

---

## Sub-Pixel Data Fusion and Edge-Enhanced Distance Refinement for 2D/3D Images

---

Marvin Lindner, Martin Lambers and  
Andreas Kolb

Computer Graphics Group  
Institute for Vision and Graphics  
University of Siegen, Germany  
E-mail: [marvin.lindner, martin.lambers, andreas.kolb]@uni-siegen.de

**Abstract:** An important field of research in computer vision is the 3D analysis and reconstruction of objects and scenes. A rather new technology in this context is the *Photonic Mixer Device (PMD)*, based on the time-of-flight principle, which measures full-range distance information in real-time. Unfortunately, PMD-based devices have still limited resolution and provide only IR intensity information.

This paper describes a fast algorithmic approach to combine high resolution RGB images with PMD distance data, acquired using a binocular camera setup. The resulting combined RGBZ-data not only enhances the visual result, but also represents a basis for advanced data processing in e. g. object recognition with sub-pixel accuracy. A simple but efficient method is used to detect geometric occlusion caused by the binocular setup, which otherwise will lead to false color assignments.

Additionally, we introduce an enhanced filtering technique used for the edge-enhanced distance refinement of the geometry provided by the PMD camera. The technique incorporates a proper handling of boundaries and an iterative refinement approach, which can be used to enhance the 2D/3D-fusion accuracy.

**Keywords:** TOF; Photonic Mixer Device (PMD); RGB; Fusion; Range Image Refinement

---

### 1 Introduction

In automatization areas like robotics or automotive, the reconstruction of objects and scenes is a necessary fundamental with respect to computer vision. Information obtained from digitized scenes represent important input data for position determination, online object recognition or collision prevention.

In this context, typical expensive and complex setups including laser scanners or stereo vision systems are used for distance measurements. However, laser scanning techniques are rather time-consuming and thus impracticable for dynamic scenes. Stereo vision camera systems, on the other hand, completely rely on the fast identification of corresponding points. Mismatching correspondences especially in homogeneous object regions lead to an incomplete reconstruction.

A rather new and promising approach developed in recent years estimates the distance by time-of-flight measurements for modulated, incoherent light based on the new Photonic Mixer Device (PMD) technology [2, 12]. Integrating a PMD camera into a vision system, full-range distance information is available at interactive frame rates up to 15 fps. Nevertheless, there are still problems to be solved, e. g. the relatively low sensor resolution or the simple IR intensity image provided by the camera. Thus, many algorithms known from image processing cannot be applied to PMD data directly.

The contribution of this paper is a combined 2D/3D vision system, consisting of a PMD and an RGB-camera, which improves the capabilities of downstream processing tasks like object recognition. The system incorporates the refinement of low resolution range images including boundary handling and edge enhancement as well as a 2D/3D fusion technique consisting of full range color mapping and occlusion detection. The distance refinement improves the geometry provided by the PMD camera and thus increases the accuracy of 2D/3D data fusion.

A short overview of known related work is given in Sec. 2. The design of our fusion model is described in Sec. 3. The distance refinement approach is discussed in more detail in Sec. 4, whereas the final data fusion is treated in Sec. 5. Finally, the results are given in Sec. 6.

## 2 Related Work

### 2.1 Data Fusion

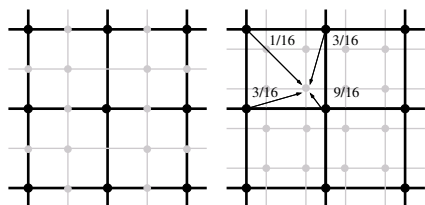
So far, there exist only a few distinguished approaches to combine PMD distance information with high-resolution RGB image data, differing in their particular realization concerning hardware and software implementation.

Prasad et al. [6] describe a hardware-based realization, which combines a traditional CCD and a PMD chip in a monocular device using an optical splitter. Due to the monocular setup, no special mapping transformation between both images has to be done as both images consist of the same view.

Another approach using the combination of a 2D- and a 3D camera has been presented by Reulke [8]. Unlike the hardware-based realization, a binocular camera setup is used, as in our case. The final data fusion is achieved by an algorithm called *orthophoto generation* [5]. Here, the 2D image is distorted in order to eliminate the perspectivity of the image by taking the 3D information into account. The result is an orthographic image, where all optical rays are parallel. After the image rectification, the data fusion is straight forward applying a parallel mapping of the color information onto the appropriate distance data.

