Depth Imaging by Combining Time-of-Flight and On-Demand Stereo

Uwe Hahne and Marc Alexa

TU Berlin, Germany
hahne@cs.tu-berlin.de, marc.alexa@tu-berlin.de
http://www.cg.tu-berlin.de

Abstract. In this paper we present a framework for computing depth images at interactive rates. Our approach is based on combining time-of-flight (TOF) range data with stereo vision. We use a per-frame confidence map extracted from the TOF sensor data in two ways for improving the disparity estimation in the stereo part: first, together with the TOF range data for initializing and constraining the disparity range; and, second, together with the color image information for segmenting the data into depth continuous areas, enabling the use of adaptive windows for the disparity search. The resulting depth images are more accurate than from either of the sensors. In an example application we use the depth map to initialize the z-buffer so that virtual objects can be occluded by real objects in an augmented reality scenario.

1 Introduction

Real time depth imaging is a building block in many interactive vision systems and, in particular, is necessary for enabling realistic occlusions in augmented reality (AR). Despite the improved speed of general purposes computing as well as development of new types of sensors, providing depth images in *real time* continues to be a challenging problem. For the purposes of enhancing AR with convincing occlusions current approaches are limited to either reducing quality in the depth maps [SNV02] or realizing occlusions by compositing [VH08]. We are demonstrating a first step toward an affordable and lightweight solution by fusing information from a low cost but also low resolution time-of-flight range sensor with standard correlation-based stereo.

Quite generally, we can distinguish active and passive approaches to real time depth imaging. Active optical techniques involve relighting the scene and usually require an expensive and heavy setup. Recently, sensors based on the time-of-flight principle for sensing depth have become affordable and fast with the introduction of photonic mixer devices (PMD) [MKF+05,XSH+05]: the reflection of modulated IR light is collected in a CMOS matrix. Comparing the signal to the source modulation yields the phase, which is a linear function of distance to the reflecting surface. PMD depth imaging works at interactive rates, but suffers from comparably low spatial resolution of the sensor and noise in the depth values, especially for surfaces with low reflectance.

Passive techniques, at least when several frames per second are required, are based on multiple views of scene captured with two or more cameras. We have decided to

R. Koch and A. Kolb (Eds.): Dyn3D 2009, LNCS 5742, pp. 70–83, 2009.

use a single binocular stereo camera, as more not only make the setup more complicated, but also require processing more images. From the large number of stereo vision approaches [SS02] only local correlation based methods are fast enough for real time application [FHM+93, Hir01]. This limits the quality of the resulting depth maps, most obviously in large featureless areas but also at depth discontinuities, where the correlation window might compare different objects because of occlusion.

We combine the camera systems (time-of-flight and stereo) and fuse the data so that limitations of each of the individual sensors are compensated. Our goal is enhancing the high resolution color image from one of the stereo camera oculars with depth information, gathered from the PMD camera as well as from disparity estimation. The mapping of the PMD depth image into a color image acquired with another camera (resp. texturing the depth data with the color image) has been analyzed by Reulke [Reu06] and Lindner et al. [LKH07,LLK07]. Our setup combines the PMD camera with a binocular stereo camera, similar to [BBK07, GAL07, HA07, NMCR08, KS06, ZWYD08].

For better explaining our choice of algorithm, we need to briefly touch on the setup, calibration, and properties of the cameras (section 2). The physical properties of the PMD camera give rise to the preprocessing of its data, most importantly the estimation of confidence values for each depth value (section 3). Kuhnert and Stommel [KS06], as well as Netramai et al. [NMCR08] use a similar confidence map to choose either the depth value acquired with the PMD camera or depth from stereo – we exploit this depth/confidence map for initializing and steering a local correlation based stereo algorithm (section 4), in particular by choosing adaptive windows for the correlation based on the information in both the color images and the range image.

In an earlier approach [HA07], we combined the TOF data from the PMD camera with high resolution images from two photo cameras, using graph cuts to find a globally optimal solution for a depth map of a single perspective. The use of graph cuts leads to computation times that are insufficient for real-time video processing. Similarly, Guomundsson et al. [GAL07], Zhu et al. [ZWYD08], and Beder et al. [BBK07] generate depth images by fusing TOF and stereo data. Their approaches appear to be much faster than using graph cuts, however, they target single images and provide no information on the computation times. The choices of stereo algorithm, however, indicate that they are not amenable to real-time processing in their current form. We explicitly start from the restrictive setting of real-time applications, which severely restricts the choice of stereo algorithm, mostly to local correlation with fixed windows. We use the TOF information particularly to adapt the windows, as fixed windows fail at depth discontinuities. We believe our approach yields depth images at interactive rates

- that are are more reliable than the information from the PMD camera without compromising the interactive frame rate and
- that are more accurate around depth discontinuities than real time stereo vision approaches based on fixed window correlation.

