

# Enhanced Omnidirectional Image Reconstruction Algorithm and its Real-Time Hardware

Abdulkadir Akin<sup>†</sup>, Elif Erdede<sup>\*</sup>, Hossein Afshari<sup>†</sup>, Alexandre Schmid<sup>†</sup>, Yusuf Leblebici<sup>†</sup>

<sup>†</sup>Microelectronic System Laboratory, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland

<sup>\*</sup>Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey

{abdulkadir.akin, hossein.afshari, alexandre.schmid, yusuf.leblebici}@epfl.ch, erdede@sabanciuniv.edu

**Abstract**— Omnidirectional stereoscopy and depth estimation are complex problems of image processing to which the Panoptic camera offers a novel solution. The Panoptic camera is a biologically-inspired vision sensor made of multiple cameras. It is a polydioptric system mimicking the eyes of flying insects where multiple imagers, each with a distinct focal point, are distributed over a hemisphere. Recently, the omnidirectional image reconstruction algorithm (OIR) and its real-time hardware implementation have been proposed for the Panoptic camera. This paper presents an enhanced omnidirectional image reconstruction algorithm (EOIR) and its real-time implementation. The proposed EOIR algorithm provides improved realistic omnidirectional images and residuals compared to OIR. As a processing core of EOIR, 57% of the available slice resources in a Virtex 5 FPGA are consumed. The proposed platform provides the high bandwidth required to simultaneously process data originating from 40 cameras, and reconstruct omnidirectional images of 256x1024 pixels at 25 fps. This proposed hardware and algorithmic enhancements enable advanced real-time applications including omnidirectional image reconstruction, 3D model construction and depth estimation.

**Keywords**- Angle of View; Hardware Implementation; Omnidirectional Image Reconstruction; Multiple Camera System

## I. INTRODUCTION

Three dimensional (3D) visualization of a full 360° scene forms the foundation for enabling the emergence of novel applications in security systems, automotive platforms and mobile robots, realistic computer games and 3D cinematography. The utilization of such systems with 360° 3D visual characteristics has recently significantly increased, with the commercialization of consumer electronics products.

Omnidirectional stereoscopy and depth estimation are the essential requirements of 360° 3D visual systems [1]. Omnidirectional stereoscopy entails capturing and subsequently displaying the image of a scene over a very large angle-of view (AOV) up to 360°. In order to provide a wide AOV, recent research has mainly focused on developing complex lenses, mirrors and multiple camera systems.

Fish-eye lenses are major lens types that provide 180° to 200° AOV for a single camera. However, distortion in the

edges of the images limits the use of single complex lens, for a single image sensor. In [2], a camera system with a convex mirror is developed in order to create 360° omnidirectional images. This method does not require intricate hardware or software for constructing omnidirectional images, and is thus widely cited in literature related to omnidirectional image reconstruction.

In [3], the real-time hardware implementation of a system consisting of a convex mirror placed over a single camera is introduced; that provides 360° panoramic images of 3200x768 resolution at 40fps. The presented hardware fits into a Spartan 3 FPGA, and utilizes only 2 SDRAMs. Nevertheless, the resolution of omnidirectional images is limited to the resolution of a single camera. Moreover, it is incapable of showing the overall top view since the AOV of the camera is restricted to the area located below the mirror. Also, the convex mirror provides distortion at the edges of the image. Furthermore, the platform cannot be used for depth estimation and 3D reconstruction due to its single observation point feature.

An alternate method enabling wide AOV image acquisition is based on a single-camera multiple-lens optical sensors system [4–6]. Devices taking advantage of this technique and presenting very large AOVs are mainly inspired by the insect faceted eyes. Faceted insect eyes consist of hundreds to thousands channels, each called ommatidia [7]. These channels capture light within a defined angle and transmit it to light sensitive receptor cells. In [4], artificial insect eyes, cluster eyes and artificial apposition compound eyes (APCO) are fabricated using wafer-scale techniques. In [5], the AOV of eye clusters and APCOs is improved to 124° by using two additional micro-lenses in each channel, thereby causing significant distortion. In [6], a system named Krill-eye is presented. The Krill-eye utilizes 180 GRIN lenses and hence, provides distortion-free 180° AOV. However, designing a Krill-eye composed of 180 lenses is not trivial, while the AOV remains significantly limited under 360°.

