J Sign Process Syst (2013) 70:305–328
DOI 10.1007/s11265-012-0668-4

The PANOPTIC Camera: A Plenoptic Sensor with Real-Time Omnidirectional Capability

Hossein Afshari · Laurent Jacques · Luigi Bagnato · Alexandre Schmid · Pierre Vandergheynst · Yusuf Leblebici

Received: 15 February 2011 / Revised: 11 November 2011 / Accepted: 8 February 2012 / Published online: 14 March 2012

Abstract A new biologically-inspired vision sensor made of one hundred "eyes" is presented, which is suitable for real-time acquisition and processing of 3-D image sequences. This device, named the Panoptic camera, consists of a layered arrangement of approximately 100 classical CMOS imagers, distributed over a hemisphere of 13 cm in diameter. The Panoptic camera is a polydioptric system where all imagers have their own vision of the world, each with a distinct focal point, which is a specific feature of the Panoptic system. This enables 3-D information recording such as omnidirectional stereoscopy or depth estimation, applying specific signal processing. The algorithms dictating the image reconstruction of an omnidirectional observer located at any point inside the hemisphere are presented. A hardware architecture which has the capability of handling these algorithms, and the flexibility to support additional image processing in real time, has been developed as a two-layer system based on FPGAs. The detail of the hardware architecture, its internal blocks, the mapping of the algorithms onto the latter elements, and the device calibration procedure are presented, along with imaging results.

Keywords Omnidirectional · Polydioptric · Plenoptic · Image rendering · Computer vision · Image processing hardware · 3-D vision system · FPGA

1 Introduction

Uncovering the nature of the information content carried by light rays that allow humans to "see" is a fundamental question that has been tackled by Leonardo da Vinci himself, who provided the following clue: "*Every body in the light and the shade fills the surrounding air with infinite images of itself; and these, by infinite pyramids diffused in the air, represent this body throughout space and on every side.*" [1]. With this description of the *radiant pyramid* [2] Leonardo strikingly determined that light rays originating from objects surfaces are

This research has been partly conducted with the support of the Swiss NSF under grant number 200021-125651.

H. Afshari, A. Schmid and Y. Leblebici are with the Microelectronic Systems Laboratory, L. Bagnato and P. Vandergheynst are with the Signal Processing Laboratory of the Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. L. Jacques is with the Institute of Information and Communication Technologies, Electronics and Applied Mathematics (ICTEAM), Electrical Engineering Department (ELEN), University of Louvain (UCL), Louvain-la-Neuve, Belgium. He is funded by the Belgian Science Policy (Return Grant, BELSPO) joined to the Belgian Interuniversity Attraction Pole IAP-VI BCRYPT.

H. Afshari (✉) · L. Jacques · L. Bagnato · A. Schmid · P. Vandergheynst · Y. Leblebici
EPFL-STI-IEL-LSM, Bldg ELD, Station 11,
1015 Lausanne, Switzerland
e-mail: hossein.afshari@epfl.ch

L. Jacques
e-mail: laurent.jacques@uclouvain.be

L. Bagnato
e-mail: luigi.bagnato@epfl.ch

A. Schmid
e-mail: alexandre.schmid@epfl.ch

P. Vandergheynst
e-mail: pierre.vandergheynst@epfl.ch

Y. Leblebici
e-mail: yusuf.leblebici@epfl.ch

filling the 3-D space without interference and outside the definition of an observer, i.e. light information exists prior to its measurement. More recently, Bergen and Adelson mathematically studied this principle in their original paper [2]. They define the concept of *plenoptic function* $\mathcal{L}(\mathbf{q}, \boldsymbol{\omega}, t, \lambda)$, that is, the function that represents the intensity (or radiance) that a perfect observer inside a given 3-D scene would record at any position $\mathbf{q}$, direction $\boldsymbol{\omega}$, time t and wavelength λ.