### 2.2 Up-Scaling Techniques

Simple techniques for color images usually refine the pixel-grid by adding new pixels between existing pixels applying nearest neighbor or bilinear interpolation (see Fig. 1 left). In contrast to this, biquadratic filtering instead places the new pixel grid according to biquadratic B-spline subdivision scheme [11], thus neglecting the original sample locations on the new level (see Fig. 1 right) and resulting in



**Figure 1** Left: Standard refinement reusing existing pixel; Right: Biquadratic scheme using a shifted pixel grid.



**Figure 2** 2D/3D camera setup consisting of a 19k PMD and a high-resolution CCD-camera mounted on top.

a slightly shifted pixel grid. By making use of bilinear interpolation, biquadratic filtering can be implemented easily, giving improved results by avoiding the typical diamondshaped artifacts of bilinear interpolation and yielding a  $C^1$  continuous limit function.

Higher interpolation schemes like bicubic interpolation exist, but do not give crucial advantages for image filtering. Furthermore, high order interpolation schemes have been discarded due to the size of their filter masks and the resulting case differentiation for invalid pixel handling in the border regions (cmp. Sec. 4).

### 3 Overview

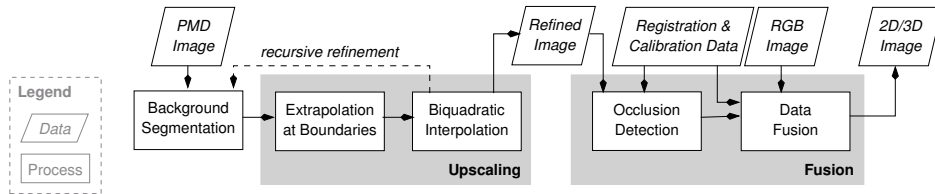
In Fig. 3 an overview about the overall fusion process is given. The processing pipeline consists of two main blocks: the distance refinement and the actual data fusion. Prior to both, a simple background segmentation based on the PMD pixel's amplitude is performed, discarding unreliable distance values, so-called *invalid pixels*, by simple thresholding.

During the distance refinement two main processing tasks are performed due to the perhaps invalid pixels:

1. An extrapolation of valid pixels to restore missing data for the biquadratic interpolation scheme and thus avoid shrinking of the valid pixel areas.
2. An enhanced biquadratic upscaling of the valid regions incorporating edge preserving as well as special treatment of still incomplete grid cells.

During the data fusion an occlusion detected prevents false color mapping in regions where the color camera is unable to provide proper information due to the bifocal setup of the combined vision system (see Fig. 2).

The actual data fusion takes the refined distance information as well as the additional high-resolution color image as input in order to combine both images to an enhanced scale independent dataset. Here both images do not necessarily have the same resolution. Especially the color image is normally of higher resolution providing additional information for computer vision tasks.



**Figure 3** The overall distance refinement approach.

As the data fusion approach highly depends on the correct reconstruction geometry and therefore on proper distance information, the PMD has been calibrated first using the depth calibration model described in [3, 4].

Due to the fixed setup, these parameters have to be determined only once during an initialization step.

#### 4 Distance Refinement

The goal of distance refinement is to supply more accurate depth information for the data fusion as it is done by simple linear interpolation. In our case, we concentrate on mono-modal upscaling schemes using depth data only, so no additional input data or preprocessing steps are necessary.

Since we head for real-time data processing, we apply filter-based biquadratic upscaling techniques preventing exhaustive pre-processing, beside the mentioned simple background segmentation. The proposed upscaling technique implements a *pyramidal upscaling*, i.e. the image is scaled by a factor of two in a recursive manner [11].