Multiple-camera systems are proposed as an alternative method, which enables obtaining wide AOVs. In addition to the synchronization of cameras, the very large involved data rates present new challenges for the implementation of these systems. Primary camera array systems were developed for

recording; the records were processed later offline on PCs [8]. Many multiple camera systems are used for only object tracking and detection, where the goal does not involve the creation of single omnidirectional images [9, 10]. Each image obtained from multiple cameras should be combined appropriately in order to prevent unrealistic discontinuity. As a result, insect-eye inspired multiple camera systems have been proposed. An image mosaic algorithm and its real-time hardware implementation are presented in [11]. This multi-camera system supports three cameras, and is capable of reaching an AOV smaller than  $180^\circ$ . The system operates in real-time benefiting from the utilization of digital signal processors (DSPs). Furthermore, the resulting omnidirectional image does not exhibit significant distortion or discontinuity.

In [12], a single-lens and camera based omnidirectional reconstruction is used for object detection. Single-lens-based omnidirectional cameras are reported to present non-uniform resolution in [12]. Therefore, a multiple-camera system which uses an additional PTV camera in order to compensate the deficiencies of the single lens based omnidirectional camera is proposed. In [13], a planar multiple-camera system composed of 100 cameras is presented as a solution to reach high resolution, high frame rates and high AOV. The presented platform consisting of 100 FPGAs is aimed to record large amount of data to process offline, and thus provides limited local processing at the camera level. However, it still reaches less than  $180^\circ$ AOV due to its planar structure.

Recently, a spherical multi-camera vision sensor called the Panoptic camera and a specific omnidirectional image reconstruction algorithm (OIR) for the Panoptic camera have been proposed to enable  $360^\circ$ AOV with a high resolution [14]. The Panoptic camera is a biologically-inspired multi-camera vision sensor mimicking the eyes of flying insects where multiple imagers, each with a distinct focal point, are distributed over a hemisphere. The first real-time hardware implementation of OIR is presented in [15].

In this paper, an enhanced version of the omnidirectional image reconstruction algorithm (EOIR) and its first real-time hardware implementation are presented. The proposed EOIR algorithm increases the realistic aspect of omnidirectional images and improves residuals. The entire system provides the high bandwidth required to simultaneously process data originating from 40 cameras, and reconstruct omnidirectional images of  $256 \times 1024$  pixels at 25 frames per second. Moreover, the hardware architecture is designed to provide flexibility in the selection of image resolution, AOV, contributing camera as well as algorithm choice. The 3D reconstruction can be viewed real-time on PC, after omnidirectional images are transmitted to the PC through a USB channel. The proposed hardware and algorithmic developments enable advanced real-time applications including omnidirectional image reconstruction, 3D model construction (3D display) and depth estimation.

This paper is structured as follows. The geometrical arrangement and the physical realization of the Panoptic camera are briefly introduced in Section 2. Section 3 reviews the OIR algorithm, and Section 4 details the enhanced omnidirectional image reconstruction algorithm (EOIR). The flexible architecture of the hardware platform of both OIR and EOIR is presented in Section 5, and results of the implementation of the OIR and EOIR are discussed in Section 6.

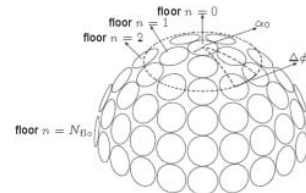


Figure 1. Hemispherical structure with five floors

## II. PANOPTIC CAMERA

The construction of the Panoptic camera follows the method earlier prescribed by the authors in [14]. A hemisphere is fully covered with cameras following a systematic arrangement depicted in Figure 1. The Panoptic camera has 7 floors and 40 cameras. The angle-of-views (AOV) of each camera are  $53^\circ$  and  $43^\circ$  for the horizontal and vertical axis, and the resolution of each camera is  $352 \times 288$  pixels in horizontal and vertical directions, respectively. The details of the intrinsic and extrinsic calibrations, multiple camera synchronization procedures, the efficient memory organization scheme and the camera interface modules are described in [15].

## III. OMNIDIRECTIONAL IMAGE RECONSTRUCTION ALGORITHM

The omnidirectional image reconstruction algorithm assumes that the space located around the Panoptic camera is hemispherical, where the surface of the structure is divided into an equiangular grid with  $N_\theta$  latitude and  $N_\phi$  longitude pixels. Figure 2(a) shows a hemispherical surface over which a linear pixelization scheme is applied, and Figure 2(c) depicts its 2D reconstruction. The latter is divided into 256 and 1024 pixels for  $N_\theta$  and  $N_\phi$  respectively, which are linearly distributed. The direction of each pixel is described by a unit vector,  $\omega$  with spherical coordinates  $(\theta_\omega, \phi_\omega)$ , which is visually presented in Figure 2(b).

