

## Subpixel Scanning Invariant to Indirect Lighting using Quadratic Code Length

Nicolas Martin  
Université de Montréal

`martinic@iro.umontreal.ca`

Vincent Couture  
Université de Montréal

`chapdelv@iro.umontreal.ca`

Sébastien Roy  
Université de Montréal

`roys@iro.umontreal.ca`

### Abstract

We present a scanning method that recovers dense subpixel camera-projector correspondence without requiring any photometric calibration nor preliminary knowledge of their relative geometry. Subpixel accuracy is achieved by considering several zero-crossings defined by the difference between pairs of unstructured patterns. We use gray-level band-pass white noise patterns that increase robustness to indirect lighting and scene discontinuities. Simulated and experimental results show that our method recovers scene geometry with high subpixel precision, and that it can handle many challenges of active reconstruction systems. We compare our results to state of the art methods such as micro phase shifting and modulated phase shifting.

### 1. Introduction

Active scanning approaches using a camera and a projector have gained popularity in various 3D scene reconstruction systems [15, 14]. One or many known patterns are projected onto a scene, and a camera observes the deformation of these patterns to calculate surface information. Camera-projector correspondence is achieved by identifying each projector pixel by a code defined by the projected patterns.

The resolution of a projector being finite, several methods attempt to recover subpixel correspondences, thus giving better reconstruction results. In practice, it is often the case that a camera pixel observes a mixture of intensities from two or more projector pixels. The camera pixel integrates their intensities reflected from the scene, and the problem is then to estimate the composition of the measured intensity.

The main contribution of this paper is to present a method that recovers very high precision subpixel correspondence and is robust to indirect illumination. Our method uses a sequence of gray level band-pass white noise patterns to encode each projector pixel uniquely [4]. These

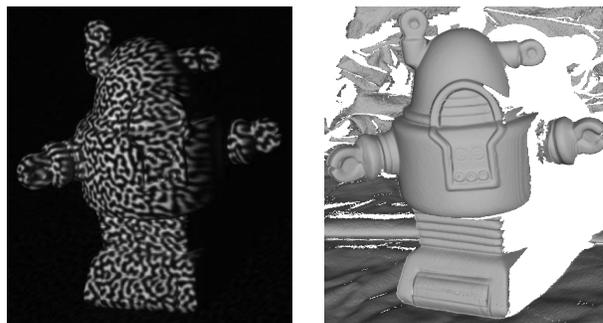


Figure 1. A band-pass gray level pattern projected on a scene (left) and its 3D reconstruction using our method (right).

are called *unstructured* patterns because the codes do not represent projector pixel position directly and a search is required to find the best correspondence for each camera pixel [12, 5, 17, 4]. This approach is robust to challenging difficulties in active systems such as indirect illumination and scene discontinuities. Our method yields the same robustness as [4] while using a lot less patterns. Besides, it produces dense subpixel correspondence whereas the original method did not.

The key to achieving both subpixel correspondence and reducing the number of patterns is to increase the length of the code generated from the patterns. Instead of using the signed differences between each pattern and a reference as in [4], we consider differences between all possible pairs of blurred gray level unstructured patterns. The resulting codes are much longer than the number of patterns albeit with some redundancy. Every sign change between neighboring projector pixels provides a zero-crossing which is used as a constraint to recover subpixel correspondence. An example of our patterns is shown in Fig. 1 along with a 3D reconstruction.

The method we propose uses two-dimensional patterns and is designed to avoid the need for geometric or photometric calibration of both the camera and the projector. While our method could rely on epipolar geometry to allow using one-dimensional patterns, we argue that they create

more indirect lighting because of their low frequency in one direction [9]. Moreover, estimating epipolar geometry can in some cases be a tedious or impossible task. For example, it is nowadays quite common to use catadioptric or other non conformal cameras or projectors in multi-projector systems [16].

In Sec. 2, we summarize previous works in coded light systems, in particular to achieve subpixel precision. In Sec. 3, we introduce our method to increase the amount of information of unstructured light patterns. In Sec. 4, we show how to recover subpixel correspondence on synthetic data. We validate the method on real scenes in Sec. 5 and compare with results of state of the art methods. We conclude and propose future works in Sec. 6.

## 2. Previous work

The goal of this paper is to achieve a high precision subpixel reconstruction for static scenes in the presence of several challenges like indirect illumination, scene discontinuities or projector defocus (see [13] for a list of standard problems). Many active reconstruction methods can work at subpixel precision levels (see [15, 14] for extensive reviews). However, their accuracy is widely affected by their lack of robustness to the aforementioned difficulties [10, 4]. Some improvements were made possible lately by a careful redesign of the projected patterns [10, 3, 7, 8].

Several methods are based on the projection of sinusoidal patterns which encode the projection position by a unique phase [18, 19]. The pattern must be shifted several times and several frequencies are often needed [11]. A limited photometric calibration is required since the phase estimation is directly related to the intensities affected by the gamma of the projector. Modulated phase shifting [3] was introduced to generate less indirect illumination and increase the accuracy of the subpixel correspondences. The method modulates the highest frequency patterns with orthogonal high frequency sine waves. The number of projected patterns needed is very high however since each pattern is itself modulated by several shifted patterns. The method described in [6] can be used to reduce the required number of patterns by multiplexing the modulated patterns together. Due to the periodic nature of the pattern, all the above methods require a "phase unwrapping" step to disambiguate the phase recovered. Phase unwrapping involves lower frequency patterns that can introduce large errors [11], in particular because of indirect lighting [13]. Recently, micro phase shifting was introduced in [10] to unwrap the recovered phases using only high frequency patterns. Due to low frequencies in one direction, the projected patterns still produce some indirect illumination that can affect the results.