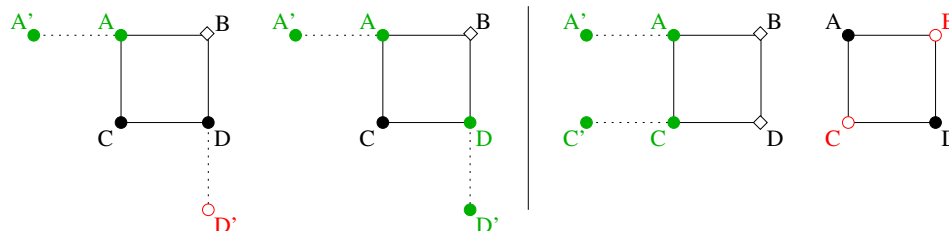
In contrast to standard images, low-resolution PMD images provide inaccurate distance information for pixel covering depth inhomogeneity, yielding so-called *flying pixels*, or areas with insufficient incident active light. The latter is marked as *invalid* during the background segmentation.

When upscaling a PMD depth image, it is very important to avoid propagating data from both flying and invalid pixels during the interpolation. Thus, a special treatment of both is necessary and discussed more into detail in the following sections.

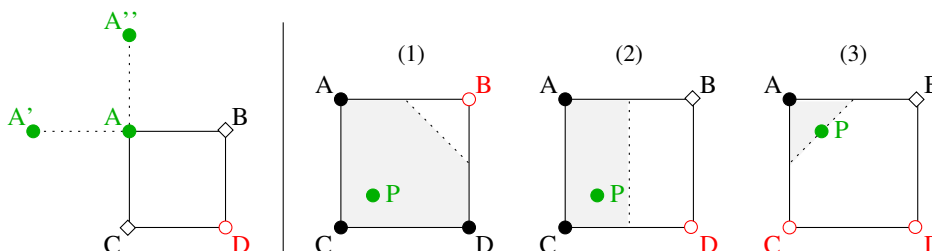
##### 4.1 Enhancement of Jumping Edges

Instead of a plain PMD pixel interpolation, the implemented interpolation incorporates a gradient-based resampling approach as proposed by Kraus et.al. [1]. Gradient-based resampling techniques are used for upsampling RGB images, preserving sharp edges in the upscaled image version. In our case, the gradient and Laplacian is used to adjust the sampling locations for the interpolation, i.e. the coordinates of each sample location are displaced by an offset vector  $\vec{d}$

$$\vec{d}(x) = \frac{-\sigma \cdot \Delta f(x)}{|\nabla f(x)|^2} \nabla f(x) \quad (1)$$



**Figure 4** Extrapolation of invalid pixel ( $\bullet$  valid,  $\circ$  invalid and  $\diamond$  extrapolated pixel). Left: Extrapolation in case of one invalid sample:  $B = 2A - A'$  or  $B = \frac{1}{2}(2A - A' + 2D - D')$ ; Right: Extrapolation in case of two invalid samples:  $B = 2A - A'$ ,  $D = 2C - C'$ . In the second case, no extrapolation is performed.



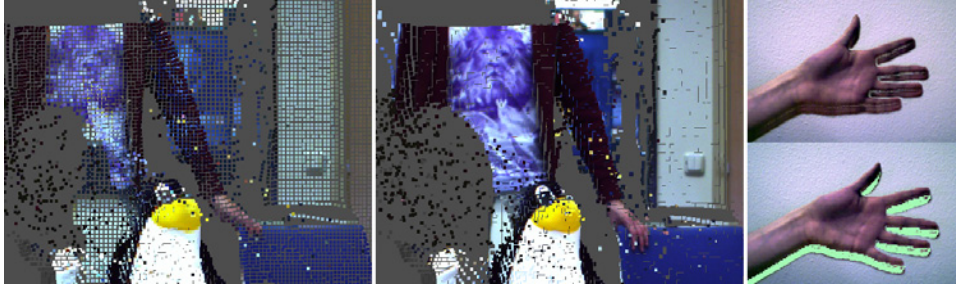
**Figure 5** Left: Extrapolation in case of three invalid samples:  $B = 2A - A'$ ,  $C = 2A - A''$ ; Right: Interpolation using valid ( $\bullet$ ) or extrapolated ( $\diamond$ ) samples only: (1)  $P = \frac{1}{4}A + \frac{1}{2}C + \frac{1}{4}D$ , (2)  $P = \frac{1}{4}B + \frac{3}{4}C$ , (3)  $P = \frac{3}{4}A + \frac{1}{4}B$ .

away from the edge avoiding edge smoothing and discarding flying pixel information for the sampling of sub-pixel locations. Here, the parameter  $\sigma \in [0, 1]$  influences the displacement, i.e. for  $\sigma = 0$  the method is equivalent to the original biquadratic interpolation.