We demonstrate our use of the system in an augmented reality (AR) scenario for computing accurate occlusions between virtual and real objects.

2 Setup

We mount a compact time-of-flight camera PMDTec type [vision]19k, capturing a depth range of about 7.5 meters at a resolution of 160x120 pixels, together with PointGrey Bumblebee2 stereo camera capturing color video at a resolution of 640x480, on an aluminum rack (see Figure 1 for an image of the cameras). Both cameras are aligned to parallel viewing directions.

2.1 Photonic Mixer Device Depth Camera

Photonic Mixer Devices (PMD) are semiconductor sensors that can be used to measure distances per pixel based on the time-of-flight principle [XSH+05, MKF+05]. The camera system includes a light source based on infrared LEDs, illuminating the scene with a continuously modulated signal at $f = 20$MHz. The sensor detects the phase shift between the source and received signal by sampling four values per period. The phase shift is in principle independent of the amplitudes of the signals, and since modulation frequency f and speed of light c are constants, it relates linearly to the distance of the reflecting object.

Measuring phase shifts has several inherent limitations: first, phase shifts have symmetries along the signal and are unique only in an interval of π, in this case $c/2f = 7.5$m. More subtly, measuring phase shifts assumes perfectly sinusoidal signals, yet this is not the case. This leads to a “wiggling" error in the depth measurement (see Rapp et al. [Rap07]). Most importantly, phase shifts can only be measured accurately if the sensor receives the right amount of light. If objects are too dark sensor noise dominates the signal, while too much light leads to saturation and makes modulation undetectable. As a consequence, measurements depend on object reflectance and distance to the light source and camera. The camera allows controlling the integration time (similar to

Fig. 1. All three cameras mounted on an aluminum bar

exposure time in standard cameras), however, just as in photography it is often impossible to keep all objects captured in the scene within the dynamic range of the sensor.

The camera provides the raw signal samples as well as consolidated amplitude and depth values computed from the four samples. Interestingly, the consolidated amplitude value is accurate only as long as the sensor is not being saturated. It is possible to detect saturation based on the raw signal samples (see Rapp et al. [Rap07]).

The resulting depth images captured with PMD technology are noisy and contain wrong depth values because of the phase ambiguity and objects reflecting not enough light; whereas we have eliminated the case of reflecting too much light by adjusting the integration time.

2.2 Calibration

Sensor fusion requires registration and accurate calibration. For both the intrinsic and extrinsic calibration we use the calibration algorithm of Zhang [Zha99, Zha00]. The PMD camera, however, yields intensity images that are too noisy for direct application and they are preprocessed following the ideas of Reulke [Reu06] as well as Lindner and Kolb [LK06]. A relevant practical problem for the extrinsic calibration is the misalignment of optical center and zero depth plane of the PMD camera. Interestingly, Gudmundsson et al. [GAL07] perform a stereo calibration between all pairs of cameras. We have found this to be cumbersome, because of the combination of noisy amplitude images and the mismatch between optical center and depth image for the PMD camera. We rather consider only the extrinsic calibration between the depth image from PMD camera and the systems of the color cameras.

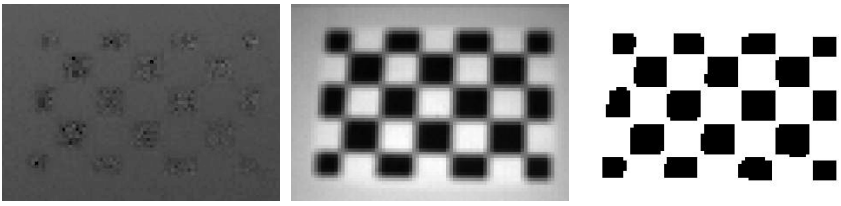
The stereo camera color images are rectified to reduce the correspondence problem to a single line. Our calibration is accurate enough so that the depth difference between stereo system and the PMD camera is within the accuracy of the PMD camera.

3 Preprocessing and Confidence Estimation

As explained in the last section, the quality of the depth values captured by the PMD camera depends strongly on the surface of the objects in the scene. Dark and glossy surfaces lead to artifacts as the modulated IR light is not reflected as expected. Especially when using the depth values for determining occlusions in AR applications, these artifacts become clearly visible. We process the data prior to using it with the stereo system, trying to improve the data by simply filtering and assigning confidence values to each depth value. Very low confidence depth values are replaced by interpolated values with higher confidence.

