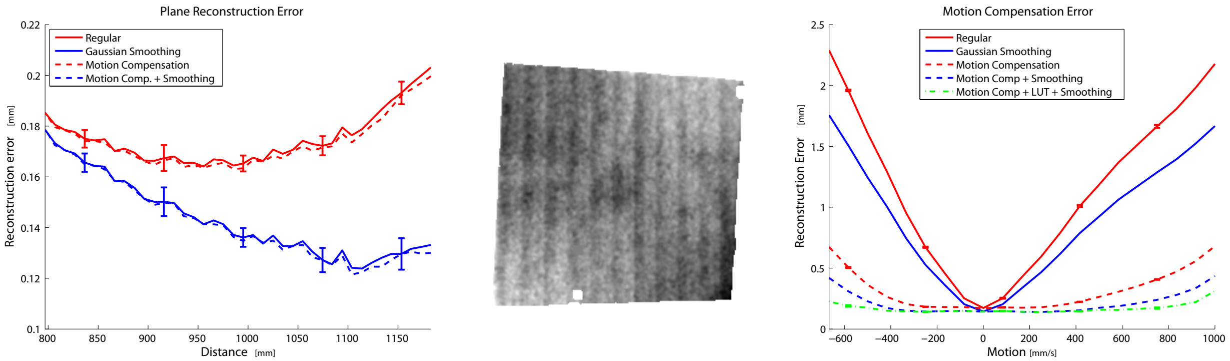
Effect of Accelerated Motion. Fig. 5(c) already contains small acceleration errors due to the restrictions of the measurement setup. The simulation in Fig. 6(b) shows that the phase error due to acceleration is similar to a biased and shifted version of constant motion. Motion compensation will partially remove the sinusoidal structure, but the bias will remain. However, as can be seen in Fig. 5(c), bias only has a small effect, and is not problematic in practice.

Accuracy of Motion Estimate. Fig. 6(c) displays the estimated motion of the plane against the true motion of the plane. The motion is estimated through the calculation of $\Delta\theta$ from the estimated phase modulation y. For the static scene, the motion estimate is less than $1\frac{mm}{s}$; for $500\frac{mm}{s}$ the estimate is approximately $700\frac{mm}{s}$. The motion estimate is slightly biased because of the nature of the Taylor expansion. The advantage of the LUT can clearly be seen, as it removes the bias and estimates almost exactly the true motion: for $500\frac{mm}{s}$ it estimates $495\frac{mm}{s}$.

5.2. Qualitative Results

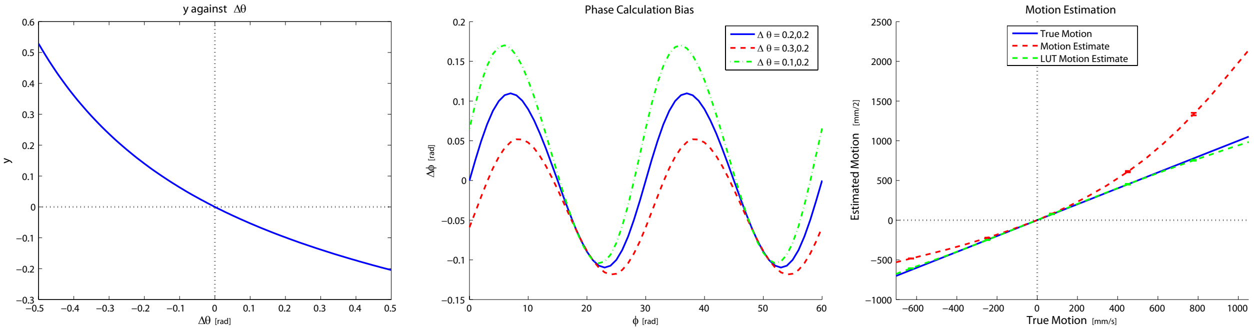
Fig. 7 displays the reconstruction of a complex scene. The result shows that the scanning system can robustly reconstruct multiple objects with high accuracy. The geometry is correctly textured using the color camera. Conventional two-dimensional unwrapping would fail, as the phase discontinuity between fruit and teapot is greater than 2π.

The next examples demonstrate that deformable objects are handled correctly by our system, although the planarity assumption does not hold. Fig. 8 shows a waving cloth. It is reconstructed at high accuracy, and motion ripples are removed from the surface. Fig 9 displays the motion compensated reconstruction of a person speaking. Almost no motion artifacts are visible, although the person moves his



(a) (b) (c)

Figure 5. a) Reconstruction error against distance to camera and projector. b) Residuals of fitted plane. The remaining error is caused by distortion, projector non-linearity and noise. c) Reconstruction error against surface motion where positive motion is towards the camera.