As the offset  $d$  depends on the gradient  $\nabla f(x)$  and the Laplacian  $\Delta f(x)$ , a simple Gaussian is applied to smooth the data for the gradient and the Laplacian computation and thus to reduce the influence of noise on the 2<sup>nd</sup> order scheme. Furthermore, to apply the method to PMD depth images, we ensure that the gradient and Laplacian are approximated based on valid depth values only. Invalid depth values were simply discarded from the respective filtering masks.

Since the computation of gradient and Laplacian may be unstable, the magnitude of  $\vec{d}$  is clamped to a user-specified maximum  $m$ . Kraus et.al. [1] suggest a value of  $m \approx 0.25$ , which worked fine in our experiments. To avoid strong edge sharpening in rather homogeneous regions, we also introduced a lower threshold  $t_g$ . Thus, if the length of the gradient  $\nabla f(x)$  is smaller than  $t_g$ , the location correction is omitted.

We further improved the original edge-directed approach by applying the offset correction iteratively: the sampling coordination are repeatedly displaced by  $\vec{d}$  until the length of  $\vec{d}$  is smaller than a threshold  $t_d$ , or until the maximum number of iterations is used. This moves the sample location away from the edge center until a position on neighboring objects has been reached. This copes also with the fact, that a single step approach generally does not identify a proper resample location as the difference between the real edge and the second order edge model behind Eq. 1 increases with farther distance to the original sampling location.



**Figure 6** Left: Simple color mapping using quads as underlying geometry. Middle: 2D/3D data fusion using projective texturing. In both cases no distance refinement or outlier suppression has been applied. Right: Image mapping without (top) and with hidden surface removal (bottom). Areas not seen by the RGB camera are marked light green.

#### 4.2 Handling of Invalid Pixels

In the context of upscaling PMD data, the interpolation should be applied to cells with four valid pixels only in order to suppress invalid pixel propagation (see Fig. 1). For this reason, we apply an extrapolation of the boundary depth values, in order to expand the valid pixel region artificially without actually changing the original validity of a pixel. The detailed extrapolation schemes for one and two invalid pixels are depicted in Fig. 4 whereas the three pixel case is shown in Fig. 5 left.

During the interpolation step, the interpolation itself is still restricted to meaningful sample locations w.r.t. the original boundary to preserve smooth edges. Invalid pixels still may contribute to the interpolation if extrapolation has failed. In these situations simpler interpolation schemes like based on linear interpolation schemes are applied in order to obtain smooth boundary contours also in these cases (see Fig. 5 right).

Instead of using biquadratic interpolation only and thus skipping still understaffed cells, special treatment provides a much smoother boundary shape even if it is less accurate compared to biquadratic interpolation. An extrapolation should be performed in all cases as it leads to more accurate depth continuity than interpolating valid border pixels only. Thus it is a necessary step, whether biquadratic or simpler interpolation schemes are used.

## 5 Data Fusion

Having the acquired 2D and (possibly upscaled) 3D image of both cameras, the correct color value for a PMD pixel is given by its back-projection into world coordinates and a subsequent perspective projection onto the RGB image plane. However, pixelwise projection describes only an elementary one-to-one mapping, which generally results in an RGB image undersampling and a loss of information. Often a more detailed mapping is desired, e. g. in object recognition. Fig. 6 shows an example.

### 5.1 Projective Texturing

Unlike the approach taken by Reulke [8], our fusion approach is based on projective texture maps rather than on orthophoto generation. Projective textures have first been introduced by Segal et al. [9]. Here, the image of the RGB camera is assumed to be projected onto the geometry provided by the PMD camera distance information. Due to the involved perspective projection a simple linear color interpolation yields incorrect results.