Another category of methods [12, 4] use so-called unstructured light patterns that form temporal codewords to

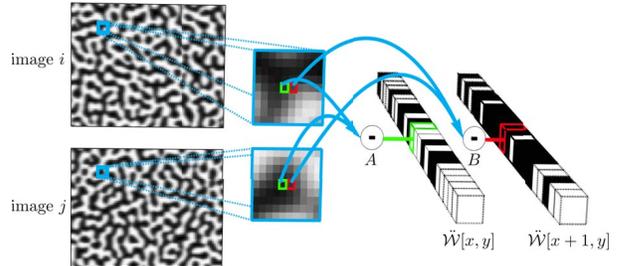


Figure 2. Bits are recovered by taking intensity differences between pairs of images. Two quadratic codes are shown for two adjacent pixels of the image pair  $(i, j)$ . The labels  $A$  and  $B$  illustrates the computation of a bit of  $\hat{\mathcal{W}}[x, y]$  and  $\hat{\mathcal{W}}[x + 1, y]$  as  $\text{bit}(c_i[x, y] - c_j[x, y])$  and  $\text{bit}(c_i[x + 1, y] - c_j[x + 1, y])$ .

identify each projector pixel uniquely, but require an explicit search to obtain correspondences. In [4], the patterns were designed to make constant the amount of indirect illumination, and the method was shown to be very robust. However, it did not yield subpixel accuracy reconstruction and required a lot of patterns.

## 3. From linear to quadratic code length

In [4], a camera pixel recovered a bit from the observed intensity by looking at the sign of the difference with the mean intensity over all patterns. The mean was considered a good reference because it is expected to be near constant when using a high enough frequency. For  $N$  patterns, a linear codeword  $\hat{\mathcal{W}}$  of  $N$  bits is generated by comparing each captured pattern  $c_i$  with the average image  $\bar{c}$  for each pixel  $\mathbf{p} = (x, y)$ . We have

$$\hat{\mathcal{W}}[\mathbf{p}] = \{\text{bit}(c_i[\mathbf{p}] - \bar{c}[\mathbf{p}]), 1 \leq i \leq N\} \quad (1)$$

where  $\text{bit}(a)$  has been defined as

$$\text{bit}(a) = \begin{cases} 0 & a < 0 \\ 1 & a > 0 \\ \text{random 0 or 1} & a = 0 \end{cases} \quad (2)$$

We propose to increase the codelength by considering all possible pairs of pattern images as illustrated in Fig. 2. This provides a codeword  $\ddot{\mathcal{W}}$  of quadratic length  $\binom{N}{2}$  defined as

$$\ddot{\mathcal{W}}[\mathbf{p}] = \{\text{bit}(c_i[\mathbf{p}] - c_j[\mathbf{p}]), 1 \leq i \leq N, i < j \leq N\}. \quad (3)$$

This quadratic code is very unstable for binary patterns however, since half the intensity comparisons will yield differences of 0. We next explain how to generate our patterns which alleviate this problem.

### 3.1. Blurred gray level pattern generation

We propose to use band-pass gray level patterns which are generated as follows. Similarly to [4], we first apply

a band-pass filter on white noise in the frequency domain, keeping only frequencies ranging from  $f$  to  $2f$  where  $f$  is the same parameter as in [4]. After taking the inverse Fourier transform, the pattern is a random gray level signal composed of a limited range of frequencies. To produce uniform contrast across the whole pattern, we then binarize the pattern using a threshold at its average intensity, and then apply a blur kernel to make the pattern grayscale once again. The blur deviation should be close to  $\frac{W}{6f}$  where  $W$  is the width of the image patterns, which is the average "radius" of black and white regions in our pattern (though the exact value used is not critical, see Sec. 4.4). In the next section, we analyse the number of patterns required to match.

### 3.2. Number of required patterns

Using these gray-level patterns, the quadratic code  $\ddot{W}$  now contains more information for each pixel than its linear counterpart  $\dot{W}$  but also some redundancy. The entropy of  $\dot{W}$  is clearly  $N$  bits. Since the entropy of  $N$  pairwise distinct elements is  $\log_2(N!)$  bits, out of the  $\frac{N^2-N}{2}$  bits of  $\ddot{W}$ , only  $\log_2(N!)$  actually provide information. As an example, 50 images will provide a quadratic code of length 1275 bits which effectively contains 214 bits of information. So a quadratic code from only 50 images is equivalent to a linear code of 214 images.

A minimum of 24 patterns is needed to uniquely encode each pixel of a  $800 \times 600$  projector. This number could be slightly decreased if one allows the use of median filtering on the correspondence map (we do not advise this however, see our results in Sec. 5.2). The number of patterns is also expected to be lower when the epipolar geometry is known. Note that, in our experiments, we chose to use more than the minimal number of patterns to remove the number of images as a source of errors and better assess the remaining reconstruction errors.