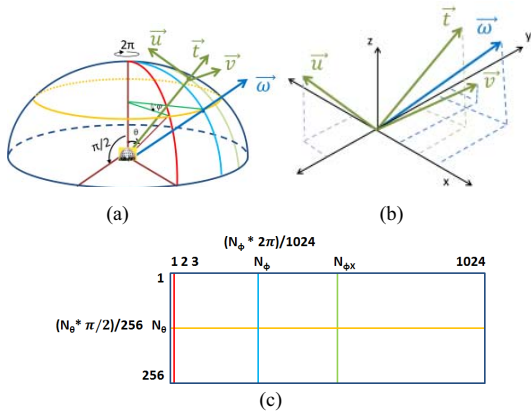
The vector  $t$ , which is shown in Figure 2(a), represents the focus direction, whereas  $u$  and  $v$  vectors stand for the vertical and horizontal directions of the camera in pixel representation. After the computation of  $\omega$ , all cameras having  $\omega$  in their angle of view are determined by processing the dot product of  $\omega$  with  $t$ . Then, pixel grid locations onto which each  $\omega$  projects are found. However, due to the fact that  $\omega$  may not coincide with the exact pixels on the camera images, each projected pixel is interpolated with its adjacent pixels in order to extract the light intensity in that particular  $\omega$  direction.



(a)



(b)

Figure 3: Omnidirectional 2D reconstructions using a) *Equal area distribution* b) *Linear pixelization*Figure 2: (a) hemispherical surface showing  $\theta$  and  $\phi$  angles and unit vectors used in calculations (b)  $\omega$  vector and corresponding unit vectors;  $t$  (focus direction of the camera),  $u$  (vertical direction in the pixel representation of the camera),  $v$  (horizontal direction in the pixel representation of the camera) (c) 2D reconstruction using *linear pixelization*

(a)



(b)

Figure 4: Captured images from two different cameras a) top camera b) side camera

Since the AOVs of the cameras may overlap, the final light intensity of each pixel is determined by assigning a *weight* to each contributing camera. The *weight* of a camera is calculated by computing its 2D distance to the pixel that is being considered, and taking its reciprocal.

Two optional schemes are proposed for the final interpolation process, namely i) nearest neighbor interpolation and ii) linear interpolation. The nearest

neighbor interpolation method extracts pixels from the cameras that provide maximum weights, whereas the linear interpolation scheme linearly interpolates the light intensity extracted from each contributing camera.

The constructed omnidirectional image using the OIR algorithm is shown in Figure 3(b), and images captured by two cameras are shown in Figure 4.

#### IV. ENHANCED OMNIDIRECTIONAL IMAGE RECONSTRUCTION ALGORITHM

Two alternate algorithms are proposed for enhanced reconstruction. The first one is named equal area distribution, which provides homogeneous resolution over the entire reconstruction area, and the second is the farthest neighbor interpolation, which is introduced for better residual that is useful for depth analysis.

##### A. Equal Area Distribution on Omnidirectional Image

As observed in Figure 3(b) where linear pixelization is applied as the pixelization method, the details on the sides of the image, exemplified by desks located on the right side, occupy less area compared to the contribution originating from the top camera, considering ceiling and lamps in this case. Conversely, since the human eye naturally observes along the horizontal axis naturally followed by the majority of conventional imaging applications, data located in the horizontal plane is more important than data issued from the top plane.

The ‘On hemisphere’ line in Table 1 indicates the ratio of the covered surface area over total hemispherical area for various  $\theta$  angles, assuming that the surrounding space is observed from the center of the hemisphere in Figure 2(a). The other entries in the table present an elliptical distribution (an alternate way of pixelization method based on elliptical surface area distribution), and the existing linear distribution method of OIR. As expected due to the basic 3D geometry, the Equal Area Distribution of EOIR results completely match with the On Hemisphere calculations, resulting in equal resolution distribution on the whole hemispherical

surface. In contrast, when the linear pixelization proposed in [14] (Linear Distribution in Table 1) is applied, the resolution of the reconstructed image decreases from the top to bottom, and the contribution from top cameras is overestimated, while data from side cameras is undervalued. The picture taken from top cameras utilize more pixels on the omni-directional image than the side view cameras as observed in Figure 3(b) and Figure 4 which results in providing less resolution for side cameras than the top cameras.

Table 1. Surface Area Coverage (%) with respect to  $\theta$  angles (radians) for Possible Pixelization Methods.

