

Real time depth mapping performed on an autonomous stereo vision module

Jeroen Smit¹, Richard Kleihorst², Anteneh Abbo², Jan Meuleman¹ and Gerard van Willigenburg¹

¹ Wageningen University, Bornsesteeg 59, 6708 PD Wageningen, The Netherlands

² Philips Research Laboratories, Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands

E-mail: richard.kleihorst@philips.com

Abstract—**Keywords:** stereo vision, real-time, depth estimation

At the moment many applications such as robot navigation, obstacle avoidance, 3D user interfaces are limited in their capabilities due to limited availability of 3D measurement systems. Therefore, research in stereo imaging is a hot topic, because it offers a multilateral and robust way to reconstruct lost depth information. The few available state of the art stereo vision solutions have disadvantages such as cost, size, inflexibility, high power consumption and are often incapable of making depth maps in real time. In this publication a new low cost, autonomous, stereo vision module which computes a real time (30 fps) dense depth map in VGA format (640 * 480), is presented.

The module houses two uncalibrated off-the-shelf CMOS sensors, and a massive parallel programmable processor (Xetal). Due to the parallelism the processing performance of the actual stereo matching algorithm is highly efficient.

I. INTRODUCTION

The research on obtaining depth information of a real world scene is very active [1] [2] [3] [4] [5] [6], because applications where depth or distance clues could be applied are increasing. Depth estimation is one of the most vital visual tasks which humans can do almost effortlessly while for computers it is a difficult and challenging task. Nowadays most depth estimation systems don't perform depth maps in real time. However, these systems are in high demand, especially looking at the latest developments of consumer robotics, mobile telecommunications and 3D user interfaces applications. Some available systems are able to process depth estimation in real time, but these are expensive, power consuming and large, which is undesirable for the last mentioned applications. Therefore in this publication we want to show that it is possible to achieve real time depth mapping on a cheap, flexible, small and low power module.

Depth estimation by vision can be performed in several ways such as motion sensing, time-of-flight of (infrared light) and stereo vision:

1. Depth sensing by motion has an advantage that only one camera is required to obtain depth information from image sequences [4]. On the other hand, the system is rather limited because exact movement of the camera with respect to the scene or dimensions of the objects must be known. Therefore, this method is highly sensitive for an incorrect depth estimation. Also only the relative depth is measured and objects or camera have to move.
2. Since (infrared) light travels essentially at a constant speed, if one knows the elapsed travel time, the distance to a certain feature can be computed [7] [8]. In other words, it is possible to develop a 3D map of the surfaces in the scene. A big disadvantage of these systems is that they are sensitive for differences in object reflectances.
3. Stereo vision provides the most multilateral and robust way to reconstruct lost depth information. It relies on one fundamental finding: if two shots of a given scene are captured from two different viewpoints, then the resulting images will differ slightly due to the effect of a perspective projection. The correspondences of the stereo pair can be used effectively to reconstruct the three-dimensions of the scene, via a procedure known as stereo matching. The distance that the coordinates of an object in one image are shifted with respect to the same object in the other image, relative to its local coordinate system, is expressed as a disparity, and this is the fundamental measure required to reconstruct a scene.

In this research we have chosen stereo vision, because of above mentioned advantages. Besides that, a proper sensor setup and a hardware configuration is required to minimize the effect of several potential sources of errors, that makes locating correct image pairs difficult:

1. Occlusion problems can occur because projection takes place from different viewpoints. A symmetrical arrangement of more than two cameras, helps to reduce the effects of occlusion substantially [9]. However, these are more expensive and dimensions of these systems are inevitably greater.
2. Symmetries presented in a stereo pair gives multiple

potential correspondents for a given pixel and leads to ambiguous matches. Also, objects with a monochrome surface are only detected correctly at the edges. Projected imagery from a light source, such as a Digital Light Projector (DLP), can be used to artificially create a structure onto the objects [10]. On the other hand these artificial features are subject to change with variations of objects and environmental conditions.

3. Changes of intensity of the same 3D point in a stereo pair may occur due to the different viewing positions.
4. Accuracy is limited to the resolution and depth of input data.

