59th ILMENAU SCIENTIFIC COLLOQUIUM
Technische Universität Ilmenau, 11 – 15 September 2017
URN: urn:nbn:de:gbv:ilm1-2017iwk-036:6

3D Image Acquisition and Processing with high continuous Data Throughput for Human-Machine-Interaction and Adaptive Manufacturing

[1]Susann Winkler, [1]Maik Rosenberger, [2]Daniel Höhne, [2]Christoph Munkelt,
[3]Chang Liu, [1,2]Gunther Notni

[1]Technische Universität Ilmenau, Department of Mechanical Engineering,
Group of Quality Assurance and Industrial Image Processing
[2]Fraunhofer Institute for Applied Optics and Precision Engineering
[3]Friedrich-Schiller-University Jena, Institute for Applied Physics

ABSTRACT

Many applications in industrial environment are able to detect structures in measurement volumes from macroscopic to microscopic range. One way to process the resulting image data and to calculate three-dimensional (3D) images is the use of active stereo vision technology. In this context, one of the main challenges is to deal with the permanently increasing amount of data. This paper aims to describes methods for handling the required data throughput for 3D image acquisition in active stereo vision systems. Thus, the main focus is on implementing the steps of the image processing chain on re-configurable hardware. Among other things, this includes the pre-processing step with the correction of distortion and rectification of incoming image data. Therefore, the approach uses the offline pre-calculation of rectification maps. Furthermore, with the aid of the rectified maps, each image is directly rectified during the image acquisition. Afterwards, an FPGA and GPU-based approach is selected for optimal performance of stereo matching and 3D point calculation.

Index Terms *– three-dimensional measurement, high-speed projection, re-configurable hardware*

1. INTRODUCTION

Three-dimensional reconstruction using active stereo vision is one of the most important technologies for 3D image acquisition in an industrial environment. This enables major acquisition rates with simultaneously high resolution and high measurement accuracy. The simultaneous acquisition and processing of 3D data from a plurality of sensors is a current research topic. [1] Three aspects are always involved: Pattern projection/data acquisition, 3D calculation and evaluation/representation.
Some examples of the current state of the art are listed in table I. This illustrates that the frame rate depends on the number of pixels, because the frame rate decreases with increasing number of pixels. For example, AHLBERG ET AL. achieves with 1920 x 1080 pixels (~ 2 megapixels) a frame rate of 60 frames per seconds (fps), whereas at 3648 x 2736 pixels (~ 10 megapixels) only 15 fps can be realized. [2] This demonstrates, that the data acquisition is possible with high data rates about a range of 1 GB/s.
Furthermore, the 3D calculation and evaluation have to be considered, because both are difficult to realize [3]. Currently, these are only possible with limited data throughput. This results from the major data volume and the complexity of the necessary calculations. To increase the data rates, specialized hardware has to be used for calculation. For this purpose, technology

components such as re-configurable hardware can be utilized (e.g. FPGA). As displayed in table I, hardware platforms from Xilinx and Altera are used.
It is noticeable that in current research projects, none known statements on the picture quality in general and the latency of the rectification are carried out. Mainly, the results for the search of correspondence points are related to the number of pixels. In addition, it is unclear which internal computation accuracy was used.

TABLE I. AN EXTRACT OF CURRENT RESEARCH PARAMETERS IN THE FIELD OF IMAGE PROCESSING WITH THE USE OF RE-CONFIGURABLE HARDWARE

Researcher	Image size [pixel]	Disparity [pixel]	Frame rate [fps]	Algorithm	Platform
Chang et al. [4]	384 x 288	16	50	SAD*	TMS320C6414T-1000 platform
Murphy et al. [5]	320 x 240	20	40 (to 150)	CT*	Xilinx Spartan-3 XC3S2000
Kalarot et al. [6]	1024 x 768 (256 x 256)	n.s.	30 (450)	SDPS*	Altera Stratix III
C. Banz. et al. [7]	640 x 480	128	30	SGM* Rank transformation	Xilinx Virtex-5 LX220T-1
Greisen et al. [8]	1920 x 1080	n.s.	30	ZSAD*	Altera Stratix III
Mattoccia [9]	640 x 480	32	30	SGM*	Spartan 6 LX family – models 45, 75, 100
Ahlberg et al. [2]	3648 x 2736 (10MP) 1920 x 1080 (2 MP)	n.s.	15 60 (12bit)	SAD*	Xilinx Zynq 7020

*SAD - Sum of absolute differences; CT - census transformation (incl. Hamming distance); SDPS - Symmetric Dynamic Programming Stereo algorithm; ZSAD - Zero-mean Sum of Absolute Differences; SGM - Semi-Global Matching

2. THEORETICAL BACKGROUND

There are many influencing factors from the image recording to the final 3D points. Among others, the image sensors and their structure have a considerable influence. The optimal alignment of two cameras is their positioning along a base line with parallel optical axes. In this arrangement, the pixel correspondences are located along a horizontal line in the other camera image (epipolar line). For limiting the search of correspondence to a search problem along one line (1D), the incoming images have to be undistorted. This allows to correct errors that are caused by non-linear lens effects (radial distortion) and the relative positioning offsets of the camera (tangential distortion). In addition, errors caused by non-optimal alignment of the cameras have to be removed (rectification). A further aspect which is explained below are possible projection methods of measurement objects. Depending on this, a possibility for calculating correspondence points is explained.