## 4. Achieving subpixel accuracy

As is the case with [4], the non-subpixel correspondence of a camera pixel is found using the LSH algorithm[2] that finds a match between the pixels of the camera and the projector, identified by the quadratic codes  $\{\dot{W}^c\}$  and  $\{\dot{W}^p\}$  respectively (using Eq. 3).

Assuming a camera-projector pixel ratio near 1, the camera pixel will generally see a mixture of four neighboring projector pixels. This mixture can be described by two parameters  $(\hat{\lambda}_x, \hat{\lambda}_y)$  where  $0 \leq \hat{\lambda}_x, \hat{\lambda}_y \leq 1$  which represent the subpixel matching disparity between camera pixel  $\hat{\mathbf{p}}$  and projector pixel  $\mathbf{p}$ .

Consider that a camera pixel  $\hat{\mathbf{p}} = (\hat{x}, \hat{y})$  has been matched to a projector pixel  $\mathbf{p} = (x, y)$ , using the LSH algorithm. To estimate  $(\hat{\lambda}_x, \hat{\lambda}_y)$ , we first need to find which quadrant represented by four projector pixels  $\{(x, y), (x +$

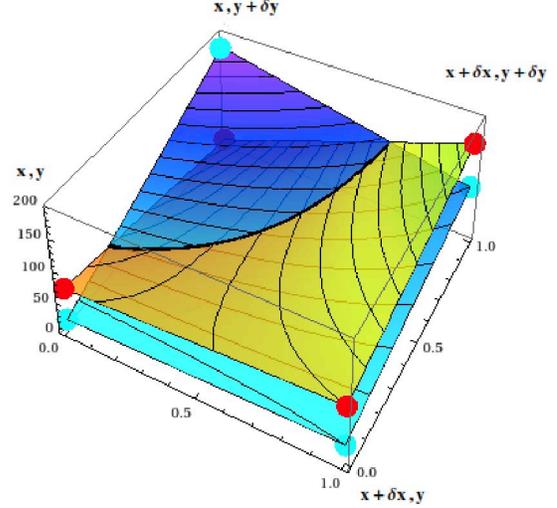


Figure 3. The red and cyan points corresponds to intensities of  $p_i$  and  $p_j$  respectively, for a quadrant out of four. The black curves represents the zero-crossing  $S_{ij}(x, y, \delta_x, \delta_y, \lambda_x, \lambda_y) = 0$ . Each pair  $(i, j)$  generates a 2D zero-crossing that provides constraints that are used to estimate the true subpixel position.

$\hat{\delta}_x, \hat{\delta}_y, (x, y + \hat{\delta}_y), (x + \hat{\delta}_x, y + \hat{\delta}_y)$  adjacent to  $\mathbf{p}$  out of the four possible, contains the sub-pixel match for  $\hat{\mathbf{p}}$ .

### 4.1. Selecting the right quadrant

There are four quadrants each composed of three projector pixels located around the matched projector pixel. The correct quadrant is selected as the pair  $(\hat{\delta}_x, \hat{\delta}_y)$  for which the difference between the camera and projector codes is minimal :

$$\hat{\delta}_x, \hat{\delta}_y = \arg \min_{\delta_x, \delta_y \in \{-1, 1\}} \left( \left| \dot{W}^c(x, y) - \dot{W}^p(x + \delta_x, y) \right| + \left| \dot{W}^c(x, y) - \dot{W}^p(x, y + \delta_y) \right| + \left| \dot{W}^c(x, y) - \dot{W}^p(x + \delta_x, y + \delta_y) \right| \right).$$

### 4.2. Estimating the subpixel position

For a projector pattern  $p_i$ , we model the interpolation of the intensities of the four neighboring projector pixels of a quadrant as a function of  $\lambda_x$  and  $\lambda_y$  using a bilinear plane:

$$K_i(x, y, \hat{\delta}_x, \hat{\delta}_y, \lambda_x, \lambda_y) = \lambda_y (\lambda_x p_i[x, y] + (1 - \lambda_x) p_i[x + \hat{\delta}_x, y]) + (1 - \lambda_y) (\lambda_x p_i[x, y + \hat{\delta}_y] + (1 - \lambda_x) p_i[x + \hat{\delta}_x, y + \hat{\delta}_y]). \quad (4)$$

The 2D intersection of the two bilinear planes defined by projector patterns  $p_i$  and  $p_j$  is obtained by solving

$K_i(x, y, \hat{\delta}_x, \hat{\delta}_y, \lambda_x, \lambda_y) = K_j(x, y, \hat{\delta}_x, \hat{\delta}_y, \lambda_x, \lambda_y)$ . We define :

$$\begin{aligned} S_{ij}(x, y, \hat{\delta}_x, \hat{\delta}_y, \lambda_x, \lambda_y) &= K_i(x, y, \hat{\delta}_x, \hat{\delta}_y, \lambda_x, \lambda_y) - \\ &K_j(x, y, \hat{\delta}_x, \hat{\delta}_y, \lambda_x, \lambda_y) \\ &= A + B\lambda_x + C\lambda_y + D\lambda_x\lambda_y \end{aligned} \quad (5)$$

where

$$\begin{aligned} A &= p_j[x, y] - p_i[x, y] \\ B &= p_j[x + \hat{\delta}_x, y] - p_i[x + \hat{\delta}_x, y] - A \\ C &= p_j[x, y + \hat{\delta}_y] - p_i[x, y + \hat{\delta}_y] - A \\ D &= p_j[x + \hat{\delta}_x, y + \hat{\delta}_y] - p_i[x + \hat{\delta}_x, y + \hat{\delta}_y] - C - B + A. \end{aligned} \quad (6)$$