Moreover, above mentioned difficulties are all highly dependent on lens distortions and sensor chips calibration. When these sensor systems are uncalibrated, they can distort 2D input data significantly [11]. Since, horizontal or vertical displacement, yaw, pitch, roll, radial and tangential lensdistortion in uncalibrated systems could occur, stereo matching objects with structure could become impossible and incorrect depth estimations and displacement errors occur with monochrome objects. Therefore many researchers calibrate their systems and can make use of the epipolarity constraint [9] [12] [3]. Because, accurate calibrating of sensor systems is a time consuming and elaborate assignment and therefore less suitable for consumer applications. We choose to challenge uncalibrated sensor systems in this paper. Initially we detect depth at the vertical edges of objects, regarding there is an epipolar line for vertical structures and subsequently fill in the objects in a second phase.

Besides the inevitable difficulties regarding the sensor setup and configuration, especially the processing required for real-time stereo matching [1] [2] [3] [12], has limited the design of small real-time stereo vision modules. In this paper we deal with these challenges and present a new low cost, autonomous, stereo vision module which computes a real-time (30 fps) dense depth map in VGA format (640 * 480). The power of this module lies in the fact that we make use of a Single Instruction Multiple Data (SIMD) processor and specially designed and optimized algorithms.

The contents of this paper are reflected in the tasks: In Section II the choice of hardware is motivated. The used stereo vision algorithms and its performance are described in Section III and Section IV respectively. The conclusions and future work are discussed in Section V.

II. HARDWARE ARCHITECTURE

As mentioned before, stereo matching is very cumbersome for general purpose computers and processors. Not only because of the computational effort, but also

because of the data rates and electrical power involved. Therefore we used the Xetal chip [13] in our stereo vision module. Besides the Xetal chip, the stereo vision module houses two uncalibrated off-the-shelf VGA (640×480 pixels) 10-bits RGB CMOS image sensors, which results in a powerful stereo-sensor-DSP combination.

The Xetal chip (Figure I) is designed especially for high-performance pixel processing in imaging applications. Xetal is a programmable digital video IC with a massive parallel processor.

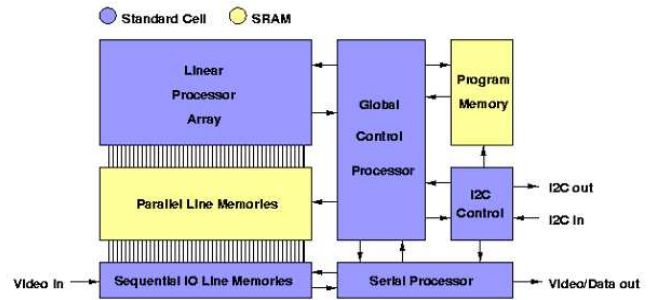


Figure I: Xetal Design

This chip consists of a Linear Processor Array (LPA) with 320 Processing Elements (PE). Each PE houses an Arithmetic Logic Unit (ALU) and a Multiply Accumulator (MAC). The input data for the PE is received through a 10 bits bus. Each PE is assigned to two image columns with the possibility to additionally access two left and two right neighboring pixels. This allows for right and

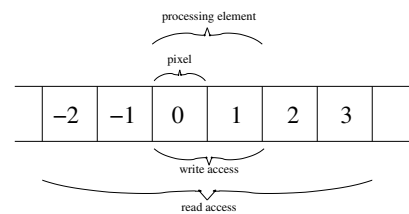


Figure II: Processing Element Layout

left direct addressing of six pixels (Figure II). Since all processor elements share the same decoding logic it is possible to simultaneously execute one LPA instruction on all 320 elements. The LPA performs all the DSP operations on the data stored in the line memories. The Global Controller (GC) performs tasks such as conditional execution, iteration and synchronization. The statistical computations are performed by the serial processor and are used to control the sensor parameters or to update filter coefficients.

The image or video input data is a VGA size frame (matrix) with up to 10-bit digitized signals at a maxi-

imum rate of 30 frames/second. Xetal can not receive two different 10 bits signals at the same time. Therefore hardware mixes the two sensor input signals of 10 bits into one signal of 10 bits. Xetal has 16 line memories for temporary storage and 4 sequential line memories for input-output purposes. Each line memory holds 640-pixels at 10-bit resolution. The interface to the video input and output is achieved via a single-channel-input and a three channel-output port. The I2C interface is used to download program code to the chip.

Because of the fully parallel architecture, high performances and data-rates can be achieved for a modest power consumption. The power consumption mainly depends on the image parameters, i.e., the number of rows per frame, the frame rate, and on the program that runs on the parallel processor. This can go down to 30 mW for simple applications such as a digital camera for video conferencing. For more sophisticated applications, such as stereo depth mapping it can go up to 200 mW.