3.1 Filtering

Reducing the noise or removing outliers is one obvious part of the pre-process. Isolated outliers can be removed at minimal cost using median filtering. Through experimentation we have found that a small kernel of 3×3 pixel is sufficient. This appears to be due to the outliers being mostly isolated pixels. Larger kernels would lead to longer processing times without showing a significant improvement.



(a) Depth image (b) Amplitude image (c) Confidence map

Fig. 2. A checkerboard is difficult to reconstruct using PMD range sensing, because of the insufficient amount of light reflected by the black areas. The acquired depth image (a) clearly holds wrong depth values in these black areas. The amplitude image (b) can be used to compute a confidence map (c), which is thresholded to classify depth values as valid (white) or invalid (black)

3.2 Confidence Estimation

As explained in the last section, in our setup, the dominant cause for systematically wrong depth estimation are objects that have bad reflection properties due to their material and color. However, this information is available in form of an amplitude image of the scene.

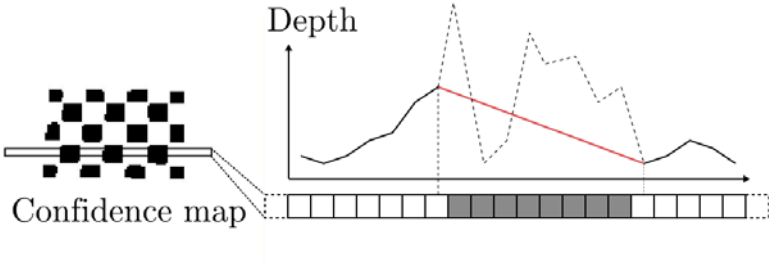
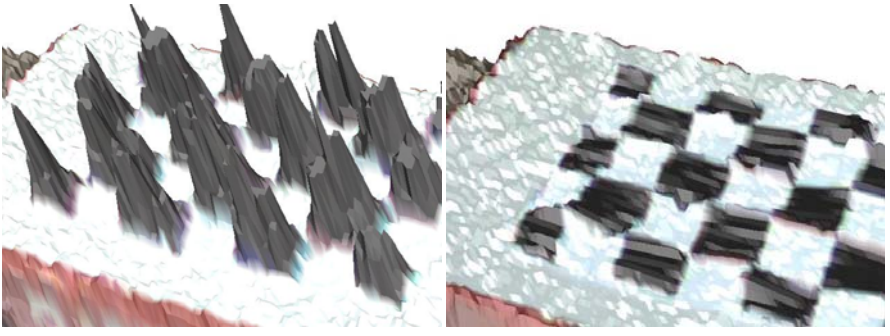


Fig. 3. Scanline interpolation of the PMD data using the confidence map. The red line is the interpolated depth, the dashed line is the original unreliable PMD depth.

The first step in turning the amplitude image into a confidence map is applying a 3×3 median filter, similar to the process for the depth image. The resulting image is thresholded, yielding a binary confidence map classifying depth values as either valid or invalid. Figure 2 highlights the problems resulting from low reflectivity at the example of a checkerboard and shows the resulting classification of valid and invalid depth values (while the 3D reconstruction from the wrong depth values can be seen in Fig. 4(a)).

3.3 Interpolation

In all regions marked as confident by the binary map, we will use depth values for initializing disparities. In most cases the depth values in insufficiently reflecting areas are far from the truth and it is better to assume continuity in depth.



(a) Original image (b) With interpolation

Fig. 4. 3D reconstruction of the scene from the depth image (and using the intrinsic camera geometry). The left image shows the reconstruction from the data in Figure 2(a) and the right image uses linearly interpolated depth values for elements with low confidence.

As in our case the interpolated depth values will later be corrected using the stereo information, we opt for a simple approach that is as fast as possible: the depth image is scanned across horizontal lines. When an invalid segment is encountered it is replaced by either a line connecting the two valid depth values at the boundary of the segment or a line with constant depth values at the boundaries of the image (see Figure 3). For textured planar surfaces (such as the checkerboard, see Figure 4(b)) this provides a reasonable estimate; if objects of low reflectance differ in depth from the surrounding they will be assigned wrong depth values, which will be corrected in the stereo part of the algorithm.

4 Algorithm

In our exemplary AR applications with dynamic occlusions we need to enhance one color image from the stereo camera with depth information. We compute depth values at the resolution of the PMD camera. The algorithm we suggest is equally applicable for computing depth at higher resolution or textured surfaces in other views.