Equation 5 defines a polynomial which can be evaluated at any position  $(\lambda_x, \lambda_y)$  inside the region defined by the quadrant. The sign of the value gives the side of the curve  $S_{ij}(x, y, \hat{\delta}_x, \hat{\delta}_y, \lambda_x, \lambda_y) = 0$  on which this point lies (see Fig. 3).

For each pair of patterns  $(p_i, p_j)$ , the pair is discarded if the two planes do not intersect. Otherwise, if  $\text{bit}(c_i[\hat{\mathbf{p}}] - c_j[\hat{\mathbf{p}}]) = \text{bit}(p_i[\mathbf{p}] - p_j[\mathbf{p}])$ , then the subpixel position should be located on the side of the curve towards  $\mathbf{p}$ . Conversely, if the bits are different, then it should be located on the other side of the curve. Thus, each pair  $(p_i, p_j)$  for which the planes intersect effectively provides a constraint on the value of the true subpixel location  $(\hat{\lambda}_x, \hat{\lambda}_y)$ . To account for the noise in camera codes, one cannot directly apply each constraint. In practice,  $(\hat{\lambda}_x, \hat{\lambda}_y)$  should be voted as the value that satisfy the most constraints. In the next section, we present a hierarchical approach to efficiently solve this problem.

### 4.3. Hierarchical voting

The true subpixel position  $(\hat{\lambda}_x, \hat{\lambda}_y)$  is the one satisfying the most constraints. It is found using a hierarchical voting scheme. At the highest level, the quadrant is divided into 4 equal square bins for  $0 \leq \lambda_x, \lambda_y \leq 0.5$ . Note that once the correct quadrant has been selected, the true subpixel location cannot be greater than 0.5 (otherwise the adjacent quadrant should have been selected). For each useful constraint, a bin gets voted if at least one of its corners is on the correct side of the curve. The process is then repeated recursively by dividing the winning square bin in four, until the desired amount of precision is reached. Note that if the two planes defined by a pair  $(p_i, p_j)$  do not intersect at some level, this constraint can be safely ignored at the next levels for more efficiency. For the experiments presented in this paper, we used 7 levels.

However, in practice, camera bits can have errors due to image noise, changes in surface albedo  $\alpha$  and the gamma

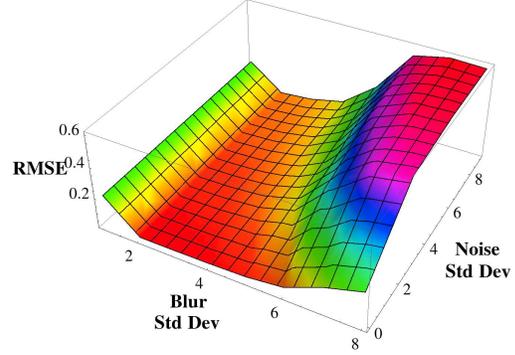


Figure 4. RMS subpixel error as a function of the standard deviations of the blur in pixels and the Gaussian intensity noise level.

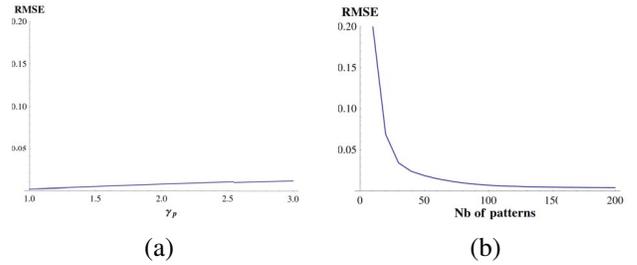


Figure 5. For our synthetic experiment, the estimated subpixel location (a) is only slightly affected by the gamma nonlinearity of the camera. (b) is improved by increasing the number of patterns.

factor  $\gamma_p$  of the projector. We evaluate their effects in the next section.

### 4.4. Effects of noise and gamma

Image noise and several other factors lead to misleading constraints on  $(\hat{\lambda}_x, \hat{\lambda}_y)$ . We ignore in this paper the effects of scene albedo as we assume that it is constant within the field of view of a single camera pixel. Fig. 4 plots the RMS subpixel error for different standard deviations of the blur in pixels and noise level. Synthetic subpixel positions were created by shifting 50 patterns of  $f = 64$  cycles per frame and a  $800 \times 600$  resolution. One can see that the exact blur deviation is not critical as there is a range going from 2 pixels to about 4 pixels that produce low error. In our experiments, we used a blur standard deviation of  $\frac{800}{6 \times 64} \approx 2$  (see Sec. 3.1). As for the gamma factor  $\gamma_p$ , Fig. 5(a) shows that its effect is very small when using 50 patterns. Finally, we also tested synthetically the error evolution when varying the number of patterns. Fig. 5(b) shows how the error decreases with the number of patterns. Note that, for all tests, we did not observe that the actual subpixel position has any effect on the RMSE (data not shown).