2.1 Lens distortion

In reality a lens system various from a perfect central projection. Barrel and pincushion distortion develop through the physical inhomogeneity of the camera sensor (Figure 1 (right)). [10] The radial lens distortion of the normalized values x, y of $x_{original}$ and $y_{original}$

$$x = \frac{x_{original} - ppx}{fx} \quad \text{and} \quad y = \frac{x_{original} - ppx}{fx} \tag{1}$$

have to be removed in order to eliminate this distortion. For converting the original values in normalized values the intrinsic camera parameters like the principal point (ppx, ppy) and the focal length (fx, fy) are needed. Therefore, an offline performed camera calibration is necessary. [11] To correct the radial lens distortion of the image points a scaling factor dr is

$$dr = k1 \cdot r + k2 \cdot r^2 + k3 \cdot r^3 \tag{2}$$

calculated with the squared normalized radius of the distorted image points (x, y)

$$r = x^2 + y^2. \tag{3}$$

A scaling factor for tangential distortion dtx and dty is defined as

$$\begin{aligned} dtx &= 2 \cdot p1 \cdot x \cdot y + p2 \cdot (r + 2x^2) \\ dty &= p1 \cdot (r + 2y^2) + 2 \cdot p2 \cdot x \cdot y. \end{aligned} \tag{4}$$

To compute the corrected image points $x_{undistort}$ and $y_{undistort}$ the following calculation with both scaling factors is applied as follows

$$\begin{aligned} x_{undistort} &= x + dr \cdot x + dtx\ , \\ y_{undistort} &= y + dr \cdot y + dty. \end{aligned} \tag{5}$$

After that, the values of $x_{undistort}$ and $y_{undistort}$ have to be rescaled by the focal length and the principal point. The basis for the computation are the camera parameters including the radial distortion coefficients $k1, k2, k3$ and the tangential distortion coefficients $p1, p2$.

2.2 Rectification

A stereo vision system is often a setup using two cameras. There are two possible spatial arrangements. On the one hand, there is the parallel alignment of optical axes of both cameras and on the other hand the alignment to a convergence point (Figure 1 (left)). Each converged camera arrangement can be converted into a parallel axis system by virtual rotation of the cameras, which is also called rectification. Normally the convergent camera arrangement is used in practice. [12] One possibility to perform the rectification is the usage of homographies T_{Left} and T_{right}. These are two 3 x 3 matrices, which allow the projective transformation of the individual image points. To calculate homographies the fundamental matrix F and corresponding points between the images are required. The fundamental matrix can be determined as described in SCHREER [12].
First, to carry out the projective transformation, the image points from the 2D space $x_{original}$ and $y_{original}$ have be converted into three dimensional homogeneous coordinates $[x_{original}\ y_{original}\ 1]$. These homogenous coordinates have to be transformed using the homographies to correct the image points.

$$\begin{bmatrix} X_{corr} \\ Y_{corr} \\ Z_{corr} \end{bmatrix} = T_{left/right} \cdot \begin{bmatrix} x_{original} \\ y_{original} \\ 1 \end{bmatrix} \tag{6}$$

After the correction of the homogeneous coordinates a rescaling into the rectified image coordinates x_{rect} and y_{rect} have to be done.

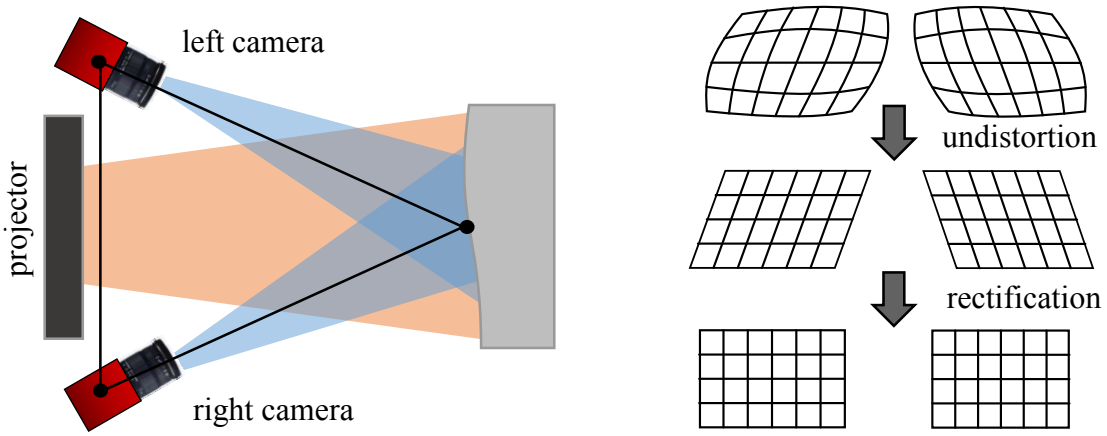

Figure 1. Schematic representation of a convergent camera setup (left) and the process of undistortion and rectification (right)

2.3 Active pattern projection

Stereo vision can be distinguished in passive and active methods. Active methods include those that actively illuminate the scene to be recorded or manipulate its illumination. In a productive and an industrial environment, it can be assumed that the measurement objects do not have enough information (e.g. texture) to apply passive methods successfully. For this reason, active methods are preferred in this research project. In the case of active illumination, the pattern is consecutively projected onto the measuring object. By this method, spatial coordinates can be assigned to points in the images of the cameras. This method offers the advantages of high accuracy and extremely high frame rates. For example, HEIST ET AL. presents an approach, where an array projector enables frame rates of several tens of kilohertz [13].
Since one pattern is generally for coding, a suitable sequence length of patterns is used. Generally, six patterns are enough, but a sequence of nine is a good compromise between accuracy and measurement time. [14] In the used scenario aperiodic sinusoidal fringes are being utilized. This projection method is described in detail in HEIST ET AL. [13].