In this paper, one (grayscale image) or three (color image) wavelength channels are considered, and time t is omitted assuming static scene observation. With this very general definition, recovering an arbitrary view of the world from a set of positions and orientations can be understood as *sampling* the plenoptic function $\mathcal{L}$. Similarly, each sample of $\mathcal{L}$ corresponds to parameterizing the intensity of light rays flowing to a point in space and from all directions. The full plenoptic function contains all the information about the observed scene. For instance, the depth $\mathcal{Z}(\mathbf{q}, \boldsymbol{\omega})$ of the scene from the point $\mathbf{q}$ in the direction $\boldsymbol{\omega}$ can be computed from $\mathcal{L}$ since, by triangulation [3]:

$$\mathcal{L}(\mathbf{q} + \mathcal{Z}(\mathbf{q}, \boldsymbol{\omega})\, d\,\boldsymbol{\omega}, \boldsymbol{\omega}) = \mathcal{L}(\mathbf{q}, \boldsymbol{\omega} + d\,\boldsymbol{\omega}). \tag{1}$$

A relationship between the derivatives of the plenoptic function is obtained by linearizing the expression above, assuming the depth sufficiently large:

$$\mathcal{Z}(\mathbf{q}, \boldsymbol{\omega})\, \nabla_{\mathbf{q}} \mathcal{L}(\mathbf{q}, \boldsymbol{\omega}) \;=\; \nabla_{\omega} \mathcal{L}(\mathbf{q}, \boldsymbol{\omega}), \tag{2}$$

where $\nabla_{\mathbf{u}}$ stands for the gradient in the corresponding coordinates $\mathbf{u}$. The system presented in this paper must be thought of as a plenoptic sampler, which is illustrated by showing that the system parameterizes and interpolates light rays to reconstruct $\mathcal{L}(\mathbf{q}, \boldsymbol{\omega})$ for all positions $\mathbf{q}$ inside a bounded volume of space, i.e., from many different viewpoints.

Also notice that the authors in [3] classify different kinds of visual sensors according to their ability to deduce the egomotion of an observer with the recorded vision of the scene. They conclude that the perfect visual sensor should be both polydioptric and omnidirectional, which are the defining properties of the camera presented in this paper.

The very efficient visual system of flying insects has provided early inspiration for the presented hardware vision system. The common fly has two faceted eyes that provide it with omnidirectional vision, egomotion and depth estimations. Each of the fly faceted eye is an omnidirectional vision system composed of several thousands of rudimentary image sensors called ommatidias [4].

Mimicking the faceted eye's concept, an omnidirectional camera realized by layering CMOS image sensors over the surface of a spherical structure is proposed. This intelligent image acquisition system is named Panoptic, and has two distinguishable features. First it is an omnidirectional camera, which is able to record light information from any direction around its center. Second it is a polydioptric system where each CMOS facet consists of a CMOS camera with a distinct focal plane; hence the whole system is a multiple aperture camera.

Early attempts in fabricating omnidirectional vision (with a single focal point) are based on regular sensors capturing the image formed on a parabolic mirror [5]. Conversely, non-omnidirectional cameras still recording plenoptic (multi focal) information have been developed for almost 10 years using a lenticular array placed above a sensor plane [6]. An alternate solution has been proposed in [7], where a number of commercial cameras are placed in orbital plane, enabling the post processing reconstruction of a panoramic image used as an omnidirectional vision turret for robotic applications. Along the same lines, a large size stereoscopic omnidirectional camera prototype using commercial camera components is presented in [8]. Recently, two attempts in miniaturizing the omnidirectional vision system have been made, specially using microfabrication technologies into mimicking the insect compound eye [9, 10].

Solutions which have been proposed so far to realize omnidirectional vision suffer from various flaws which harm their practicality and effectiveness. Most of the proposed systems involve bulky or heterogeneous hardware, in the form of computers and/or laser-based distance measurement systems prohibiting actual portability, three-dimensional mirrors which are very delicate to manipulate, may cause local image distortion due to a complex and difficult to guarantee fabrication process as well as misalignment with the imager. Alternatively, the attempts to realize micrometer size omnidirectional vision mainly focus on the vision system, where the data communication and image processing is not considered. In this work, a solution solving these issues is proposed by applying a system-level consideration of omnidirectional vision acquisition and subsequent image processing.