The main steps of assigning depth values to pixels are as follows (see also Figure 5):

1. The pixel coordinates and depth values from the PMD camera are used for generating a tessellated depth surface (i.e. quad mesh) of the scene from this viewpoint.
2. The surface is transformed into the view space of the cameras. The intersections of view rays with the surface in this coordinate system define initial disparity values; the associated confidence values define the possible range. Thresholding the confidence values yields a set of valid and invalid depth coordinates in the quad mesh.
3. The areas of pixels associated to valid and invalid depth coordinates are thinned and serve as the initialization of a segmentation of the color image into depth continuous regions.

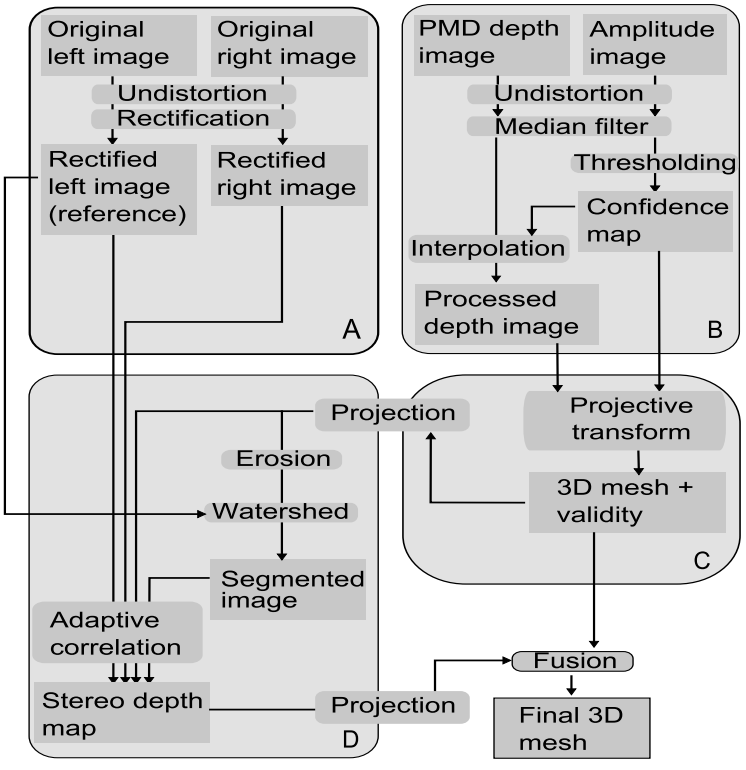


Fig. 5. Algorithm overview

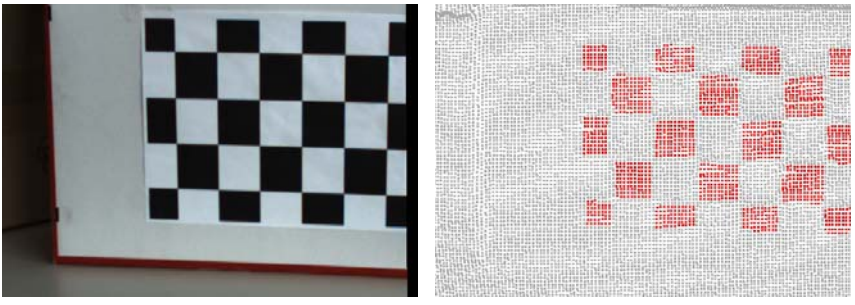
4. The segmentation steers adaptive windows for the correlation computation in a standard stereo algorithm, correcting the invalid depth coordinates of the surface mesh.

In the following, we discuss several details of these steps.

4.1 Mesh Initialization and Projection

The intrinsic calibration of the PMD camera allows computing 3D coordinates from pixel location in the image plane and the corresponding depth value. For convenience, we connect the 3D coordinates to a piecewise bilinear mesh. The extrinsic calibration between the cameras allows transforming the mesh into the coordinate systems of the stereo camera. The intrinsic calibration of the stereo cameras (including a rectification) defines a projective transformation, which yields depth and confidence values per pixel.

The labels define two regions in the color images: a region of valid and a region of invalid pixels, based on the binary confidence map. Figure 6 shows the projection of the mesh in the left camera. The projection of valid vertices are drawn as gray squares and invalid pixels are drawn as red squares.