2.4 Normalized cross-correlation (NCC)

One important aspect of calculating 3D values from image data is searching for correspondence points. The stereo correspondence problem is solved by using the normalized cross-correlation (NCC). Therefore, a stack of intensity values for one pixel of the left camera is taken and an approximate equal is searched in the right camera (Figure 2). The advantage of the aperiodic sinusoidal fringe patterns is that no knowledge about the phase-(shift) requirement is necessary. Furthermore, low influence of variability of fringe contrast or modulation are advantageous. Additionally, no synchronization of the projector is necessary. [10, 14] The basis to calculate the corresponding points is the computation of the correlation coefficient. In detail this is defined as

$$cc_{IL,IR} = \frac{\sum_{i=n}(IL_i - \overline{IL})(IR_i - \overline{IR})}{\sqrt{[\sum_{i=n}(IL_i - \overline{IL})^2]\,[\sum_{i=n}(IR_i - IR)^2]}}. \quad (7)$$

IL and IR are representing the pixel values of the images of the left and right camera. The nominator consists of the empirical covariance including the mean value of all appropriates input values. The denominator is composed of the standard deviation of the empirical variance.

- IL pixel value of image 1
- IR pixel value of image 2
- $\overline{IL}$ and $\overline{IR}$ mean value
- i image number of the sequence
- Empirical covariance: $\sum_{i=n}(IL_i - \overline{IL})(IR_i - \overline{IR})$
- Empirical variance: $\sum_{i=n}(IL_i - \overline{IL})^2$
- Standard Deviation: $\sqrt{\sum_{i=n}(IL_i - \overline{IL})^2}$

The value of the coefficient ranges between

$$-1 \leq cc_{IL,IR} \leq 1. \tag{8}$$

A blockwise correlation calculation is the correlation between a template (block) and an image excerpt. If the value 1 is accepted, then template and image excerpts agree. There is no correlation if the result is 0. The template is negative to the image excerpt if the value is -1. [10, 14] The essential calculations include determining the mean value for elimination the ambient light. In addition, the standard deviation is used for removing the reflection (e.g. reflecting surfaces). [15]

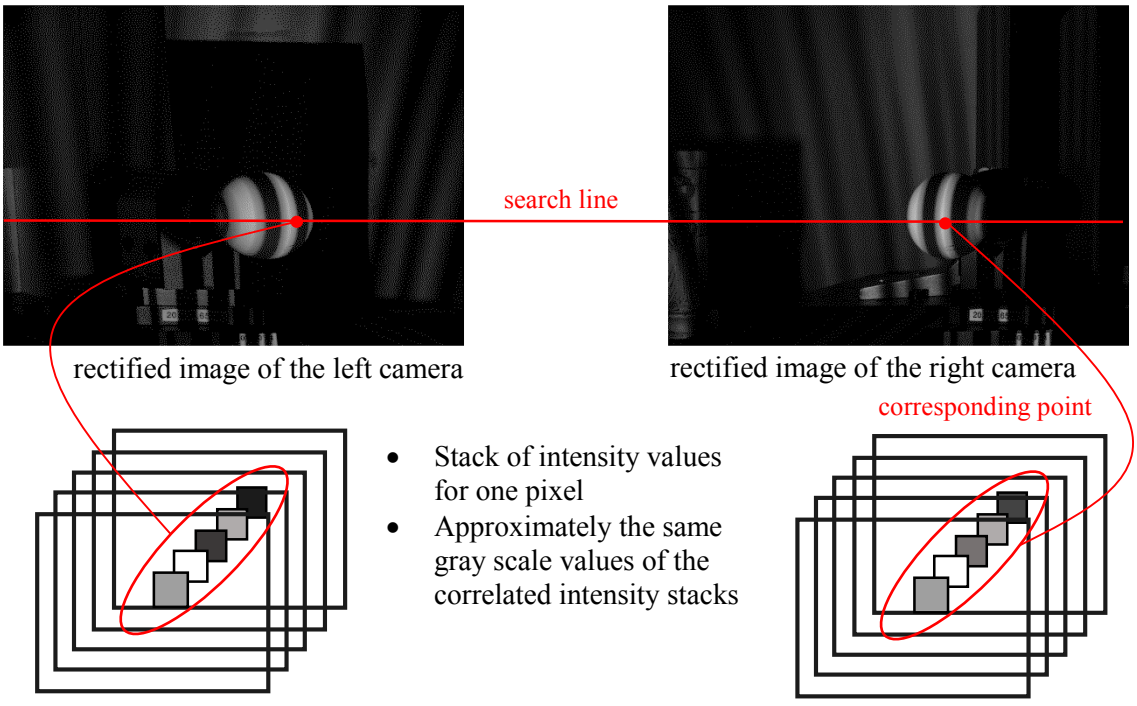

Figure 2. Rectified image of the left and right camera with an exemplary signed corresponding point on search line