The major design effort that is applied in the development of the Panoptic camera relates to its inspiration from biology, and is derived into the following four working hypotheses, namely, (i) integration of the vision acquisition and processing; the unique data acquisition system consists of identical image sensors, all of which are integrated into a compact system whose major mechanical limits is dictated by the size of the sensors and processing electronics and their interconnectivity; moreover, targeted applications only need data capture from the aforementioned image sensors, i.e., excluding the usage of any additional sensor such as distance sensor, etc. (ii) scalability of the system; various incarnations of the camera are envisioned, and the design must be scalable by construction; (iii) individual cameras with low (or limited) resolution; a Panoptic camera consisting of a large number of image sensors, each with low resolution is the favored design, in contrast with a solution consisting of few high-resolution image sensors; (iv) real-time operation is a necessity in the image capture stage as well as in an embedded early image processing stage.

This paper is structured as follows. The geometrical arrangement and the physical realization of the Panoptic camera are presented in Section 2. Section 3 theoretically demonstrates the omnidirectional sensing capability of the device by describing how to construct a virtual omnidirectional image for an observer located inside the Panoptic hemisphere. A global description of the whole Panoptic device, its viewing coverage and its minimal Full-view Coverage Distance are presented in Section 4. The analysis method is general, and can be used in the analyses of similar omnidirectional cameras which differ by the locations of the individual imagers. The architecture of the hardware platform in charge of image reconstruction and processing is presented in Section 5. Section 6 discusses the detail of its constituting hardware blocks, as well as the hardware implementation of the image processing algorithms into these blocks. Section 7 briefly addresses the calibration requirement of the Panoptic device. Several omnidirectional image reconstructions at different virtual observer location are finally presented and analyzed in Section 8 before to conclude in Section 9.

2 The Panoptic Camera Configuration

The physical realization of the omnidirectional image sensor consists of the layering of CMOS imagers on the surface of a hemisphere such that each imagers points into a determined and distinct direction.

2.1 Hemispheric Arrangement

The locations of CMOS imagers on the surface of a hemisphere follows a general coverage method of the hemisphere surface with area constant circular faces, each representing the area occupied by one camera, its package and embedded connectivity.

In order to define the spherical coordinate of each sensor location, the hemispherical surface of a unit sphere is divided into $N_{\text{flo}} + 1$ latitude floors. All circular faces located on a floor have the same latitude angle. The top most floor located on the North pole of the hemisphere only contains one circular face. The latitude angle θ_n of the n^{th} floor ($0 \leq n \leq N_{\text{flo}}$) is obtained from:

$$\theta_n = 2n\alpha_0, \quad \alpha_0 = \frac{\pi}{2(2N_{\text{flo}} + 1)}, \tag{3}$$

where α_0 is the radius of the circular face on the unit sphere such that $\theta_{N_{\text{flo}}} = \frac{\pi}{2} - \alpha_0$. Scaling this sphere allows to match α_0 with the true (half) width of each CMOS imager.

The centers $(\theta_n, \phi_{n,j})$ of the circular faces located on each latitude floor are evenly positioned according to

$$\phi_{n,j} = j\Delta\phi_n, \quad \Delta\phi_n = \frac{2\pi}{N_n}, \tag{4}$$

with $0 \leq j < N_n$ and N_n determined such that $\Delta\phi_n$ is greater than the longitudinal angle occupied by one face. According to the sine formula for spherical trigonometry [11], this last angle is given by $2\arcsin\left(\frac{\sin\alpha_0}{\sin\theta_n}\right)$, and for $n > 0$,

$$N_n = \left\lfloor \pi \;/\; \arcsin\left(\frac{\sin\alpha_0}{\sin\theta_n}\right) \right\rfloor. \tag{5}$$

Hence, the total number of centers which equals the total number of cameras in the Panoptic device, is given by $N_{\text{cam}} = \sum_{n=0}^{N_{\text{flo}}} N_n$. For instance, for $N_{\text{flo}} = 0, 1, 2$ and 3, $N_{\text{cam}} = 1, 6, 15$ and 29, respectively.