(a) Rectified Left View (b) Stereo operation map

Fig. 6. Color coded vertices of the surface mesh, where red vertices are invalid and will be corrected using stereo vision

4.2 On-Demand Stereo with Adaptive Windows

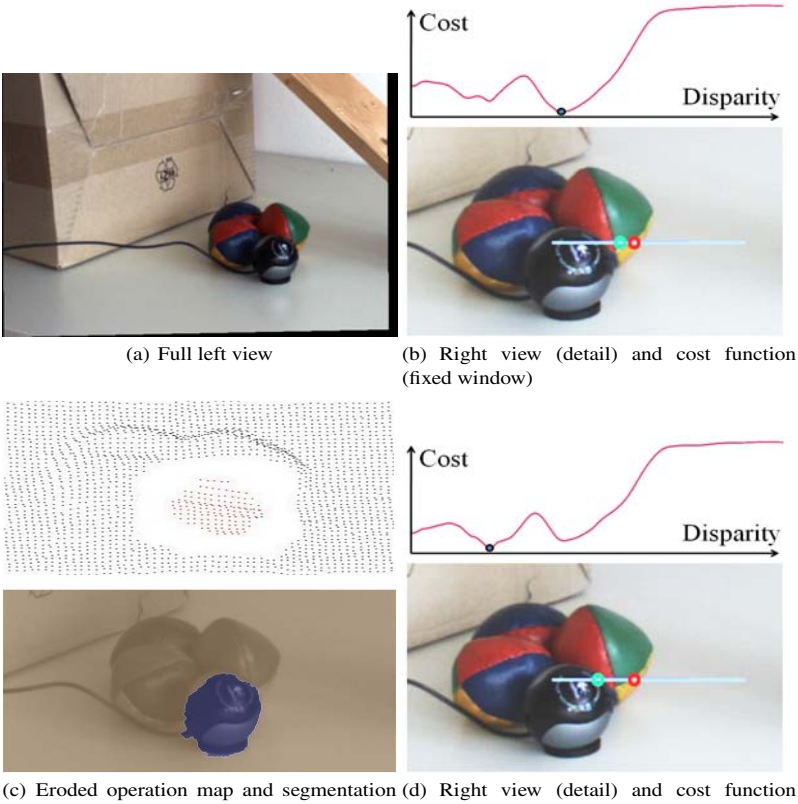
Instead of computing a whole disparity map, the use of our stereo part is computing depth values only for vertices that are marked invalid. Furthermore, the projection of these vertices into the two rectified stereo views immediately yields an initial disparity guess.

An underlying assumption of correlation based stereo algorithms is that depth is unambiguous in the correlation window. This is not the case at depth discontinuities where objects may be occluded in only one of the views so that correlation of pixel colors fails to be a good indicator for correspondence (Figure 7(b) illustrates this situation).

A solution to this problem is to adapt the correspondence window to the (likely) object boundaries. Kanade and Okutomi [KO94] suggest to adapt the size and shape of the rectangular correlation window to local disparity characteristics. Boykov et al. generalize this variable window approach [BVZ98]. They compute for each pixel a new window. This window contains all pixels with an intensity close to the considered pixel. This way, they try to model the boundaries of the objects and the depth discontinuities. Hirschmüller [Hir01] proposes a similar approach using multiple supporting windows. Unfortunately, all of these techniques are too costly to reach interactive rates at video resolution of 640 by 480 pixels.

Our main observation is that the object boundaries are only relevant if they are in regions whose depth values are labeled as invalid – otherwise the depth values have already been gathered based on TOF. Thus, we can use the information on valid and invalid regions for initializing a segmentation algorithm in the color images. The segmentation will then define the extent of the correlation windows used in our adaptive window stereo algorithm. Exploiting the confidence information makes our approach both much faster and also more robust than only working with the color images.

From the many potential segmentation algorithms we use the *marker-controlled watershed* algorithm [RM00], which we have found to be robust while being fast enough for our application scenario. The idea is that valid and invalid regions serve as markers for the binary segmentation. Because of errors in the projections for vertices with incorrect depth (i.e. especially invalid vertices), color pixels are not necessarily labeled correctly. Consequently, the sets of valid and invalid pixels are eroded independently,



(a) Full left view

(b) Right view (detail) and cost function (fixed window)

(c) Eroded operation map and segmentation (both detail)

(d) Right view (detail) and cost function (adaptive window)

Fig. 7. This figure compares correlation based stereo with fixed windows and with windows adapted to object boundaries computed from segmenting the color image into depth continuous regions. The whole scene is shown (a), while we focus on the group of balls and the webcam in front of the box (b-d). The eroded operation map is used to initialize the watershed segmentation (c) leading to a mask adapting the stereo correlation windows. The red circle shows the initial disparity guess and the green circle the disparity corresponding to minimum cost (b+d).

leaving a set of unlabeled pixels in the proximity of object boundaries (see Figure 7(c)). These sets of valid and invalid pixels serve as the markers that initialize the segmentation as starting points. If objects have boundaries in the color images, the segmentation will accurately label pixels as being connected to the valid or invalid pixels. The resulting binary map restricts the correlation window.