	$\pi/2$	$\pi/3$	$\pi/4$	$\pi/6$	$\pi/12$
<b>On Hemisphere (reference)</b>	100	50	29	13	3.4
<b>Equal Area Distribution</b>	100	50	29	13	3.4
<b>Elliptical Distribution</b>	100	25	13	6	2
<b>Linear Distribution</b>	100	66	50	33	17

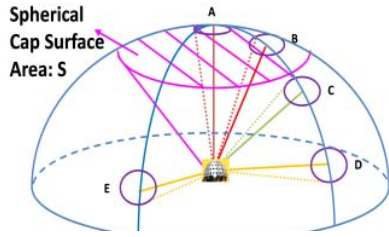


Figure 5. Hemispheric structure for resolution calculation in terms of ppaa

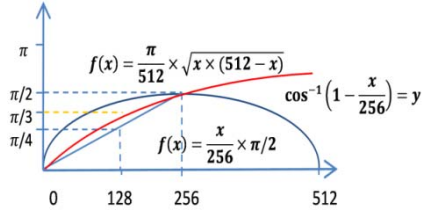


Figure 6: Relationship between  $N$  and  $\theta$  angle for different methods of pixelization

Table 2. Resolution Values [ppaa] for Different  $\theta$  Angles

$\theta$ Angle (degrees)	Corresponding Unit Angular Area	Linear	Equal Area Distribution
0	A	2913	39.92
10	B	146.3	39.92
39.6390	C	39.92	39.92
80	D	26.81	39.92
80	E	26.81	39.92

Therefore, a novel pixelization method is proposed based on equal area distribution. The anticipated method allocates  $N_\theta$  latitude pixels with respect to the surface area of hemispheres covered at particular  $\theta$  angles. An example is presented in Figure 5, where the total surface area considered

at  $\theta = \pi/6$  is shown by spherical cap S. The contribution at each  $\theta$  angle is calculated using expressions (1) and (2), derived using 3D geometry.

$$\frac{\int_0^{2\pi} \int_0^\theta r^2 \sin \theta d\theta d\varphi}{2\pi r^2} = \frac{N_\theta}{256} \quad (1)$$

$$\cos^{-1}\left(1 - \frac{N_\theta}{256}\right) = \theta \quad (2)$$

In contrast to linear pixelization, this method results in varying latitude pixel steps, which produces a non-linear relation between the  $N_\theta$  and  $\theta$  angles, which is presented in Figure 6.

The universally accepted unit of measurement of 2D image resolution is line pairs per millimeter ( $lp$ ). Since all the longitudes merge on the pole of hemisphere, this unit is not an appropriate measurement for defining hemispherical image resolution. Therefore, we provide a new suitable unit of measurement which is the Pixels Per unit Angular Area ( $ppaa$ ) metric. The unit angular area is defined in this context as the surface area of a spherical cap with exactly  $1^\circ$   $\theta$  angle which provides constant surface area on hemisphere due to the fixed radius for all possible  $\theta$  and  $\varphi$ . Resolutions at different  $\theta$  angles which are presented in Figure 5 are compared in Table 2 in terms of their  $ppaa$ . When the linear pixelization scheme is adopted, small  $\theta$  angles yield a very high resolution such as 2913  $ppaa$  for  $\theta = 0^\circ$ , gradually decreasing for higher  $\theta$  angles. For instance,  $\theta = 80^\circ$  produces a  $ppaa$  value 26.81, which is approximately 100 times smaller than  $ppaa$  of  $\theta = 0^\circ$ . Since the human eye captures constant resolution while looking at different angles, linear pixelization does not provide realistic enough hemispherical images. On the other hand, the equal area pixelization scheme provides a homogeneous resolution distribution over the entire omnidirectional image by distributing 39.92  $ppaa$ . Hence, as shown in Figure 3(a), the equal area distribution scheme realizes realistic pictures by homogeneously distributing equal number of pixels to the top view and side view cameras.

### B. Farthest Neighbor Interpolation

Depth estimation is one of the fundamental prerequisite of 3D visualization, and a non-trivial problem in signal processing. Literature suggests that distances to objects can be processed from an image of the scene using distinct reference centers to capture several images of the same view, and subsequently processing the residual of these images [16]. In [14], it is proposed that extra omnidirectional images can be obtained by changing the reference center using the Panoptic camera. Figure 7 shows a hemispherical structure where two centers are deployed in order to process the residual. Providing more observable residual indicates the success of providing two different views. This method yields a practical residual at the top parts and the sides, where the directions of the projections do not coincide on the hemispherical space. Figure 8(b) shows the residual that is obtained by using the described method of OIR. However, as shown in Figure 7, projections from reference centers

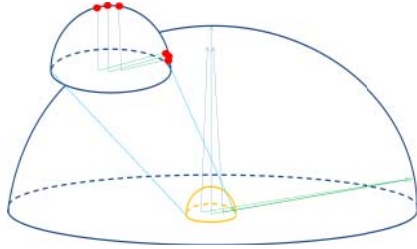


Figure 7. Residual using two centers (Big hemisphere depicts the surrounding space, and the hemisphere at its center depicts panoptic camera)