III. STEREO VISION ALGORITHM OUTLINE

The outline of the complete stereo vision algorithm is shown in Figure III.

A. Separating

Since the hardware mixed the two 10 bits *RGB* input signals into one 10 bits *RGB* input signal, the signal must be separated to obtain the left and right image. The upper 5 bits of the mixed signal belong to the left sensor and the lower 5 bits to the right sensor.

B. Convert *RGB* into *Y*

It is not desirable to perform a matching procedure in all three color spaces [14]. Therefore the *RGB* signal is converted into the luminance (*Y*). In this research we perform the stereo matching algorithm only on the intensity (*Y*) of the input signals.

C. Low-pass filtering

A low-pass filter is used to reduce the effects of noise and to cancel out large variations from pixel to pixel, because we are not relying on the epipolar line principle. This routine performs a 5×1 low-pass filter with an equal weight of 0.2 and not for example a 5×5 .

D. Stereo matching

The fundamental problem in stereo vision is that of locating corresponding or matching points in the two images. We used a 6 pixels wide area based matching algorithm, which is characterized by the fact that it compares

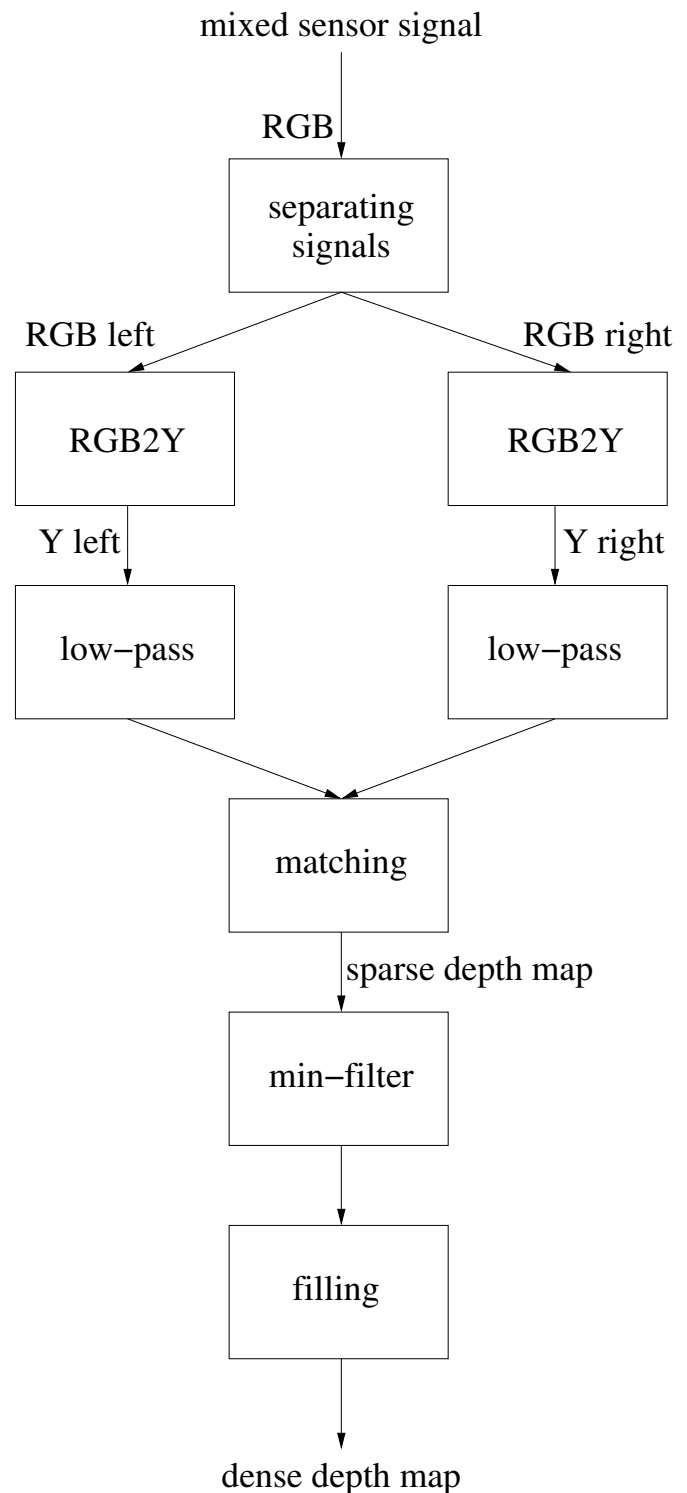


Figure III: Algorithm outline

windows of pixel values in the two images, in order to find the best match.