Figure 7 shows the influence of this border correction filter: an object is too dark for the PMD camera, yielding wrong depth values and marked as invalid. A correlation based stereo algorithm with fixed window finds the wrong corresponding point (7(b)). After eroding the sets of invalid and valid pixels, the watershed algorithm segments the object along its boundary (7(c)). Restricting the window to the segmented object yields the correct correspondence (7(d)).

4.3 Final 3D Mesh

Finally, we replace the PMD depth of each invalid pixel with the computed stereo depth value. The resulting mesh is rendered into the z-buffer of one of the camera views using the color values from the camera. This allows using the depth values in interactive applications.

5 Results

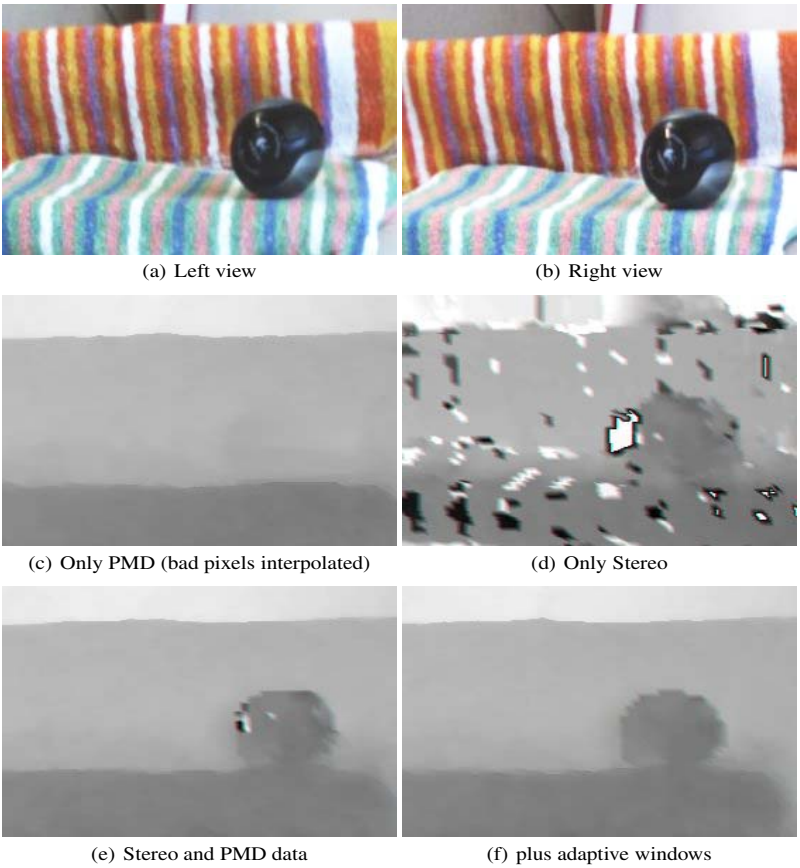
As occlusion is one of the most fundamental factors in monoscopic depth perception, AR applications become much more immersive if occlusion handling is embedded. This is demonstrated in Fig. 8 and the accompanying video [htt09]. Our approach makes it possible to handle dynamic occlusions like a hand in front of the virtual objects, unlike model based techniques [BWRT96]. This is very practicable as most AR application focus on manual interaction of virtual and real objects.

(a) no occlusion (b) occlusion

Fig. 8. The occlusion strongly enhances the depth impression of the scene

Other approaches to occlusion handling in AR have used stereo vision [KOTY00] or PMD depth sensing [FHS07]. We have compared the raw output of the PMD camera projected into the view of the color image and the raw stereo data with the results of our approach (see Fig. 9). A dark object in this scene is assigned incorrect depth values delivered by the PMD camera and, consequently, is mapped to the background through the interpolation process (Fig. 9(c)). Fig. 9(d) shows the depth map obtained using correlation based stereo with a fixed window as used in other stereo algorithms aiming at interactive frame rates. The images in 9(e) and 9(f) show the improved results using our algorithm. Notice the appearance of dark objects while they are not captured at all by the PMD camera.