## 5. Experiments

In this section we describe the experimental setup we used to assess the quality of the reconstruction obtained

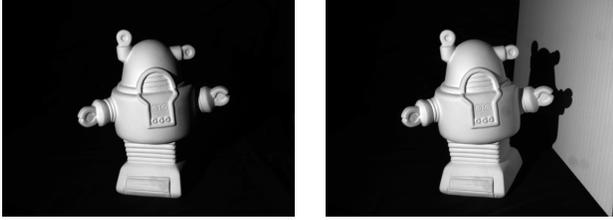


Figure 6. The robot was first scanned alone on a table (left), then a plastic board was added to the scene to create indirect lighting (right).

with our method. We first present quantitative results with respect to an object for which the ground truth was acquired using an Arius3D laser scanner. We then show various 3D reconstructions of a challenging scene and evaluate their quality by visual inspection. We compare our method to several other subpixel methods : the original phase shifting (PS) method of [19], modulated phase shifting (ModPS) presented in [3] and micro phase shifting (MicroPS) [10].

In all our experiments, we used a Samsung SP-400B projector with a resolution of  $800 \times 600$  pixels and a Prosilica GC-450C camera with a resolution of  $659 \times 493$ . If needed by the method, a gamma correction was applied to the projected patterns. Each device was weakly calibrated independently and their final intrinsic parameters and relative geometry were found by a bundle adjustment[1] for the purpose of 3D visualization. The set of points used for the bundle minimization is the intersection of the correspondences estimated by each method so as to not introduce any bias toward a specific method in the subsequent 3D error measurements.

### 5.1. A simple scene with a ground truth : the robot

As shown in Fig. 1, a robot model was used in our experiments. To obtain a ground truth, it was scanned by Arius3D<sup>1</sup> at a very high resolution (0.1mm sampling, 10 micron RMS error).

In order to measure the sensibility of each reconstruction method to interreflection, we reconstructed the robot, with and without interreflections from a nearby plane (see Fig. 6). Fig. 7 illustrates how each method performed on a section of the robot model not affected by interreflections. Each method performed equally well, and the precision of the reconstruction is quite good. For fair comparison, every method used a budget of approximately 50 patterns to perform the scan. In order to do so, PS used 8 frequencies from  $1/8$  to  $1/1024$ , 3 shifts per frequency for each direction (horizontal and vertical) for a total of 48 projected patterns. ModPS used 4 frequencies from  $1/16$  to  $1/1024$  : the highest frequency was modulated by 6 shifted versions of an orthogonal sinus wave of frequency  $1/16$  and each

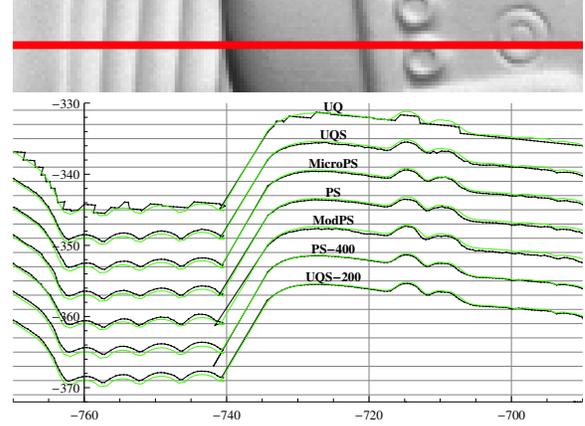


Figure 7. X-Z projection of reconstructed robot models for various methods. Units are in mm. The green curve is the reference scan. The portion of the robot which is reconstructed is illustrated in the cropped image at the top of the curves.

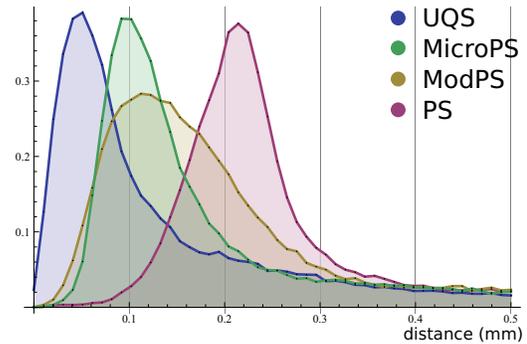


Figure 8. Histogram of reconstruction variations for the robot scene featuring strong interreflections.

frequency used 3 shifts. The unwrapping used the method of [11] and 9 patterns per direction, so ModPS used a total of 54 patterns. Finally, we slightly modified the original MicroPS method to use more than 3 shifted version of the highest frequency used to compute the wrapped phase. In [10], only 7 images were used. Since we wanted to use 50 patterns for all methods, 14 images were dedicated to unwrapping the phase in each direction (we used the frequencies recommended by the authors in the data available on their webpage<sup>2</sup>) and 11 shifted versions of a high frequency sine wave were used to compute the phase for a total of 50 patterns. We also added the results of UQ [4] which is not a subpixel reconstruction but provides a scale to appreciate how well all the subpixel algorithms perform. PS-400 is the result of PS using 25 patterns per frequency (as opposed to 3 which is the minimum). UQS-200 is our method using 200 patterns.