3. IMAGE PROCESSING WORKFLOW

3.1 System Architecture

The presented image processing system is designed as a heterogeneous sensor network which including four cameras and two projectors. In general, a sensor network consists of N cameras and M projectors. One important prerequisite is that $N > M$. Up to now, single-sensor approaches have been replaced by the heterogeneous sensor network. Current 3D sensor networks with active pattern projection have a fixed arrangement between the camera pairs and the projector. [16, 17] This fixed arrangement is broken up with the heterogeneous sensor network and enables a highly dense measurement of complex geometries.
In this context the projection system is another important aspect. In order to illuminate very large objects with a minimal number of projectors, a high light intensity (several 10,000 lumen) and a large depth of field are necessary. However, conventional optics lead to aberration. One possibility to achieve the stated reasons is the "Scheimpflug" arrangement of the projection optics. [18] The exact conception of the projector design will not be discussed here, but is similar to ZWICK ET AL. [19]
Another important facet to be considered is the intelligent data reduction of the extremely high data throughput. The data from image acquisition has to be transmitted to a general processing unit (Figure 3). In this case, attention will be paid to the connection of signal transmitters and re-configurable computation technology (i.e. FPGA). Furthermore, a hardware and software integration is used to reduce latencies and data volumes. The individual processing steps in the processing unit are described in the following.

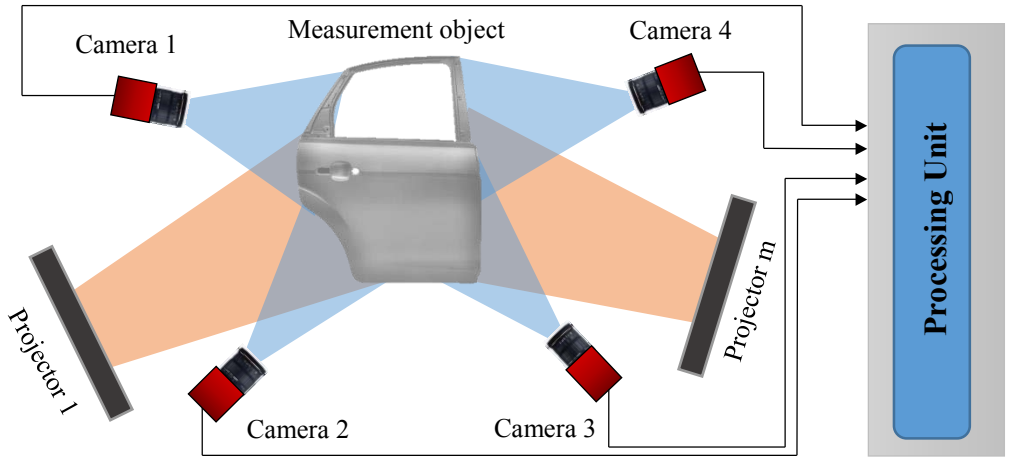

Figure 3. Illustration of a theoretical sensor network setup

3.2 Pre-calculation of undistort and rectification maps

Before starting the online measurement, the camera calibration has to be performed. As a result, the camera parameters including intrinsic and extrinsic parameters are generated. During this process of calibration, the rectification maps can be calculated directly. A rectification map includes x- and y-offset values of an undistorted and rectified image of one camera. Without any compression the matrix has a *n x m*-matrix of the source image size. These maps get used for the inverse mapping algorithm, where for each pixel in the result (corrected and rectified) image the corresponding points in the source image were displayed (Figure 7). For calculation, MATLAB or the OpenCV library can be used. The basis of the following described workflow are the tools of the MATLAB Image Processing Toolbox.

First the fundamental matrix is calculated. This describes the relationship between corresponding points of two image views. The program sequence in MATLAB is illustrated as an example in Figure 4. For this, some images of a checkerboard pattern (in this case 20 images) are needed. The *checkCheckerboardPoints* function automatically determines the intersections in the checkerboard pattern. The subsequent camera calibration results, among other things, the stereo parameters which contains the fundamental matrix.

```
FileNames1 = {'path_1', ..., 'path_n'};
FileNames2 = {'path_2', ..., 'path_m'};

[imagePoints, boardSize, imagesUsed] = detectCheckerboardPoints(FileNames1, FileNames2);
squareSize = 5;
worldPoints = generateCheckerboardPoints(boardSize, squareSize);

[stereoParams, pairsUsed, estimationErrors] = estimateCameraParameters(imagePoints, worldPoints, ...
   'EstimateSkew', true, 'EstimateTangentialDistortion', true, ...
   'NumRadialDistortionCoefficients', 3, 'WorldUnits', 'mm', ...
   'InitialIntrinsicMatrix', [ ], 'InitialRadialDistortion', [ ]);

F = stereoParams.FundamentalMatrix;
```

Figure 4. MATLAB code used to calculate the fundamental matrix

Furthermore, marked points are detected and matched to one another. To detect the points, pre-implemented feature detectors can be used in MATLAB. With the help of a tracker, common points can be determined. This results in two matrices with their respective matched points. The program sequence is shown in Figure 5.

```
imagePoint1 = detectMinEigenFeatures(FileNames1_ImageN, 'MinQuality', 0.001);

tracker = vision.PointTracker('MaxBidirectionalError', 1, 'NumPyramidLevels', 5);

imagePoint1 = imagePoint1.Location;
initialize(tracker, imagePoint1, FileNames1_ImageN);

[imagePoint2, validIdx] = step(tracker, FileNames2_ImageM);
matchedPoints1 = imagePoint1(validIdx, :);
matchedPoints2 = imagePoint2(validIdx, :);
```

Figure 5. MATLAB code used to calculate the matched points for the left and right image