In the following, the face centers are labeled with a single index $0 \leq i < N_{\text{cam}}$, so that each center is associated to the spherical coordinates $c_i = (\theta_i, \phi_i)$, with $i = 0$ assigned to the North pole, and the mapping $i = i(n, j) = n\,N_{n-1} + j$ for $0 < n \leq N_{\text{flo}}$ and $0 \leq j < N_n$.

As an example, Fig. 1 depicts the hemispherical structure with six floors ($N_{\text{flo}} = 6$). The 6-floor hemispherical structure contains $N_{\text{cam}} = 73$ circular faces.

In parallel with the spherical coordinates of the camera centers, their corresponding expression in the 3-D coordinate system $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ centered on the hemisphere center is utilized, where $\mathbf{z}$ is identified with the vertical direction of the device, that is, $\mathbf{z}$ points toward the hemisphere North pole.

In this case, the 3-D coordinate $\mathbf{c}_i$ (distinguished by a vectorial notation) of the i^{th} camera center $c_i = (\theta_i, \phi_i)$ is given by

$$\mathbf{c}_i = r_{\odot}\,(\sin\theta_i \cos\phi_i\,\mathbf{x} + \sin\theta_i \sin\phi_i\,\mathbf{y} + \cos\theta_i\,\mathbf{z}),$$

where $r_{\odot}$ stands for the radius of the Panoptic hemisphere.

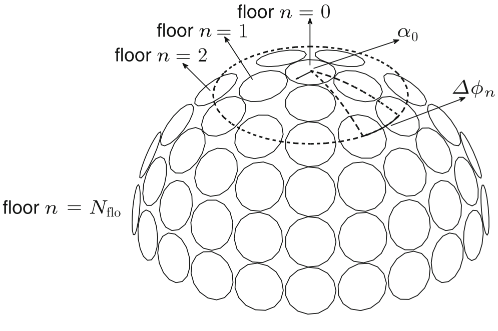


Figure 1 Hemispherical structure with six floors.

2.2 Camera Orientations

Camera positions on the Panoptic device are identified with their respective camera focal points, that is, the $i^{\rm th}$ camera is centered on $c_i = (\theta_i, \phi_i)$.

In addition to its location, each camera c_i is also characterized by three vectors: the "target" direction $\mathbf{t}_i$ pointing in the camera line of sight (focus direction), the "up" direction $\mathbf{u}_i$ providing the vertical direction in the pixel representation of the camera, and a vector $\mathbf{v}_i$ orthogonal to the two first, that is, the horizontal direction in the pixel domain. The vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ vector form an orthogonal referential for the pixel coordinates of each camera.

Given the positioning scheme defined in Section 2.1, each camera c_i (for $1 \le i < N_{\rm cam}$) is oriented so that, first, the target direction is normal to the sphere surface, that is, $\mathbf{t}_i = \mathbf{c}_i$, and second, the vectors $\mathbf{u}$ and $\mathbf{v}$ are aligned, respectively with the tangential vectors $\mathbf{e}_{\phi,i} = (\mathbf{z} \wedge \mathbf{c}_i)/\sin\theta_i$ and $\mathbf{e}_{\theta,i} = \mathbf{c}_i \wedge \mathbf{e}_{\phi,i}$ to the sphere at c_i (with $\wedge$ the common vectorial product between two vectors). For the North pole camera c_0, $\{\mathbf{t}_0, \mathbf{u}_0, \mathbf{v}_0\} = \{\mathbf{z}, \mathbf{x}, \mathbf{y}\}$ is selected.