(a)



(b)

Figure 8. Residual using (a) farthest neighbor interpolation, (b) two distinct reference centers

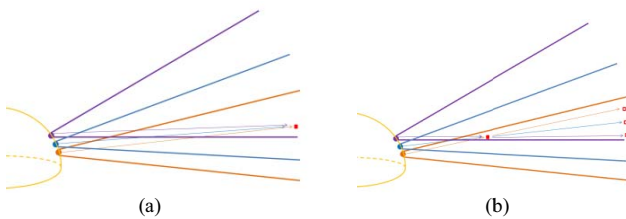


(a)



(b)

Figure 9: Residual (zoomed) using (a) farthest neighbor interpolation (b) two distinct reference centers



(a)

(b)

Figure 10. Residual using 2 centers (three dots on the hemisphere represent three cameras, and the bold lines represent their AOVs) (a) projecting the object at the infinity (b) projecting the closer object

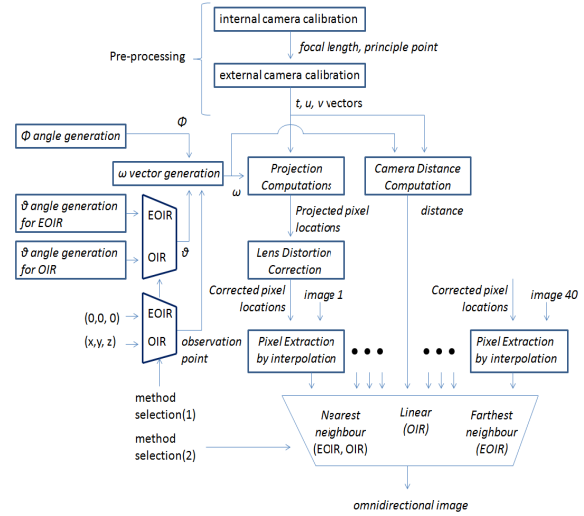


Figure 11: Block diagram of omnidirectional image reconstruction algorithms

almost overlap when the reference centers and the projection vector are on the same direction. Hence, OIR almost interpolates the same pixel value while creating different omnidirectional views. As shown in Figure 9(b), it does not provide practical residual for the overlapping regions chairs and windows that are shown in Figure 4(b). The straightforward algorithmic solution could be using more than two reference centers that are not on the same line. However, this solution is not hardware oriented since it requires additional memory bandwidth to support additional reconstruction.

As presented in Figure 10(a), when the projected object is at infinity, the projected location from different cameras provides the same pixel during the omnidirectional reconstruction. However, when the object moves closer to the panoptic camera, as shown in Figure 10(b), the projected object will be at different coordinates in the images of observing cameras. As a result, the quality of the residual can be improved by using an alternate algorithm particularly suitable for handling the problematic sides. The *farthest neighbor interpolation* with the use of *equal area distribution* is derived from this observation, and is based on the fact that the second omnidirectional projection can be obtained by making use of the *weights* of the cameras. The light intensity data is retrieved from the cameras that have minimum *weights* for each pixel. This scheme creates an image by applying the exact reverse operation with respect to the nearest-neighbor method. In addition, *equal area distribution* provides higher resolution for the sides, as shown in Table II. Thus, a very effective residual (Figure 9(a)) is extracted with the use of two distinct omnidirectional images that are obtained from the nearest and the farthest cameras, for each pixel. This technique provides a much elaborated sketch especially for the sides that are problematic, as depicted in Figure 9(a), and thus better

detectability of the depth estimation of the chairs shown in Figure 4(b).

The block diagrams of the OIR and EOIR algorithms are shown in Figure 11. Since the main computationally intensive part of the OIR and EOIR algorithms is the projection computations, i.e. pixel position generation, and it is the same module for both of them, algorithmic enhancements enable to realize a hardware which can switch between OIR and EOIR.

### V. HARDWARE IMPLEMENTATION

The hardware platform which has been presented in [14] and [15] is used for implementing the EOIR algorithm in real-time. Since the proposed EOIR algorithm is a hardware oriented algorithm, its adaptation into the system presented in [15] is straightforward. As shown in Figure 12(a), the platform consists of two concentrator FPGAs handling local image capturing with 20 imagers in parallel, a central FPGA to control concentrator FPGAs and to transmit final omnidirectional images to the PC, and four 40MB SRAM memories buffering captured images. As shown in Figure 13(b), the main processing of the algorithm is handled in the Concentrator FPGA and it involves six sub-blocks. The details about the Cameras Control, Camera Input Channels, Data Transmit Multiplexer, Data and Control Unit, Memory Controller modules and efficient memory organization are presented in [15]. In this paper, the hardware implementation of the EOIR algorithm is presented for the Image Processing and Application Unit of the concentrator FPGA.