As a result, the function *estimateUncalibratedRectification* can be used to calculate the projective transformation matrix of the left and right camera (Figure 6). This function does not require either intrinsic or extrinsic camera parameters. Because of the convergent arrangement of the cameras the camera calibration app of the MATLAB toolbox cannot be use. This tool can only be applied on a parallel camera arrangement. The input arguments for estimating uncalibrated rectification are the fundamental matrix F, the matched points of the right and left camera and the image size. Thus, the output arguments are two 3 x 3 projective transformation matrices of the left T_{left} and right camera T_{right}. With these projective matrices any image can be undistorted and rectified. The prerequisite is that nothing will change in the camera setup.

```
[Tleft, Tright] = estimateUncalibratedRectification(F, matchedPoints1, matchedPoints2, …
          size(FileNames1_ImageN));
```

Figure 6. MATLAB code used to calculate the projective transformation matrix for the left and right image

With the aid of these projective transformation matrices the rectification maps can be calculated. For this purpose, each image point is converted into normalized coordinates according to (1). Then calculate the distorted normalized coordinates by the scaling factors of the radial (2) and tangential (4) distortions. This both scaling factors lead to the undistorted normalized coordinates (5). Furthermore, the image points are rectified after the conversion into homogeneous coordinates by means of the transformation matrix $T_{left/right}$ (6).

This results of this process are two maps for each camera. However, the rectification maps are necessary to perform the rectification in the FPGA. In OpenCV a similar process where the function *initUndistortRectifyMap* can be used to get a map of the x and y offset. [20, 21] At least, a bilinear interpolation have to be performed to get a rectified image.

At this point, it has to be noted that for each possible camera combination in the sensor network the rectification maps have to be calculated and stored. In this case, there are six possible camera combinations in a sensor network with four cameras, with a total of 12 maps.