Explicitly, considering the aforementioned placement of the cameras on the hemisphere structure of the Panoptic device, the three unit vectors of the $i^{\rm th}$ camera are obtained from

$$\begin{pmatrix}\mathbf{t}_i\\ \mathbf{u}_i\\ \mathbf{v}_i\end{pmatrix} = \begin{pmatrix}\sin\theta_i\cos\phi_i & \sin\theta_i\sin\phi_i & \cos\theta_i\\ -\cos\theta_i\sin\phi_i & -\cos\theta_i\cos\phi_i & \sin\theta_i\\ -\sin\phi_i & \cos\phi_i & 0\end{pmatrix}\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\end{pmatrix}. \quad (6)$$

Figure 2a presents the arrangement of these three vectors for each camera of a Panoptic device made of seven floors ($N_{\rm flo} = 6$). A zoomed portion of Fig. 2a is shown in Fig. 2b. The $\mathbf{u}$ vectors are facing upwards to the North pole for each camera and the $\mathbf{v}$ vectors are pointing in the counterclockwise direction of the polar angle.

2.3 Intrinsic Camera Parameters

Additional important parameters characterize the intrinsic camera properties, in addition to the location and to the orientations of each camera in the Panoptic device. Since the Panoptic device is made of a collection of identical imagers, these parameters are assumed to be identical for each camera.

The main intrinsic parameters are the focal length, denoted by $f_L > 0$, the angle-of-view (AOV) $\alpha > 0$, and the resolution of the camera, that is, the size $I_h \times I_w$ of the pixel grid.

According to a pinhole camera model [12], the focal length controls the mapping between light ray direction and pixel coordinates, while the AOV determines the limit angle around its target direction (line-of-sight) beyond which the camera is unable to record light. The camera AOV strictly depends on the direction, since the field-of-view is rectangular. However, explanations are simplified by associating the AOV to the minimum

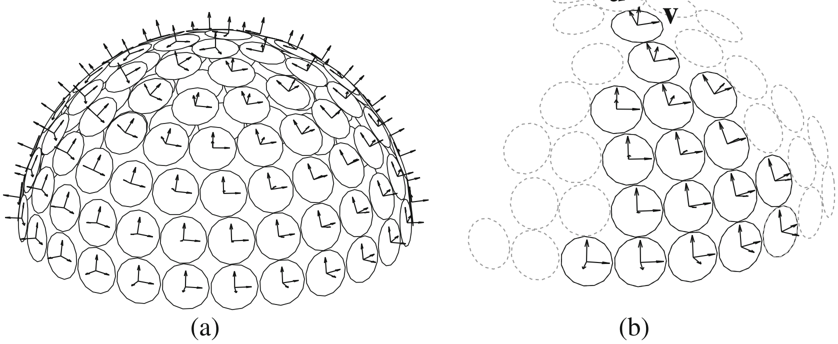


Figure 2 **a** Radial and tangential vectors identifying the target direction and the coordinates of the cameras image frame plane, **b** zoomed portion of the previous picture.

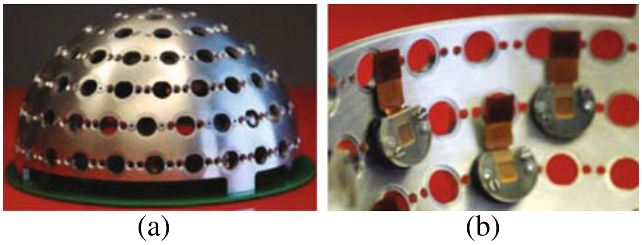


Figure 3 **a** Side view, **b** and internal view of the fabricated Panoptic camera with individual imagers.

angle-of-view disregarding the observation direction (vertical, diagonal or horizontal).

2.4 Physical Realization

A custom Panoptic camera prototype was built using a classical digital machining of an aluminum structure, and polyvinyl chloride (PVC) camera holders. The location of the cameras is based on the circular positions of the hemisphere structure with seven floors. The fabricated Panoptic camera is shown in Fig. 3. The diameter of the hemisphere is $2r_{\odot} = 129$ mm. The fabricated hemisphere structure is placed over a circular printed circuit board which provides access to embedded imagers through flexible wire connections.