The proposed hardware implementation of the omnidirectional image reconstruction is flexible to realize the OIR and EOIR algorithms with configurable resolutions, AOV and camera selection options. Options are selectable by the embedded soft controller (microblaze processor) and the GUI.

The system-level architecture of the EOIR algorithm consists of five sub-blocks depicted in Figure 13. The angle generation module shown in Figure 14 creates the spherical coordinates of pixel direction  $\omega=(\theta_\omega, \varphi_\omega)$  using two accumulators. To generate all possible combinations of  $\theta_\omega$  and  $\varphi_\omega$ , one accumulator is assigned to increment, while the other completes its full-range cycle. Here,  $\theta$  and  $\varphi$  have separate linear incrementing indexes  $K_\theta$  and  $K_\varphi$  as well as minimum and maximum values defined for both angles, which are all parameters supplied to the reconstruction system. Defining initial and final value parameters of  $\theta$  and  $\varphi$  provides flexibility to visualize either the whole reconstruction of  $360^\circ \times 90^\circ$  image or some portion of the vision of Panoptic camera. The parameterized incrementing indexes  $K_\theta$  and  $K_\varphi$  define the step amount, thereby the size of omnidirectional image.

A 13-bit look-up-table in Figure 14 is used for implementing equation (2) in order to add the new pixelization scheme based on the equal resolution

distribution. Arccos and divisions for  $\theta$  processing are pre-computed for all different values of  $N_\theta$  and placed in a lookup table (LUT) to save the hardware resources.

In order to supply the user with the ability to switch between the OIR and EOIR algorithms in real-time processing, the output of the OIR algorithm is delayed by two clock cycles using two pipelined registers as shown in Figure 14. This additional pipelining has no adverse effect on the throughput of the OIR algorithm.

Following the computation of  $\theta_\omega$  and  $\varphi_\omega$ , the  $\omega$  vector is generated in the next sub-block named  $\omega$  vector generation in Figure 13. This sub-block is used as it is presented in [15]. The camera select and distance generation module in Figure

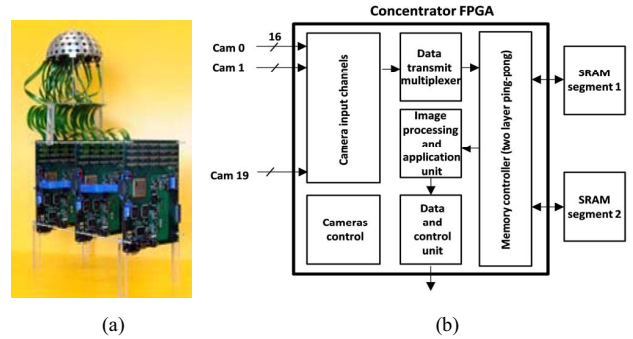


Figure 12. (a) FPGA platform for Panoptic camera (b) Block diagram of the Concentrator FPGA

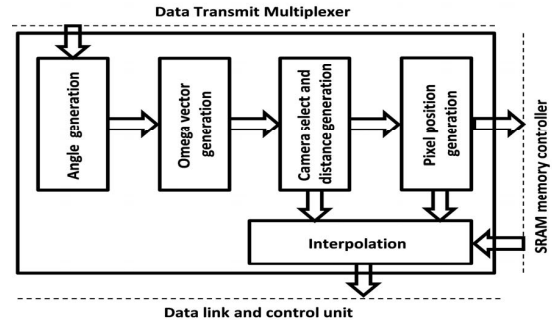


Figure 13. Block diagram of the omnidirectional vision reconstruction unit inside the image processing and application block.

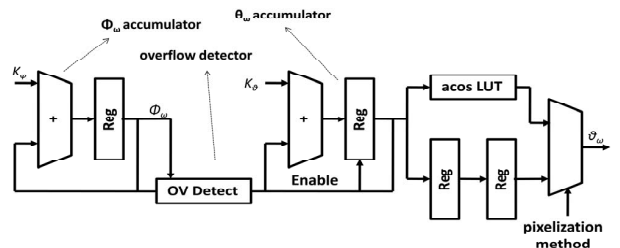


Figure 14.  $\theta$  and  $\varphi$  angle generation module architecture for the OIR and EOIR implementations