Assuming, we have two points  $S_1^P$  and  $S_2^P$  in the image plane of a virtual camera (equivalent to the real PMD camera but with higher resolution), the rasterization of any intermediate point using linear interpolation

$$S^P = (1 - t) \cdot S_1^P + t \cdot S_2^P \quad (2)$$

requires the proper computation of the associate location  $S^R$  in the RGB image plane. According to Segal [9],  $S^R$  is the  $(x, y)$ -component of

$$P^R/w^R = \frac{(1 - t) \cdot P_1^R/w_1^P + t \cdot P_2^R/w_2^P}{(1 - t) \cdot w_1^R/w_1^P + t \cdot w_2^R/w_2^P} \quad (3)$$

Eq. 3 can be computed on programmable graphics hardware using the fixed functionality of the rasterization stage interpolating  $(x, y)^R/w^P$  and  $w^R/w^P$  linearly, resulting in accurate sub-pixel RGB information.

### 5.2 Hidden Surface Removal

Due to the different viewing directions of both cameras, an incorrect mapping mainly in the near range may occur (see Fig. 6 right). These artifacts are caused by occlusion, e.g. in concave regions or at object contours. For these surface regions, the RGB camera is unable to provide a proper object color and the projective mapping described in Sec. 5.1 samples erroneous color.

In order to prevent this mismapping, we adopt a rendering approach comparable to shadow maps [7, 10]. The main idea is to store the closest geometry portion w.r.t. the RGB sensor in a depth-buffer. Therefore, each geometry vertex is transformed into the RGB camera coordinate system first, where its distance to the RGB camera's origin is automatically interpolated and written to an off-screen render target.

During the color assignment, every vertex point is again transformed into the RGB camera coordinate system, where its current distance  $z^R(x, y)$  is compared with the corresponding z-Buffer entry. If the current fragment is farther away from the RGB sensor than the distance information stored in the frame buffer, i.e.

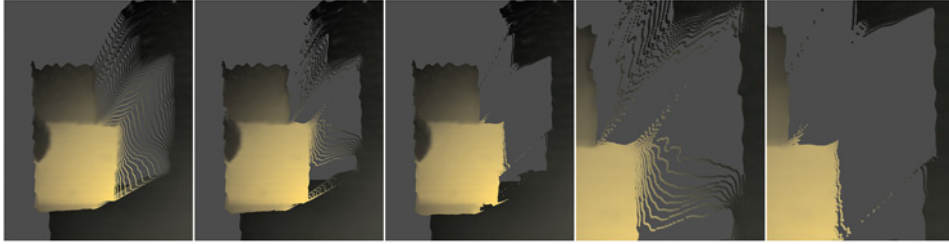
$$z^R(x, y) > z\text{Buffer}^R(x, y) + \varepsilon, \quad (4)$$

the fragment is hidden and the color assignment is omitted. Where the  $\varepsilon$ -offset is required to account for quantization and noise in the depth image.

## 6 Results

The fusion of 2D image data with the 3D geometry retrieved from the PMD-data clearly adds visual improvement to the resulting data (see Fig. 6 left). Additionally,





**Figure 7** Results of pyramidal interpolation schemes (total upscale factor 8) for a  $64 \times 48$  PMD sensor: (a) Biquadratic; (b) and (d) Edge-directed Biquadratic; (c) and (e) our iterative edge-directed approach.

the fused 2D/3D data open up new possibilities for scene analysis, e.g. for object recognition or distance refinement. A proper distance calibration is absolutely necessary due to the algorithm’s dependence on an accurate geometric reconstruction.

The distance refinement has been tested with a  $64 \times 48$  and a  $160 \times 120$  PMD camera. The  $64 \times 48$  has a better signal-to-noise ratio resulting in less distance noise. Fig. 7 shows the different interpolation approaches for a  $64 \times 48$  distance image. Using the edge-directed interpolation technique [11] already improves results. As shown in Fig. 7d, the object edges appear sharper, but a staircase effect remains. Applying our iterative method, both the staircase effect as well as the flying pixels are heavily reduced and flat regions are still smooth (Fig. 7e). Concerning the boundary handling, the proposed extrapolation technique results in much better shaped boundary curves, compared with a standard biquadratic interpolation approach (see Fig. 8).