We then evaluated the difference between 3D reconstructions and the ground truth in the area affected by interreflec-

<sup>1</sup><http://www.arius3d.com>

<sup>2</sup><http://www.cs.columbia.edu/CAVE/projects/MicroPhaseShifting>

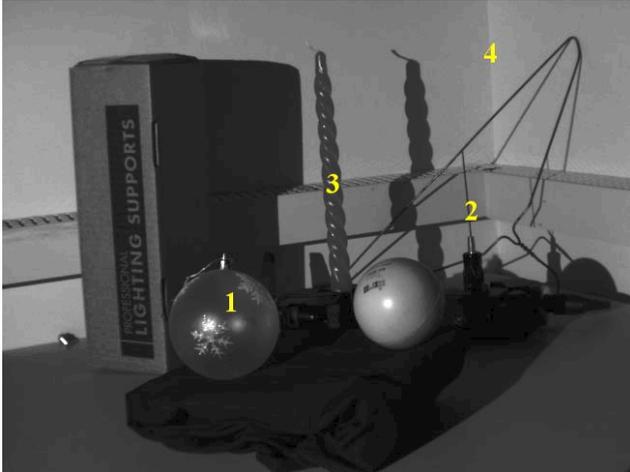


Figure 9. A complex scene featuring (1) translucency, (2) sharp discontinuities, (3) subsurface scattering and (4) interreflections.

tions. The histogram of variations is illustrated in Fig. 8. Our method (UQS) is the least affected, followed by MicroPS, ModPS and finally PS. It was expected that PS would be the worst performer since it does not feature robustness to interreflections. Overall, our results and those of MicroPS are very similar. They will be compared further on a more challenging scene in the following section.

## 5.2. Comparison with Micro Phase Shifting on a complex scene

In this experiment, we scanned a scene composed of several objects which feature different materials and properties (see Fig. 9).

The 3D reconstructions we obtained with both methods is shown in Fig. 10. The top row shows the reconstruction using 50 patterns for UQS, and 50 patterns for MicroPS. The bottom row shows the same reconstruction using 200 projected patterns for each method (for MicroPS, 86 patterns were used in each direction to estimate the phase). The reconstructions are similar for both methods at 50 patterns, even though some errors can be spotted in the reconstruction of slanted surfaces by MicroPS. It is however clear that UQS produces better results using 200 patterns. In particular, reconstruction was successful on a large region of the transparent Christmas ball, whereas MicroPS did not improve its results using 200 patterns, as shown in Fig. 11.

Note that the MicroPS method uses 1D high frequency patterns to unwrap and compute the phase. These generate more interreflections than our 2D patterns. This is especially visible in the corner at the back of the scene, where two bumps are falsely reconstructed as a result of some indirect lighting bouncing of each wall, as seen in Fig. 12.

Discontinuities can also be problematic for MicroPS. For instance, correspondences are erroneous on sharp edges or at the border of a discontinuity, as seen on Fig. 13. When

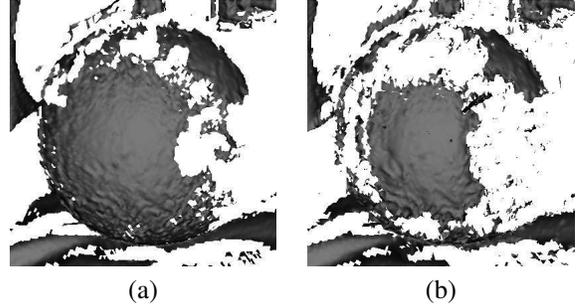


Figure 11. Reconstruction of the translucent Christmas ball with (a) UQS (b) MicroPS. A larger portion of the ball is reconstructed using UQS.

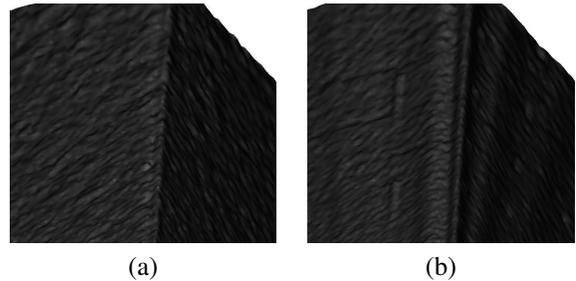


Figure 12. Reconstruction of the corner between the two walls with (a) UQS (b) MicroPS. Two bumps on each side of the corner are falsely reconstructed using MicroPS due to the indirect illumination generated by its 1D patterns.

using only 7 patterns as presented in [10], a median filter is applied to correct unwrapping errors and noisy phase estimates due to low signal to noise ratio. Since we used a lot more images, we found that the median filter was overall no longer necessary. However, when applied, the median filter does correct some errors (pixels on the edge of the ball for instance), but also removes the correspondences found on small objects like the screwdriver as shown in Fig. 13. MicroPS suffers from a trade-off between correspondence errors in discontinuities and the lack of correspondences on small objects. On the other hand, since our method does not rely on the use of a median filtering and naturally performs well in discontinuities, it does not feature this limitation.