Several parameters influence the performance of our system: the frame rates we obtain are limited by the cameras and the transmission to roughly 11fps on our test system, an AMD Athlon 2GHz Dual Core Processor with 1GB RAM. The additional cost of our algorithm depends on the size of the correlation window and the number of depth values



(a) Left view (b) Right view

(c) Only PMD (bad pixels interpolated) (d) Only Stereo

(e) Stereo and PMD data (f) plus adaptive windows

Fig. 9. In this example we compare the depth map acquired with the PMD camera (c) and using correlation based stereo (d) with our results, i.e. correcting low confidence areas of the PMD range image using stereo on-demand with a fixed window approach (e) and using adaptive correlation windows based on segmentation (f)

that have to be corrected using stereo on-demand. Table 1 compares several situations, where we have chosen the thresholds so that approximately 250 resp. 500 depth values were considered invalid. The computation times clearly show that there is a linearity between time and correlation window size for stereo computation on the one hand and on the other hand, the ratio between the amount of corrected pixels and computational timings is linear as well.

6 Discussion

Our system provides a framework for interactive AR applications, where the depth map is necessary for visual or physical interaction between synthetic and real objects. Our approach is using low cost, light weight components and exploits the properties of both

Table 1. Performance on an AMD Athlon 2.00 GHz Dual Core Processor with 1GB RAM. Window is the size of the correlation window used for stereo. The computation times are compared for each step while capturing a scene with approx. 250 and 500 corrected pixels.

Acquisition	91 ms		
Preprocessing	23 ms		
Stereo+PMD	Window	$\approx$ 250 px	$\approx$ 500 px
	5×5	**9.5** ms	20.5 ms
	7×7	16 ms	33.3 ms
	9×9	25 ms	55 ms
	11×11	42 ms	88 ms

sensor types. In particular, we use range data for narrowing the disparity search and adapting the correlation windows to potential depth discontinuities.

We demonstrate how the resulting system can be used for handling occlusions. The depth data can also be used for collision detection and other interactive solutions. Some of the possible interactions are demonstrated in the accompanying video [htt09]. The dynamic and global depth map of the scene would also allow computing shadows for virtual objects, or using higher quality rendering techniques for further improving the realism of virtual objects [BR05].

Our system can be improved in several aspects. The synchronization of the PMD camera as part of the software system results in a maximum of 11fps – using a hardware solution would allow exploiting the maximum frame rate of the cameras. The computations necessary for fusing the data and improving the depth images are easy to distribute to several cores so that exploiting a higher input frame rate would be easy if the system were coupled with a modern CPU. As mentioned in section 2.2, the accuracy is depending on the PMD range data, and we think that future developments in camera hardware and calibration would lead to an increased working range.

It would be interesting to apply different segmentation algorithms for adapting the stereo windows. Feris et al. [FRC+05] use region growing in a similar situation. Finding the best balance between performance and accuracy in this step is important future work. In addition, it might be possible to make use of other information than confidence and color, such as range data or predictions from preceding frames. Another step in enhancing the algorithm would be to define a confidence measure for the stereo data and use it for further controlling the depth reconstruction. This could reduce the error in regions where both systems fail, for example large dark and homogeneous objects.

References

[BBK07] Beder, C., Bartczak, B., Koch, R.: A combined approach for estimating patchlets from PMD depth images and stereo intensity images. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 11–20. Springer, Heidelberg (2007)

[BR05] Bimber, O., Raskar, R.: Spatial Augmented Reality. A K Peters, Ltd., Wellesley (2005)

[BVZ98] Boykov, Y., Veksler, O., Zabith, R.: A variable window approach to early vision. IEEE Trans. Pattern Anal. Mach. Intell. 20(12), 1283–1294 (1998)

[BWRT96] Breen, D.E., Whitaker, R.T., Rose, E., Tuceryan, M.: Interactive occlusion and automatic object placement for augmented reality. Computer Graphics Forum 15(3), 11–22 (1996)

[FHM+93] Faugeras, O., Hotz, B., Mathieu, H., Viéville, T., Zhang, Z., Fua, P., Théron, E., Moll, L., Berry, G., Vuillemin, J., Bertin, P., Proy, C.: Real time correlation based stereo: algorithm implementations and applications. Technical Report RR-2013, INRIA (1993)

[FHS07] Fischer, J., Huhle, B., Schilling, A.: Using time-of-flight range data for occlusion handling in augmented reality. In: Eurographics Symposium on Virtual Environments (EGVE), pp. 109–116 (2007)

[FRC+05] Feris, R., Raskar, R., Chen, L., Tan, K., Turk, M.: Discontinuity preserving stereo with small baseline multi-flash illumination. In: IEEE International Conference in Computer Vision (ICCV 2005), Beijing, China (2005)