(a) (b) (c)

Figure 6. a) Phase modulation y against shift offset $\Delta\theta$. Negative $\Delta\theta$ induces much stronger phase modulation and thus ripples in the geometry. b) Simulation of phase error due to accelerated motion. Acceleration is equivalent to two different shift offsets in the first and third pattern. c) Estimated motion against true motion.

head. The reconstruction of an upper body with moving hands can be seen in Fig. 10. Again, motion ripples are correctly compensated for. Fig. 11 shows online reconstructions of hand gestures as might be used for gesture recognition. All scenes can be seen on the accompanying video.

6. Conclusion & Future Work

We have presented a robust real-time 3D scanner that reconstructs complex scenes of several independently moving objects at 17 frames per second. The scanner relies on the phase-shift method which allows accurate reconstructions at high speed. Our system overcomes the two major problems of fast phase-shift scanning, namely discontinuities and motion artifacts. A stereo phase-shift method is presented that uses correlation-based stereo with optimization on the segment level to overcome the discontinuity problem. An analysis of the motion error is used to compensate for motion artifacts on the pixel level. Simultaneously, motion along the surface normal is estimated. High speed is achieved by implementing all methods on the GPU. High-frequency texture and object discontinuities still pose problems during motion, as the assumption of invariant surface reflectance is violated. Future work will investigate the possibility to add a stereo module which handles these cases.

Acknowledgments. This research has been funded in parts by EU projects ImmerSence (IST-2006-027141) and CHIRON (MEST-CT-2004-514539).

References

[1] J. Batlle, E. Mouaddib, and J. Salvi. Recent progress in coded structured light as a technique to solve the correspondence problem: A survey. *PR*, 31:963–982, 1998.

[2] F. Blais. Review of 20 years of range sensor development. In *Videometric VII, Proc. of SPIE EI*, pages 62–76, 2003.

[3] F. Blais, M. Picard, and G. Godin. Accurate 3D acquisition of freely moving objects. In *3DPVT*, 2004.

[4] M. Z. Brown, D. Burschka, and G. D. Hager. Advances in computational stereo. *PAMI*, 25(8):993–1008, 2003.

[5] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *PAMI*, 27(2):296–302, 2005.

[6] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *IJCV*, 70(1):41–54, 2006.

[7] B. J. Frey, R. Koetter, and N. Petrovic. Very loopy belief propagation for unwrapping phase images. In *NIPS*, 2001.

[8] D. C. Ghiglia and M. D. Pritt. *Two-Dimensional Phase Unwrapping: Theory, Algorithms and Software*. Wiley-Interscience, 1998.

[9] O. A. Hall-Holt and S. Rusinkiewicz. Stripe boundary codes for real-time structured-light range scanning of moving objects. In *ICCV*, pages 359–366, 2001.

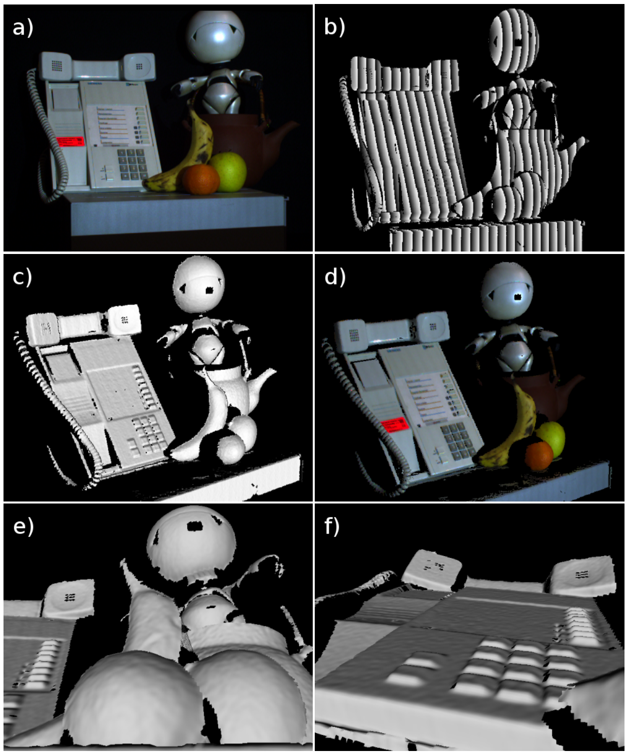


Figure 7. Reconstruction of a complex scene containing several objects (phone, teapot, figure, fruit): a) texture image, b) reconstructed phase, c) geometry, d) textured geometry, e)+f) close-ups

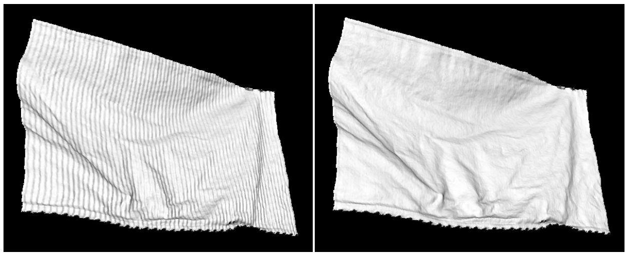


Figure 8. Reconstruction of a waving cloth. Motion correction correctly removes the ripples (right).