Combining fusion and distance refinement for a  $160 \times 120$  PMD camera reveals improved fusion results (see Fig. 9). Due to the suppression of invalid and flying pixels, color mismatches hardly occur. It should be noted, that the simple threshold based segmentation using the PMD autocorrelation amplitude does not always lead to optimal results close to object boundaries. More appropriate methods have to be investigated.

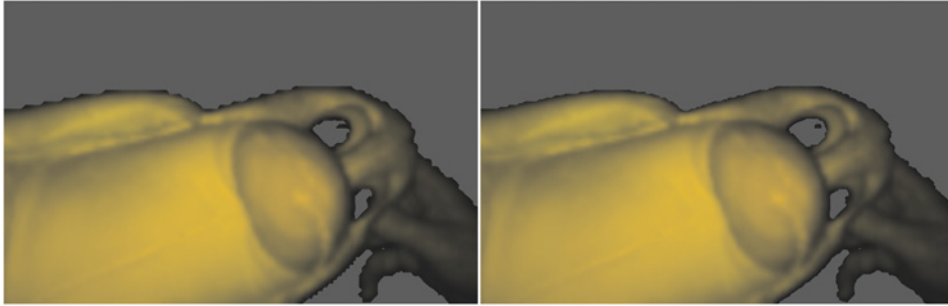
## Acknowledgment

This work is partly supported by the German Research Foundation (DFG), KO-2960/5-1.

## References and Notes

- 1 M. Kraus, M. Eissele, and M. Strengert. GPU-based edge-directed image interpolation. In *Proc. Scandinavian Conference on Image Analysis (SCIA)*, 2007.
- 2 R. Lange. *3D Time-Of-Flight Distance Measurement with Custom Solid-State Image Sensors in CMOS/CCD-Technology*. PhD thesis, University of Siegen, 2000.
- 3 M. Lindner and A. Kolb. Lateral and depth calibration of PMD-distance sensors. In *Int. Symp. on Visual Computing (ISVC)*, volume 2, pages 524–533. Springer, 2006.





**Figure 8** Boundary handling for a  $160 \times 120$  PMD sensor (upscale factor 8): (a) Biquadratic refinement; (c) Our approach based on boundary extrapolation.



**Figure 9** Results of combined refinement and 2D/3D data for a  $160 \times 120$  PMD sensor (upscale factor 8): (a) Raw input data; (b) Biquadratic refinement; (c) our approach (green pixels indicate occlusions).

- 4 M. Lindner, A. Kolb, and K. Hartmann. Data-fusion of PMD-based distance-information and high-resolution RGB-images. In *Int. Sym. on Signals, Circuits & Systems (ISSCS), session on Algorithms for 3D TOF-cameras*, volume 1, pages 121–124. IEEE, 2007.
- 5 T. Luhmann. *Nahbereichsphotogrammetrie*. Wichmann Verlag, 2. auflage edition, 2003.
- 6 T.D.A. Prasad, K. Hartmann, W. Weihs, S.E. Ghobadi, and A. Sluiter. First steps in enhancing 3D vision technique using 2D/3D sensors. In V. Chum, O.Franc, editor, *11th Computer Vision Winter Workshop 2006*, pages 82–86, 2006.
- 7 W. Reeves, D. Salesin, and R. Cook. Rendering antialiased shadows with depth maps. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 283–291, New York, NY, USA, 1987. ACM Press.
- 8 R. Reulke. Combination of distance data with high resolution images. In *Image Engineering and Vision Metrology (IEVM)*, 2006.
- 9 M. Segal, C. Korobkin, R. van Widenfelt, J. Foran, and P. Haeberli. Fast shadows and lighting effects using texture mapping. In *Proc. SIGGRAPH*, pages 249–252, 1992.
- 10 M. Stamminger and G. Drettakis. Perspective shadow maps. In *Proc. SIGGRAPH*, pages 557–562, 2002.
- 11 M. Strengert, M. Kraus, and T. Ertl. Pyramid methods in GPU-based image processing. In *Proc. Vision, Modeling and Visualization (VMV)*, pages 169–176, 2006.
- 12 Z. Xu, R. Schwarte, H. Heinol, B. Buxbaum, and T. Ringbeck. Smart pixel – photonic mixer device (PMD). In *Proc. Int. Conf. on Mechatron. & Machine Vision*, pages 259–264, 1998.