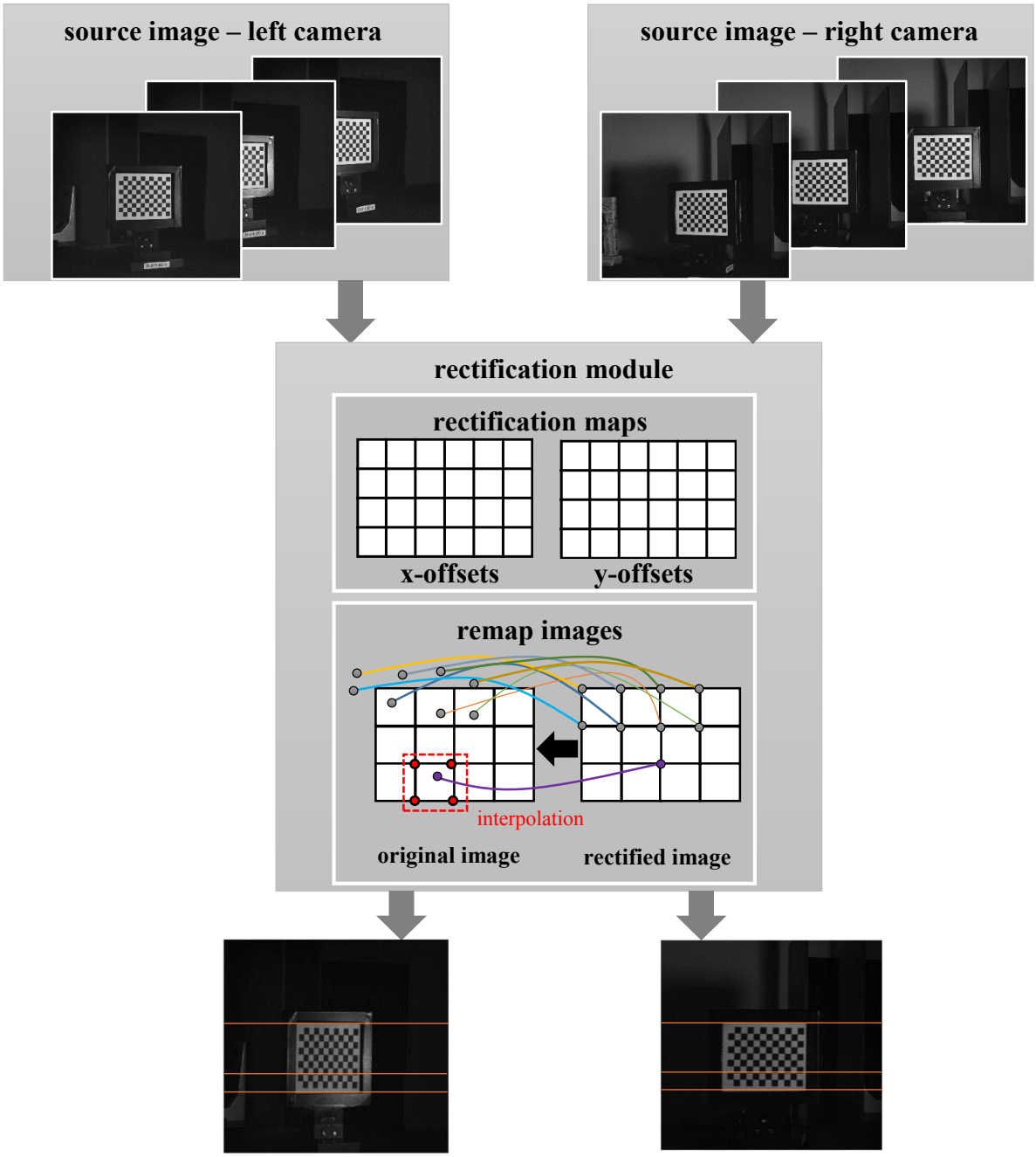

Figure 7. Illustration of the undistortion and rectification process

Stereo processing pipeline

The computation of the rectification maps is done offline. The four maps are stored in FPGA-internal Block RAM. After the camera calibration had been performed offline, the actual data acquisition can be done. Therefor several projectors project patterns on the measurement object. Using the "Scheimpflug"-principle in combination with freeform optics, very large surfaces (ranged in several meters) can be completely illuminated. The next step in the process chain includes the image acquisition and transmission of acquired data into the memory. In this context the limited internal Block RAM (BRAM) or an external DDR memory for larger amounts of data can be used. Due to the amount of data an external memory is necessary. Low-Voltage-Differential-Signaling (LVDS) is used for transmitting the data from the camera sensors to the embedded platform. Before storing the data in memory the image preprocessing has to be performed on the embedded platform. One part of the pre-processing is the definition of multiple image regions as areas of interests (AOIs). In this case selectively AOIs have to be considered more closely. The reason is that only certain areas should be transmitted in high resolution analogous to the human foveal vision. In these areas an extremely high resolution (< 0.1 mm) is required. In contrast, peripheral areas in the range of several meters can be displayed in low resolution.
Another important part of preprocessing is the implementation of rectification. Therefor the previously stored rectification maps were read from memory. The incoming raw images get remapped with the help of the x- and y-offset values. To save computation time a parallel calculation of similar calculation steps has to be realized. This step is performed on the programmable logic (PL). For eliminating outliers and fill data gaps, filters (e.g. median filter) can be used. After the images are undistorted and rectified, the search of correspondence is reduced to a 1D problem. That means, the search area can be limited to the epipolar lines. In this way it is not necessary to look for correspondences in the entire image.

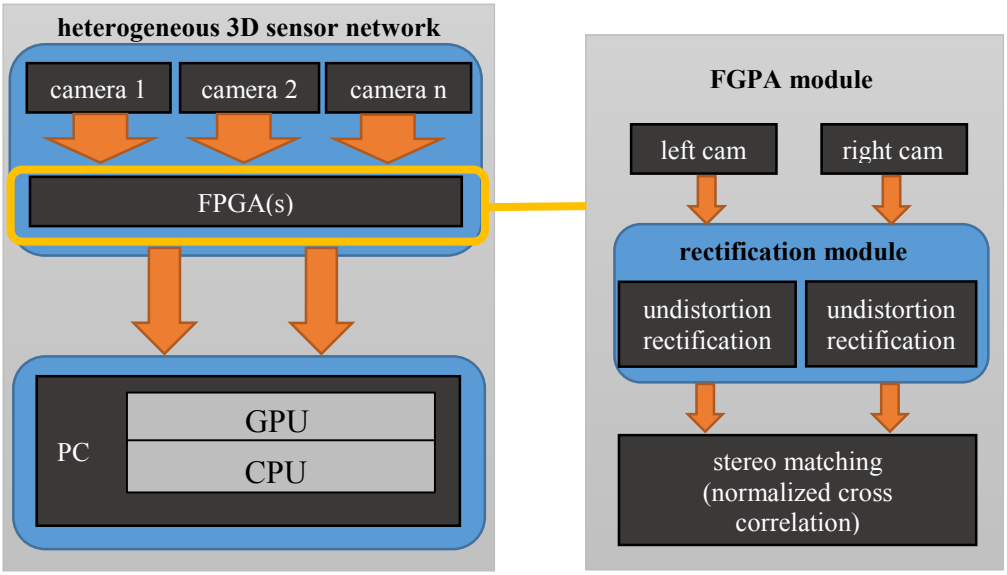

Figure 8. Dataflow in a theoretical 3D sensor network with separate dataflow in one FPGA module

One approach for stereo matching is the normalized cross-correlation as described in section 2.4. When implemented in the FPGA, the calculation in equation (7) is separated into its individual calculation steps. This procedure is necessary because each calculation requires special intellectual property (IP) cores in the hardware description. With the help of these IP cores arithmetic operations such as square root, multiplication or division can be implemented. The decomposition of the calculation of the NCC into its individual components results in ten

calculation steps, which have to be implemented. A total of four different IP cores are involved, which are used to perform mathematical operations.
This includes the Adder/Subtracter (Add/Sub) IP core, which performs the addition or subtraction of two input values. [22] The CORDIC IP core is used for square root. [23] In addition, DSP slices are used to perform the multiplication of the two roots. In addition, the Divider Generator performs the division. [24] An overview of the individual operations is illustrated in Table II.

TABLE II. COMPONENTS INVOLVED IN THE CALCULATION OF THE NORMALIZED CROSS-CORRELATION

Calculation step	**Mean Value $\overline{IL}/\overline{IR}$**	**$(IL_i - \overline{IL})$**	**$(IL_i - \overline{IL})$**	**Emp. covariance**	**Emp. Variance $\overline{IL}/\overline{IR}$**	**Square root**	**Multi-plication**	**Division**
IP core	Add/ Sub	Add/Sub	Add/Sub	Add/ Sub	Add/Sub DSP Slices	CORDIC	DSP Slices	Divider Gen.