## 6. Conclusion

We proposed a method to produce highly accurate sub-pixel correspondence using a projector and a camera. It relies on the principles of unstructured light scanning methods that are robust to common and challenging difficulties arising in active scanning systems. We use continuous gray scale patterns produced in the frequency domain. Subpixel position is estimated by comparing every pair of images and considering the location of zero-crossings. Each pair of images contributes a bit in quadratic codes that increase the information used in the subpixel estimation but also decreases the number of patterns needed to match. The method shown

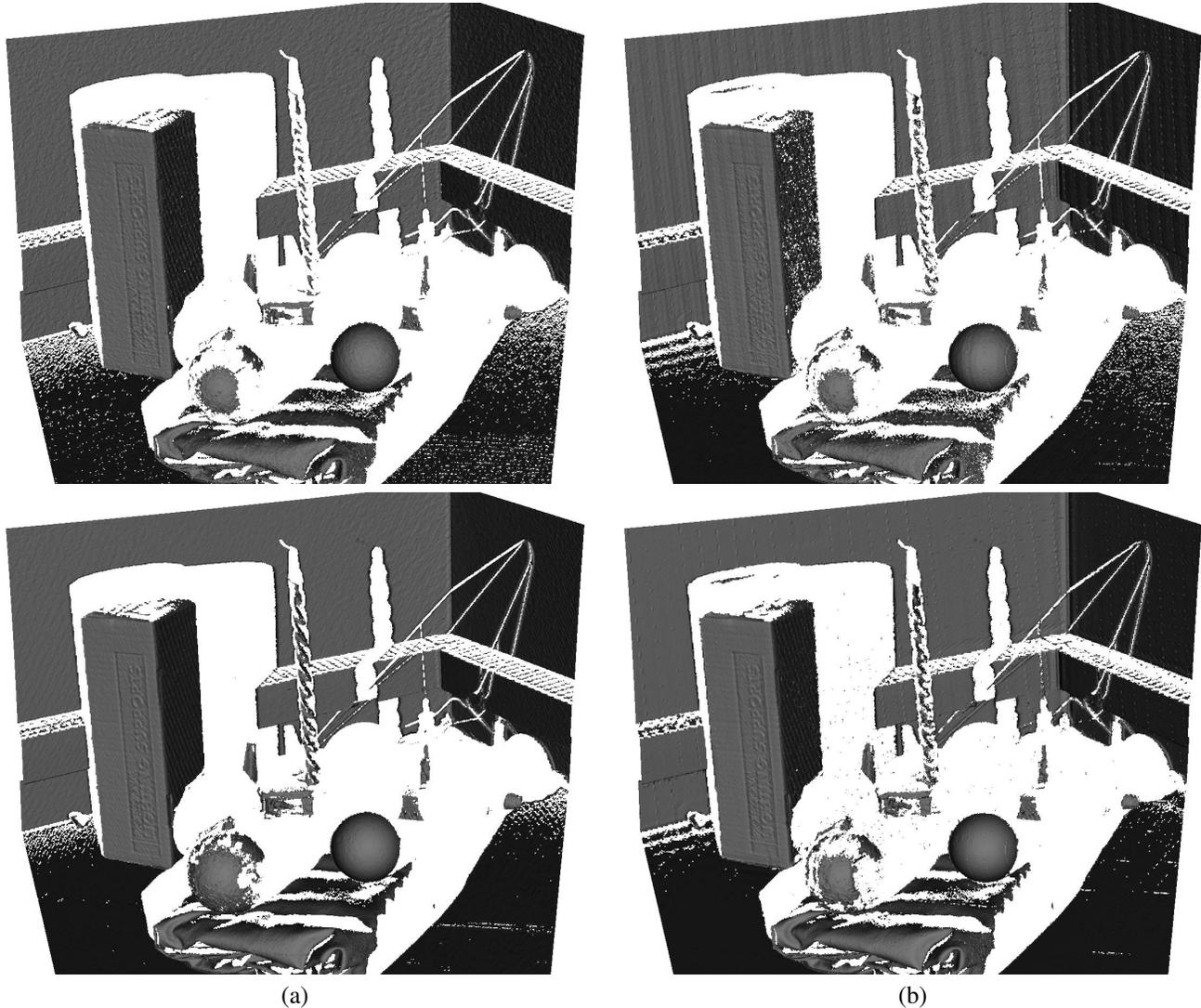


Figure 10. Reconstruction of a complex scene with (a) UQS (b) MicroPS. The number of patterns used was 50 (top row) and 200 (bottom row).

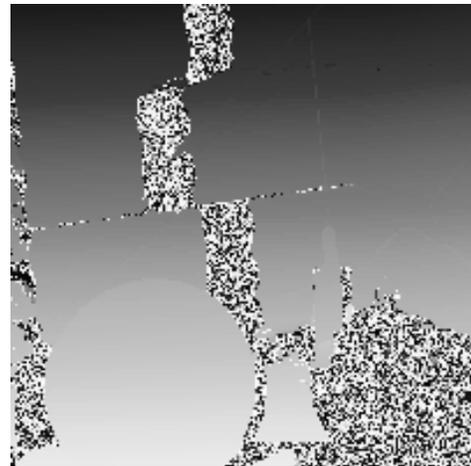
does not require knowledge of the epipolar geometry nor any photometric calibration. Reconstructions produced by our method were in general comparable to the ones produced by state of the art phase shifting methods, but showed increased robustness to indirect illumination and depth discontinuities.