[GAL07] S. Guðmundsson, S., Aanæs, H., Larsen, R.: Fusion of stereo vision and time-of-flight imaging for improved 3D estimation. In: International workshop in Conjuction with DAGM 2007: Dynamic 3D Imaging, September 2007, vol. 1, pp. 164–172 (2007)

[HA07] Hahne, U., Alexa, M.: Combining time-of-flight depth and stereo images without accurate extrinsic calibration. In: International workshop on Dynamic 3D Imaging, Heidelberg, September 2007, pp. 78–85 (2007)

[Hir01] Hirschmüller, H.: Improvements in real-time correlation-based stereo vision. In: SMBV 2001: Proceedings of the IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001), Washington, DC, USA, p. 141. IEEE Computer Society, Los Alamitos (2001)

[htt09] `http://www.cg.tu-berlin.de/fileadmin/fg144/Research/Publications/video/dyn3d09.mp4` (June 2009)

[KO94] Kanade, T., Okutomi, M.: A stereo matching algorithm with an adaptive window: Theory and experiment. IEEE Trans. Pattern Anal. Mach. Intell. 16(9), 920–932 (1994)

[KOTY00] Kanbara, M., Okuma, T., Takemura, H., Yokoya, N.: A stereoscopic video see-through augmented reality system based on real-time vision-based registration. In: Proceedings. IEEE Virtual Reality, 2000, pp. 255–262 (2000)

[KS06] Kuhnert, K.-D., Stommel, M.: Fusion of stereo-camera and pmd-camera data for real-time suited precise 3d environment reconstruction. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, October 2006, pp. 4780–4785 (2006)

[LK06] Lindner, M., Kolb, A.: Lateral and depth calibration of PMD-distance sensors. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Nefian, A., Meenakshisundaram, G., Pascucci, V., Zara, J., Molineros, J., Theisel, H., Malzbender, T. (eds.) ISVC 2006. LNCS, vol. 4292, pp. 524–533. Springer, Heidelberg (2006)

[LKH07] Lindner, M., Kolb, A., Hartmann, K.: Data-fusion of pmd-based distance-information and high-resolution rgb-images. In: International Symposium on Signals, Circuits and Systems (ISSCS), Iasi, Romania (2007)

[LLK07] Lindner, M., Lambers, M., Kolb, A.: Sub-pixel data fusion and edge-enhanced distance refinement for 2d/3d images. In: Dynamic 3D Imaging (Workshop in Cunjunction with DAGM 2007), Heidelberg, Germany (September 2007)

[MKF+05] Moeller, T., Kraft, H., Frey, J., Albrecht, M., Lange, R.: Robust 3d measurement with pmd sensors. Technical report, PMDTec (2005)

[NMCR08] Netramai, C., Melnychuk, O., Chanin, J., Roth, H.: Combining pmd and stereo camera for motion estimation of a mobile robot. In: The 17th IFAC World Congress (July 2008) (accepted)

[Rap07] Rapp., H.: Experimental and theoretical investigation of correlating tof-camera systems. In: Physics, Faculty for Physics and Astronomy, University of Heidelberg, Germany (September 2007)

[Reu06] Reulke, R.: Combination of distance data with high resolution images. In: ISPRS Commission V Symposium 'Image Engineering and Vision Metrology' (2006)

[RM00] Roerdink, J.B.T.M., Meijster, A.: The watershed transform: Definitions, algorithms and parallelization strategies. FUNDINF: Fundamenta Informatica 41 (2000)

[SNV02] Schmidt, J., Niemann, H., Vogt, S.: Dense disparity maps in real-time with an application to augmented reality. In: WACV 2002: Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision, Washington, DC, USA, p. 225. IEEE Computer Society, Los Alamitos (2002)

[SS02] Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vision 47(1-3), 7–42 (2002)

[VH08] Ventura, J., Höllerer, T.: Depth compositing for augmented reality. In: SIGGRAPH 2008: ACM SIGGRAPH 2008 posters, p. 1. ACM, New York (2008)

[XSH+05] Xu, Z., Schwarte, R., Heinol, H., Buxbaum, B., Ringbeck, T.: Smart pixel - photonic mixer device (pmd) new system concept of a 3d-imaging camera-on-a-chip. Technical report, PMDTec (2005)

[Zha99] Zhang, Z.: A flexible new technique for camera calibration. Technical report, Microsoft Research (1999)

[Zha00] Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(11), 1330–1334 (2000)

[ZWYD08] Zhu, J., Wang, L., Yang, R., Davis, J.: Fusion of time-of-flight depth and stereo for high accuracy depth maps. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR (2008)