First, the hardware platform Xilinx Zynq 7020 was selected. For more information, see section 4. A decisive factor when executing a program run on an FPGA is the resource utilization. These hardware resources allow the implementation of the calculations. For this purpose, components such as look-up-tables (LUT), flip-flops and DSP slices have to be considered. An overview of the total number of existing resources is illustrated in Table III. The Zynq Ultrascale+^{TM} platform was used as a comparison. This is currently the most powerful and most popular hardware platform of the Zynq family. The values reflect the fact that the ZU9EG has more than five times as many resources as the Zynq 7020.

TABLE III. OVERVIEW OF AN EXTRACT OF THE RESSOURCES OF THE XILINX Z-7020, ZU3EG AND ZU9EG

Platform		**Resources**		
		LUT	**Flip-Flops**	**DSP Slices**
Zynq-7000 SoC	Z-7020	53200	106400	220
Zynq UltraScale+™ MPSoC	ZU3EG	70560	141120	360
	ZU9EG	274080	548160	2520

The basis for the following results are recordings of two cameras with a camera resolution of one megapixel with a sequence length of ten images. First, the resource utilization was considered for one correlation calculation. This provides an overview of the complexity of the calculation. In this content, one correlation calculation is only the correlation between a pixel stack of one image with a pixel stack of the other image. However, in practice this step has to be repeated over the entire image width. Since rectified images are present, it is a 1D search problem.
As mentioned, 10 calculation steps are required for the calculation of one correlation, which are realized by four different IP cores. Each IP core has a different resource utilization. The Add/Sub is used most frequently (seven out of ten times). Despite the frequency of application within the calculation, this IP core has the least influence on the system utilization. The CORDIC IP core (square root) is more significant. A calculation operation requires nearly 2428 LUTs and 1328 flip-flops. In addition, the divider generator contributes to high resource utilization with 1271 LUTs and 3334 flip-flops.
This calculation of a correlation results in 8717 LUTs, 13270 flip-flops and 11 DSP slices. The utilization percentage of such a calculation is illustrated in Table IV. It is evident that the

Z-7020 already has a high level of utilization with even just one calculation. The benefit of parallelization, which an FPGA offers, cannot be taken advantage at this point. Since 50 % of the resources were already exhausted with three correlation calculations. The more modern platform of the Zynq UltraScale+™ offers a significant advantage. This requires for one correlation calculation only 4 % of the available LUTs and 3 % of the flip-flops.

TABLE IV. PERCENTAGE UTILIZATION OF THE RESSOURCES FROM THE XILINX Z-7020, ZU3EG AND ZU9EG

Platform		**Resources**		
		LUT	**Flip-flops**	**DSP Slices**
Total result for ONE correlation		8717	13270	11
Zynq-7000 SoC	Z-7020	17 %	13 %	5 %
Zynq UltraScale+™ MPSoC	ZU3EG	13 %	10 %	3 %
	ZU9EG	4 %	3 %	1 %

Thereby, many calculations lead to a high resource utilization. For minimizing this, some appropriate actions were pre-calculated. For example, a map with mean values and standard deviation can be stored in memory. Furthermore, this process is implemented in PL and processing system (PS). Afterward, the estimation and the storage of a disparity map can be performed. At last the 3D points out of the corresponding points can be computed (e.g. via triangulation) via GPU/CPU.

4. HARDWARE ARCHITECTURE

By using the Xilinx Zynq technology a hybrid architecture can be deployed. The Zynq platform combines FPGA-fabric with a dual-core ARM CPU on a single chip. In detail it contains FPGA based programmable logic (PL), an ARM-based processing system (PS) and a graphic processing unit (GPU). This combination enables extensive possibilities of image processing. For example, the preprocessing can be accelerated by the combination of PS and PL in the system.
As interfaces between PS and PL, there exist direct connections (EMIO) or direct interrupt connections in addition to shared DDR4 random access memory (RAM). For connecting the cameras with the embedded platform, the camera sensor is linked via Low Voltage Differential Signaling (LVDS) to the PL. After the data is processed the output signal is transmitted via Gigabit Ethernet Vision (GigE vision).
Vivado Design Suite is used as development environment for designing and programming the Xilinx FPGA devices. The version Vivado 2016.4 is used.
One important part in this research project is on preprocessing and computing corresponding points in the FPGA. Furthermore, storing the generated images in memory is an important issue. Due to the large amount of data, additional external memory is used. This is used in addition to the embedded memory (BRAM) for processing the data. The two units PS and GPU will be initially served as a reserve. [25, 26]

5. CONCLUSION AND FUTURE WORK

In this contribution, a theoretically approach for a stereo vision based system with an image processing unit existing of an FPGA is presented. Incoming image data from cameras arranged in a homogenous 3D sensor network are transmitted directly into a re-configurable hardware

platform. An offline calibration provides the necessary rectification maps which are required for further processing in the FPGA. Using this, the images can be rectified during the image acquisition. The subsequent stereo matching is performed by the calculation of the normalized cross correlation. This approach contains the problem of extremely high resource utilization due to the complex computing operations. The hardware resources depend on the chosen hardware platform. First attempts were made on a Xilinx Zynq 7020.
Future work involves the realization of a demonstrator including the acquisition of an object with a large volume in the range of several meters. Furthermore, a setup of a 3D sensor network calibration up to self-calibration has to be realized. The focus is on developing massive parallel algorithms for high-speed 3D-calculations with a combination of an FPGA and an GPU-based approach.

6. ACKNOWLEDGMENTS

This research project (2016 FGR 0044, project acronym *DIADEM*) is supported by the Free State of Thuringia, the European Social Fund (ESF) of the European Union and the Thüringer Aufbaubank (TAB).