## References

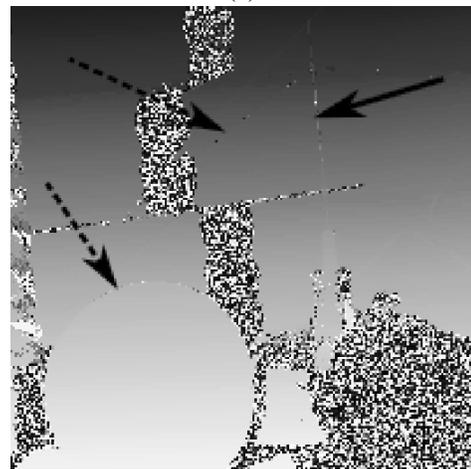
- [1] S. Agarwal and K. Mierle. *Ceres Solver: Tutorial & Reference*. Google Inc. 5
- [2] A. Andoni. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *IEEE Symposium on Foundations of Computer Science*, pages 459–468. IEEE Computer Society, 2006. 3
- [3] T. Chen, H.-P. Seidel, and H. P. A. Lensch. Modulated phase-shifting for 3d scanning. In *CVPR*, 2008. 2, 5
- [4] V. Couture, N. Martin, and S. Roy. Unstructured light scanning to overcome interreflections. In D. N. Metaxas, L. Quan, A. Sanfeliu, and L. J. V. Gool, editors, *ICCV*, pages 1895–1902. IEEE, 2011. 1, 2, 3, 5
- [5] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(2):296–302, Feb. 2005. 1
- [6] J. Gu, T. Kobayashi, M. Gupta, and S. K. Nayar. Multiplexed Illumination for Scene Recovery in the Presence of Global Illumination. In *IEEE International*

*Conference on Computer Vision (ICCV)*, pages 1–8, Nov 2011. 2

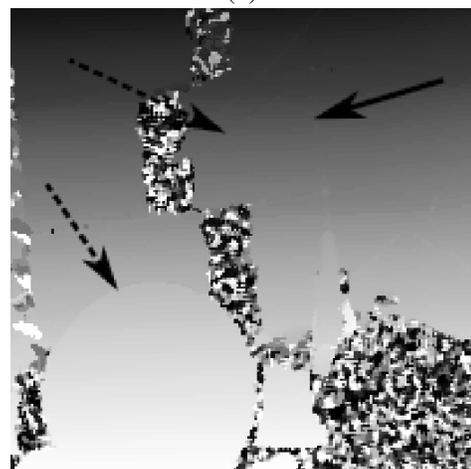
- [7] J. Gühring. Dense 3-d surface acquisition by structured light using off-the-shelf components. *Videometrics and Optical Methods for 3D Shape Measurement*, January 2001. 2
- [8] M. Gupta, A. Agrawal, and A. Veeraraghavan. A practical approach to 3D scanning in the presence of inter-reflections, subsurface scattering and defocus. *International Journal of . . .*, 2013. 2
- [9] M. Gupta, A. Agrawal, A. Veeraraghavan, and S. G. Narasimhan. Structured light 3d scanning in the presence of global illumination. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 713–720. IEEE Computer Society, 2011. 2
- [10] M. Gupta and S. K. Nayar. Micro Phase Shifting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, Jun 2012. 2, 5, 6
- [11] J. M. Huntley and H. Saldner. Temporal phase-unwrapping algorithm for automated interferogram analysis. *Appl. Opt.*, 32(17):3047–3052, Jun 1993. 2, 5
- [12] A. Kushnir and N. Kiryati. Shape from unstructured light. In *3DTV07*, pages 1–4, 2007. 1, 2
- [13] S. K. Nayar, A. Krishnan, M. D. Grossberg, and R. Raskar. Fast separation of direct and global components of a scene using high frequency illumination. *ACM Transactions on Graphics*, 25:935–944, 2006. 2
- [14] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado. A state of the art in structured light patterns for surface profilometry. *Pattern Recognition*, 43(8):2666 – 2680, 2010. 1, 2
- [15] J. Salvi, J. Pagès, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37:827–849, 2004. 1, 2
- [16] J.-P. Tardif, S. Roy, and M. Trudeau. Multi-projectors for arbitrary surfaces without explicit calibration nor reconstruction. *International Conference on 3-D Digital Imaging and Modeling*, pages 217–224, 2003. 2
- [17] Y. Wexler, A. W. Fitzgibbon, and A. Zisserman. Learning epipolar geometry from image sequences. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:209, 2003. 1
- [18] C. Wust and D. Capson. Surface profile measurement using color fringe projection. *Machine Vision and Applications*, Jan 1991. 2
- [19] S. Zhang and S. Yau. High-speed three-dimensional shape measurement system using a modified two-plus-one phase-shifting algorithm. *Optical Engineering*, Jan 2007. 2, 5



(a)



(b)



(c)

Figure 13. Correspondence maps of the screwdriver and hanger using (a) UQS (b) MicroPS without median filtering (c) MicroPS with 5x5 median filtering. Errors on the edges (dashed arrows) are present without median filtering, but sharp edges on small objects (plain arrow) disappear when applied.