We compute the Sum of Absolute Differences (*SAD*) (1) for each window between the left and right image by shifting one image line a preset number of pixels. Because we look for dominant vertical features, it is not necessary that one has to search for a certain pixel from

the left image in the complete right image. A point to be matched essentially, becomes the centre of the window of pixels, which is compared with similarly sized windows in the other image. Matching measures were used to provide a numerical measure of the similarity between a window of pixels in one image and a window in another image, and hence are used to determine the optimum match. There are two simple matching measures which are suitable for hardware implementation, the *SAD* (1) and Zero mean Sum of Squared Differences (*SSD*). Previous research showed that the introduced complexity in the *SSD* matching measure does not give any significant improvement in matching quality [15].

$$SAD = \sum_{i=1}^N |Y_i^r - Y_i^l|, \quad (1)$$

where Y_i^r and Y_i^l are respectively the intensity (Y) of the right and left pixel.

Since the processor is parallel, each computation is performed at the same time for each pixel on the current image line. When the actual *SAD* is smaller than the previous stored smallest *SAD* at the specific location, the value will be overwritten by the actual *SAD* and corresponding number of pixel shifts. After all, the number of pixel shifts at the lowest *SAD* per pixel, represents the depth of an object. Due to this direct linear relation, absolute depth information can be easily derived by triangulation methods.

E. Minimum filter

A minimum (gray value erosion) filter is used to reduce the effects of small pixel regions which contain false depth information. The minimum filter replaces the current depth value of a pixel by the minimum depth value within a window of 3×3 pixels.

F. Object filling

Optimal stereo matches are only found at vertical edges of or in the objects. The found sparse depth map has to be filled in. Filling of objects in a scene is a difficult task for a PC or processor. Some complex methods exist for object filling [16].

IV. MEASUREMENTS AND PERFORMANCE

A. Stereo matching performance

We observed that the uncalibrated CMOS sensors especially had a significant horizontal offset (of several lines). Since we are making use of a virtual epipolarity constraint only vertical edges of monochrome objects can

Table I: Relative number of instructions for the stereo matching routines per line time (in %)

task	load
separating sensor signals	8%
RGB to Y conversion	5%
low-pass filtering	3%
stereo matching	64%
minimum filter	3%
object filling	10%
control	7%

be correctly matched. We also experienced that a depth resolution of only 5 bits is limiting the stereo matching performance of objects. This is due to the fact that minor real world color differences have been canceled out by the 10 to 5 bits conversion, which causes inaccurate intensity levels.

Our disparity search range is 19 pixels, which results in our specific setup that we can detect objects as close as 1 m to the stereo sensors, independent on the size of the object. The disparity range is limited because of the amount of instructions per line time. The actual stereo matching routine requires more than half of the available instructions per line time (Table I).

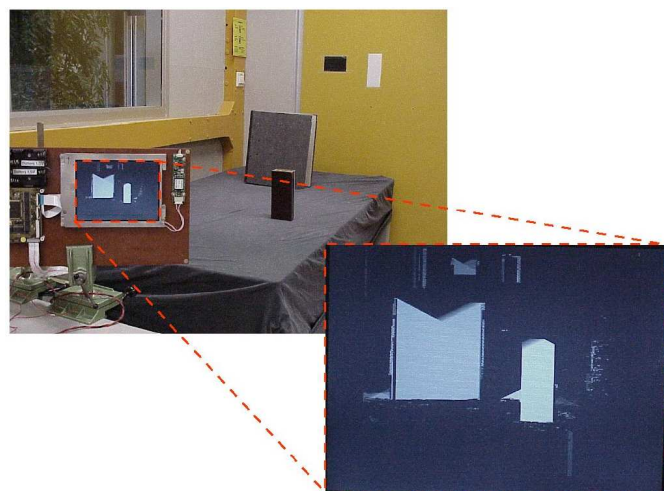


Figure IV: Setup overview: The larger figure shows the setup while the screenshot shows the real time depth map. From the intensity of the pixels the distance to the camera can be obtained. For instance, the larger (and further) is slightly darker than the smaller object, which is closer. The two objects at the wall are even further away, therefore even darker.

V. CONCLUSIONS AND FUTURE WORK

Depth estimation is becoming an important application in several consumer applications. However, only few available solutions exist and these have disadvantages such as cost, size, inflexibility, high power consumption and are often incapable of making depth maps in real-time.