REFERENCES

[1] T. Bell and S. Zhang, "Toward superfast three-dimensional optical metrology with digital micromirror device platforms," *Optical Engineering,* vol. 53, 2014.

[2] C. Ahlberg, F. Ekstrand, M. Ekstrom, G. Spampinato, and L. Asplund, "GIMME2 - an embedded system for stereo vision and processing of megapixel images with FPGA-acceleration," in *2015 International Conference on ReConFigurable Computing and FPGAs (ReConFig)*, 2015, pp. 1-8.

[3] G. Frankowski and R. Hainich, "DLP-based 3D metrology by structured light or projected fringe technology for life sciences and industrial metrology," 2009, pp. 72100C-72100C-12.

[4] N. Chang, T. M. Lin, T. H. Tsai, Y. C. Tseng, and T. S. Chang, "Real-Time DSP Implementation on Local Stereo Matching," in *2007 IEEE International Conference on Multimedia and Expo*, 2007, pp. 2090-2093.

[5] C. Murphy, D. Lindquist, A. M. Rynning, T. Cecil, S. Leavitt, and M. L. Chang, "Low-Cost Stereo Vision on an FPGA," in *15th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM 2007)*, 2007, pp. 333-334.

[6] R. Kalarot and J. Morris, "Comparison of FPGA and GPU implementations of real-time stereo vision," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, 2010, pp. 9-15.

[7] C. Banz, S. Hesselbarth, H. Flatt, H. Blume, and P. Pirsch, "Real-time stereo vision system using semi-global matching disparity estimation: Architecture and FPGA-implementation," in *2010 International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation*, 2010, pp. 93-101.

[8] P. Greisen, S. Heinzle, M. Gross, and A. P. Burg, "An FPGA-based processing pipeline for high-definition stereo video," *EURASIP Journal on Image and Video Processing,* vol. 2011, p. 18, 2011.

[9] S. Mattoccia and M. Poggi, "A passive RGBD sensor for accurate and real-time depth sensing self-contained into an FPGA," presented at the Proceedings of the 9th International Conference on Distributed Smart Cameras, Seville, Spain, 2015.
[10] B. Jähne, *Digitale Bildverarbeitung und Bildgewinnung*, 7., neu bearb. Aufl. ed. Berlin [u.a.]: Springer Vieweg, 2012.
[11] J. G. Fryer and D. C. Brown, "Lens distortion for close-range photogrammetry," *Photogrammetric engineering and remote sensing,* vol. 52, pp. 51-58, 1986.
[12] O. Schreer, *Stereoanalyse und Bildsynthese. Mit 6 Tabellen*. Berlin [u.a.]: Springer, 2005.
[13] S. Heist, P. Kühmstedt, A. Tünnermann, and G. Notni, "Theoretical considerations on aperiodic sinusoidal fringes in comparison to phase-shifted sinusoidal fringes for high-speed three-dimensional shape measurement," *Applied Optics,* vol. 54, pp. 10541-10551, 2015/12/10 2015.
[14] S. Heist, A. Mann, P. Kühmstedt, P. Schreiber, and G. Notni, "Array projection of aperiodic sinusoidal fringes for high-speed three-dimensional shape measurement," *Optical Engineering,* vol. 53, pp. 112208-112208, 2014.
[15] J. P. Lewis, "Fast Normalized Cross-Correlation," *Industrial Light & Magic,* 1995.
[16] S. Heist, P. Lutzke, I. Schmidt, P. Dietrich, P. Kühmstedt, A. Tünnermann*, et al.*, "High-speed three-dimensional shape measurement using GOBO projection," *Optics and Lasers in Engineering,* vol. 87, pp. 90-96, 12// 2016.
[17] G. Notni and P. Kühmstedt, "Hochgeschwindigkeits-3D-Messungen," *PHOTONIK,* 2016.
[18] P. Ott, "Imaging with tilted surfaces: an efficient matrix method for the generalized Scheimpflug condition and its application to rotationally symmetric triangulation," *Journal of the Optical Society of America A,* vol. 22, pp. 1077-1085, 2005/06/01 2005.
[19] S. Zwick, P. Kühmstedt, and G. Notni, "Phase-shifting fringe projection system using freeform optics," 2011, pp. 81690W-81690W-9.
[20] MATLAB, "Matlab Documentation," *https://de.mathworks.com/help/vision/index.html,* 20.06.2017.
[21] O. CV, "Open CV documentation " *http://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html,* 20.06.2017.
[22] Xilinx, "Performance and Resource Utilization for Adder/Subtracter v12.0," *https://www.xilinx.com/support/documentation/ip_documentation/ru/c-addsub.html,* 24.07.2017.
[23] Xilinx, "Performance and Resource Utilization for CORDIC v6.0," *https://www.xilinx.com/support/documentation/ip_documentation/ru/cordic.html,* 24.07.2017.
[24] Xilinx, "Performance and Resource Utilization for Divider Generator v5.1," *https://www.xilinx.com/support/documentation/ip_documentation/ru/div-gen.html,* 24.07.2017.
[25] Xilinx, *https://www.xilinx.com/,* 20.06.2017.
[26] Xilinx, "Vivado Design Suite User Guide," *https://www.xilinx.com/support/documentation/sw_manuals/xilinx2017_1/ug910-vivado-getting-started.pdf* 20.06.2017.

CONTACTS

M.Sc. Susann Winkler susann.winkler@tu-ilmenau.de
Dr. Maik Rosenberger maik.rosenberger@tu-ilmenau.de