[10] A. Klaus, M. Sormann, and K. F. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *ICPR*, pages 15–18, 2006.

[11] T. P. Koninckx, A. Griesser, and L. Van Gool. Real-time range scanning of deformable surfaces by adaptively coded structured light. In *3DIM*, pages 293–301, 2003.

[12] T. P. Koninckx, P. Peers, P. Dutre, and L. Van Gool. Scene-adapted structured light. In *CVPR*, pages 611–618, 2005.

[13] R. Lange, P. Seitz, A. Biber, and R. Schwarte. Time-of-flight range imaging with a custom solid state image sensor. In *Laser Metrology and Inspection*, pages 180–191, 1999.

[14] J. Salvi, J. Pagès, and J. Batlle. Pattern codification strategies in structured light systems. *PR*, 37(4):827–849, 2004.

[15] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1-3):7–42, 2002.

[16] http://cat.middlebury.edu/stereo/.

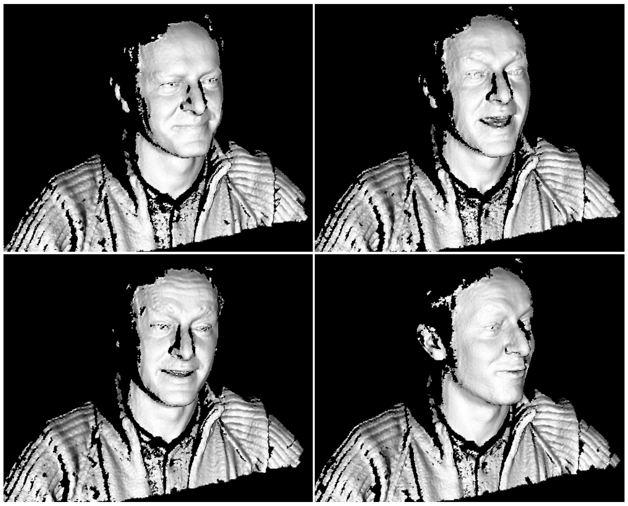


Figure 9. Reconstruction of a person speaking.

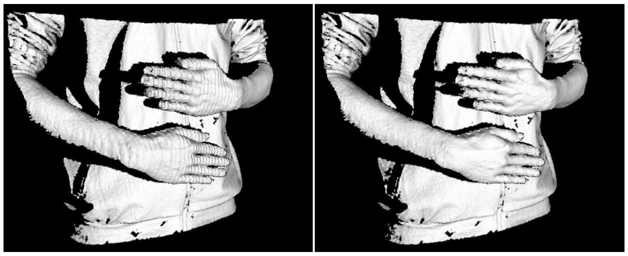


Figure 10. Reconstruction of moving hands in front of the torso. On the right with motion compensation.

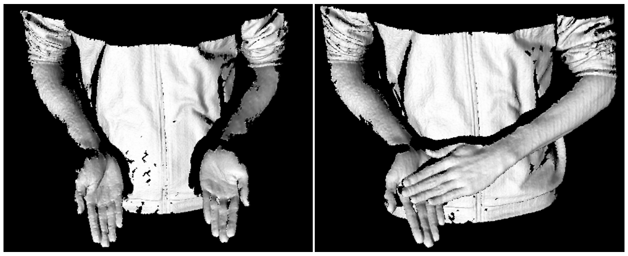


Figure 11. Online reconstruction of hand gestures.

[17] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. *CVPR*, 01:195–202, 2003.

[18] C. Wust and D. W. Capson. Surface profile measurement using color fringe projection. *MVA*, 4:193–203, 1991.

[19] L. Zhang, B. Curless, and S. M. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *3DPVT*, pages 24–36, 2002.

[20] L. Zhang, B. Curless, and S. M. Seitz. Spacetime stereo: Shape recovery for dynamic scenes. In *CVPR*, 2003.

[21] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: High-resolution capture for modeling and animation. In *ACM Annual Conf. on Comp. Graphics*, 2004.

[22] S. Zhang and P. Huang. High-resolution, real-time 3d shape acquisition. In *CVPR Workshop*, pages 28–28, 2004.

[23] S. Zhang, D. Royer, and S.-T. Yau. Gpu-assisted high-resolution, real-time 3-d shape measurement. *Optics Express*, 14(20):9120–9129, 2006.