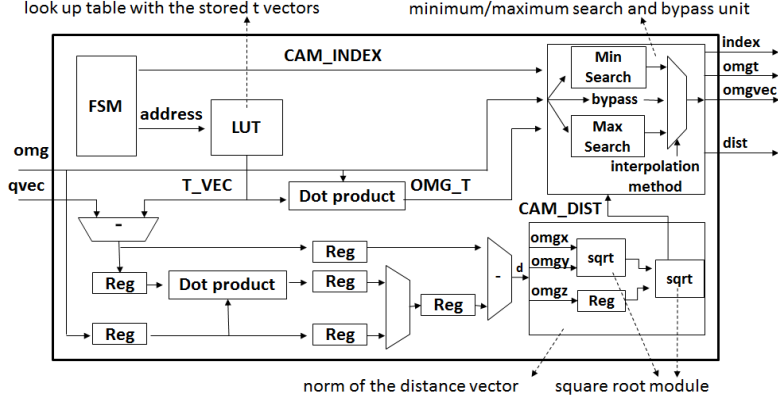


Figure 15. Architecture of the camera select and distance generation sub-block for EOIR algorithm



Figure 16. Reconstructed 360° views

13 identifies cameras that contribute into pixel direction  $\omega$ . This block is also in charge of calculating weights for interpolation. If the linear interpolation method is chosen, the weights of contributing cameras and their corresponding indexes are sent to the interpolation sub-block. On the other hand, if the nearest neighbor scheme is chosen, a maximum search of the weights of the contributing cameras is carried out, and the index of the camera with the maximum weight is passed to the pixel position generation sub-block, as depicted in Figure 15.

A maximum-distance search algorithm is supplied to the omnidirectional image reconstruction hardware for the implementation of the farthest neighbor interpolation method, in order to retrieve the light intensity value from the farthest camera. This operation is handled as the exact inverse operation of the nearest neighbor method that is presented above. The block diagram of minimum-maximum search and bypass unit is also shown in Figure 15.

After determining which cameras correspond to the projection of each  $\omega$  vector, the pixel position module retrieves the contributing pixel value on the image frame of the contributing cameras from the SRAM. Subsequently, this retrieved value is sent to the interpolation sub-block in order to implement linear interpolation using the pixels received from the contributing cameras. Finally, this sub-block supplies the calculated intensity value to the data link and control unit.

## VI. IMPLEMENTATION RESULTS

The proposed hardware architectures are implemented using VHDL. The VHDL RTL codes are synthesized and

mapped to Xilinx XC5VLX50-1FF1153 FPGAs using Xilinx ISE 11.5. The hardware implementations are verified with post place and route simulations using Mentor Graphics Modelsim 6.2b and the full verification is carried out on the Panoptic camera platform.

The resource allocation of omnidirectional image reconstruction hardware on the selected platform is presented in Table 3, in which EOIR indicates the configurable hardware that is able to implement both OIR and EOIR algorithms, providing a selection for the user to choose either of them. The results given in Table 3 indicate the required hardware resources for a single FPGA that performs operations for 40 cameras. The hardware resource consumption is divided by two using two concentrator FPGAs each performs operations for 20 cameras. Hardware resource consumption details about the central FPGA is provided in [15].

The total pipeline latency of the OIR implementation is equal to 116 clock cycles. This latency increases in the EOIR implementation to 118 clock cycles, without causing an effect on the throughput of the pipelined system. Arithmetic operations are implemented in 16 bit fixed-point precision format, which provides good tradeoff between image quality, hardware area and clock frequency.

The post place and route maximum operating frequency is measured by the XILINX ISE timing analyzer tool at 212 MHz, which is identical for both OIR and EOIR implementations. As observed in Table 3, the hardware implementation of the EOIR algorithm does not cause a significant increase of resources over the OIR's implementation. The area and the memory usages approximately match, while the resolution of the 3D reconstructed image and the quality of the residual are visibly improved.

Finally, after the omnidirectional image is reconstructed at 25 fps, the omnidirectional images are transmitted to the PC through a USB channel. Subsequently, 2D omnidirectional images are converted to 3D using OpenGL C++ and the Qt library, as a real-time video display process. Figure 16 shows

two different views of a real-time 360°×90° image produced after the construction of the real-time omnidirectional image on the computer. This system may artificially make the viewer feel himself being at the place of the Panoptic camera using prospective future 360° hemispheric displays, and the improvements presented in this paper may participate in the effort of providing virtual reality.