Till now, especially the processing required for complete real time stereo matching prohibited integration of the whole application into a flexible, small sized, cheap, autonomous, low-power module. This publication shows that this integration can be achieved by:

- Making an adequate choice for the processing architecture,
- Designing and optimizing algorithms specific for the selected architecture and purpose.

As a result we present a low cost, autonomous, stereo vision module which computes a real-time (30 fps) dense depth map in VGA format (640 * 480). Despite some hardware limitations such as limited number of intensity levels and uncalibrated sensor systems, very promising results have been shown. Applications such as object avoidance, navigation, 3D user interfaces and detecting objects of interest can take the benefits from this new module.

Future research will focus on further development of the algorithms, e.g. improving the stereo matching and object filling algorithms. However, adjustments on the algorithms are dependent on the quality of the input data from the image sensors and especially the available processing architecture. Action will be taken to obtain more bits from the sensors, increase the disparity range and to see the effects of low cost calibration.

REFERENCES

- [1] Y. Ruichek, "A hierarchical neural stereo matching approach for real-time obstacle detection," in *IEEE Intelligent Transportation Systems Proceedings*, October 2003.
- [2] M. Hariti, Y. Ruichek, and A. Koukam, "A voting stereo matching method for real-time obstacle detection," in *IEEE Robotics and Automation Proceedings, ICRA*, vol. 2, September 2003.
- [3] S. Kagami, K. Okada, M. Inaba, and H. Inoue, "Design and implementation of onbody real-time depthmap generation system," in *IEEE Robotics and Automation Proceedings, ICRA*, vol. 2, April 2003.
- [4] Y. L. Murphey, J. Chen, J. Crossman, J. Zhang, P. Richardson, and L. Sieh, "Depthfinder, a real-time depth detection system for aided driving," in *IEEE Intelligent Vehicles Symposium Proceedings*, October 2000.
- [5] S. K. Park and I. S. Kweon, "Robust and direct estimation of 3-d motion and scene depth from stereo image sequences," *Pattern Recognition*, vol. 32, no. 9, pp. 1713–1728, 2001.
- [6] M. I. Fanany and I. Kumazawa, "A neural network for recovering 3d shape from erroneous and few depth maps of shaded images," *Pattern Recognition Letters*, vol. 25, no. 4, pp. 377–389, 2004.
- [7] Canesta, "Canesta infrared sensor chips." <http://www.canesta.com>, 2004.
- [8] A. B. Robert Lange, Peter Seitz and S. Lautherman, "Demodulation pixels in ccd and cmos technologies for time-of-flight ranging," in *IEEE Virtual Reality Proceedings/IST/SPIE International Symposium on Electronic Imaging*, January 2000.
- [9] T. Kanade, A. Yoshida, K. Oda, H. Kano, and M. Tanaka, "A stereo-machine for video-rate dense depth mapping and its new applications," in *IEEE Computer Vision and Pattern Recognition Conference*, vol. 24, June 1996.
- [10] W. B. Seales, G. Welch, and C. O. Jaynes, "Real-time depth warping for 3-d scene reconstruction," in *IEEE Aerospace Conference Proceedings*, vol. 3, March 1999.
- [11] J. Smit, L. G. van Willigenburg, and J. Meuleman, "Accurate calibration and 3d data recovery based on physical camera models including lens distortions," Master's thesis, Wageningen University, The Netherlands, 2004.
- [12] S. B. Goldberg, M. W. Maimone, and L. Matthies, "Stereo vision and rover navigation software for planetary exploration," in *IEEE Aerospace Conference Proceedings*, vol. 5, March 2002.
- [13] A. Abbo and R. Kleihorst, *Xetal Software Framework Programming Guidelines*. Philips Research Laboratories, NatLab, 2001.
- [14] J. Ninot, "Real time depth estimation from binocular cameras using the xetal processor," Technical Report PR-TN-2003/00797, Philips Research, November 2003.
- [15] J. Banks, M. Bennamoun, and P. Corke, "Non-parametric techniques for fast and robust stereo matching," in *IEEE Speech and Image Technologies for Computing and Telecommunications, TENCON*, December 1997.
- [16] M. Martin, M. Martin, C. Alberola-López, and J. Ruiz-Alzola, "A topology based filling algorithm," *Computers & Graphics*, vol. 25, no. 3, pp. 493–509, 2001.