Table 3. Implementation Results

Resources&Parameters	Unit	OIR [15]	EOIR
Occupied Slices	Occupied Utilization	4127 57	4105 57
Slice LUTs	Occupied Utilization	9343 32	9351 32
Slice Registers	Occupied Utilization	10538 36	10621 36
BRAM / FIFO	Occupied Utilization	22 45	24 50
Max. Clock Frequency	MHz	212	212
Latency	cycles	116	118

### CONCLUSION

This paper presents the enhanced omnidirectional image reconstruction algorithm, its real-time hardware architecture, realization and implementation results. The algorithmic details of the enhanced omnidirectional vision reconstruction of the Panoptic camera that handles 360° AOV for a 256×1024 image are presented. The proposed bio-inspired hardware is capable of providing realistic omnidirectional images at 25 fps with high and constant resolution over the entire constructed image, and promising residuals for depth estimation. Future work related to Panoptic camera focuses on improving the resolution of omnidirectional images using higher resolution cameras, 3D model constructions, 3D displays and depth estimation.

### ACKNOWLEDGEMENT

This research has been partly conducted with the support of the Swiss NSF under grant number 200021-125651. The authors gratefully acknowledge the support of Xilinx, Inc., through the Xilinx University Program.

### REFERENCES

[1] Y. Liu, X. Cao, Q. Dai, and W. Xu, "Continuous depth estimation for multi-view stereo," *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'09*, Florida, USA, June, 2009.

[2] J. Gluckman, S. K. Nayar, and K. J. Thoresz, "Real-time omnidirectional and panoramic stereo," *In Proceedings of the 1998 DARPA Image Understanding Workshop*, vol. 1, 1998, pp. 299–303.

[3] L. Chen, M. Zhang, B. Wang, Z. Xiong, and G. Cheng, "Real-Time FPGA-Based Panoramic Unrolling of High-Resolution Catadioptric Omnidirectional Images," *International Conference on Measuring Technology and Mechatronics Automation*, Washington, USA, 2009.

[4] J. Duparre, P. Schreiber, P. Dannberg, T. Scharf, P. Pelli, R. Volkel, H. Herzig, and A. Brauer, "Artificial compound

eyes - different concepts and their application to ultra flat image acquisition sensors," *Moems and Miniaturized Systems IV*, vol. 5346, 2004, pp. 89–100.

[5] Els Moens, Youri Meuret, Heidi Ottevaere, Mukul Sarkar, David San Segundo Bello, Patrick Merken and Hugo Thienpont, "An insect eye-based image sensor with very large field of view", *Proc. SPIE 7716*, 77162D, 2010.

[6] S. Hiura, A. Mohan, and R. Raskar, "Krill-eye: Superposition compound eye for wide-angle imaging via grin lenses," *IPSN Transactions on Computer Vision and Applications*, vol. 2, pp. 186–199, 2010.

[7] K. Hamanaka and H. Koshi, "An artificial compound eye using a microlens array and its application to scale-invariant processing," *Optical Review*, vol. 3, pp. 264–268, 1996, 10.1007/s10043-996-0264-6.

[8] P. Rander, P. J. Narayanan, and T. Kanade, "Virtualized reality: constructing time-varying virtual worlds from real world events," *IEEE Visualization'97*, pages 277-284, 1997.

[9] J.-E. Ha and I.-S. Choi, "Simple method for calibrating omnidirectional stereo with multiple cameras," *SPIE Optical Engineering*, vol. 50, no. 4, p. 43608, April, 2011.

[10] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, pp. 267–282, February 2008.

[11] G. Feng, W. Tian, C. Huang, T. Liu, and S. Zhang, "Wide field of view ccd camera based on multi-sensors image mosaics," *Proceedings of the 2008 Congress on Image and Signal Processing, Vol. 2 - Volume 02*. IEEE Computer Society, 2008, pp. 432–435.

[12] C. hao Chen, Y. Yao, B. Abidi, S. Member, A. Koschan, and M. Abidi, "Heterogeneous fusion of omnidirectional and ptz cameras for multiple object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 8, pp. 1052–1063, Jan 2008.

[13] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," *ACM Trans. Graph.*, vol. 24, pp. 765–776, July 2005.

[14] H. Afshari, L. Jacques, L. Bagnato, A. Schmid and P. Vanderghenst et al. Hardware Implementation of an Omnidirectional Camera with Real-time 3D Imaging Capability. 3DTV 2011, Antalya, Turkey, 2011.

[15] H. Afshari, L. Jacques, L. Bagnato, A. Schmid, P. Vanderghenst, and Y. Leblebici, "The PANOPTIC Camera: A Plenoptic Sensor with Real-Time Omnidirectional Capability," *Journal of Signal Processing Systems – Signal Image and Video Technology*, 2012.

[16] L. Bagnato, P. Vanderghenst and P. Frossard. A Variational Framework for Structure from Motion in Omnidirectional Image Sequences, in *Journal of Mathematical Imaging and Vision*, vol. 41, num. 3, p. 182-193, 2011.