The camera module utilized in the built panoptic prototype is PixelPlus PO4010N single chip CIF 368 × 304 pixels (with an effective resolution of 352×288) camera. The diagonal, vertical and horizontal angle-of-view are equal to 72.3°, 66° and 68°, respectively. Hence, $\alpha = 66\pi/180$ is assumed. The effective focal length is $f_L = 1.27$ mm.

3 Omnidirectional Vision Construction

Ideally, an omnidirectional view could be obtained by projecting light rays onto a spherical or hemispherical image sensor. Currently available silicon fabrication technologies do not enable the systematic placement of photodiodes over a spherical structure. It is shown in this section that the Panoptic camera can be used to emulate the omnidirectional vision of a *virtual* observer located anywhere inside the hemisphere by combining the information collected by each camera.

The fundamental method used in this processing is an interpolation of light information in the light ray space domain (or *light field* [13]). Constructing an omnidirectional view on a point inside the hemisphere corresponds to estimating the intensity value of all light rays in the 3-D scene that would cross the observer location point for all directions. This processing combines the available data recorded by the Panoptic imagers, and takes advantage of all the light rays crossing all the imager focal points in all the directions present in the imager angle-of-view (AOV). In this operation, the polydioptric organization of the imagers over the hemispherical structure is mandatory to collect a sufficient amount of light rays, in order to obtain a fine sampling of the light field domain [13–15].

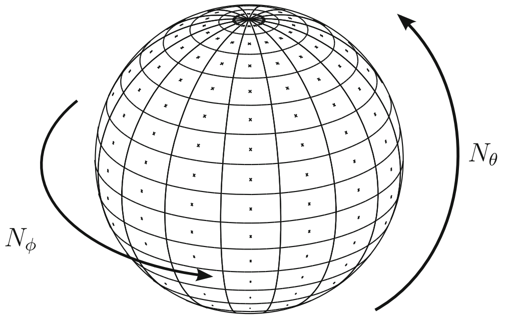


Figure 4 Discretized sphere surface $\mathcal{S}_d$ with $N_\theta = 16$ latitudes and $N_\phi = 16$ longitudes (256 pixels).

In this construction process, the omnidirectional view on a discretized sphere $\mathcal{S}_d$ of directions is estimated.[1] The surface of this sphere is pixelized into an *equiangular* grid with N_θ latitudes and N_ϕ longitudes. An example of a pixelized sphere surface with sixteen spherical pixels for N_ϕ and N_θ is shown in Fig. 4. The direction of each pixel in the omnidirectional view is identified by the unit vector $\boldsymbol{\omega} \in \mathcal{S}_d$ or by the spherical coordinates $\omega = (\theta_\omega, \phi_\omega)$.

The construction of the virtual omnidirectional view $\mathcal{L}(\mathbf{q}, \boldsymbol{\omega}) \in \mathbb{R}$ located on $\mathbf{q}$ for each omnidirectional pixel in direction $\boldsymbol{\omega}$ is performed in two algorithmic steps. First, all cameras participating to the construction, i.e., having $\boldsymbol{\omega}$ in their AOV, are determined. Since $\boldsymbol{\omega}$ does not necessarily belong to each camera pixel grid, a first level of interpolation is required to obtain the light intensity in that direction for each contributing camera. Second, an additional interpolation is performed in the space of light rays given by the direction $\boldsymbol{\omega}$ and passing through the camera origins. The aim is to estimate the intensity of a parallel light ray crossing the virtual camera center from these light rays. For the sake of simplicity, it is assumed that this center is localized in the hemisphere center, that is, $\mathbf{q} = 0$, but the same developments can be generalized to any other observation point. Following shorthand is used $\mathcal{L}(\boldsymbol{\omega}) = \mathcal{L}(0, \boldsymbol{\omega})$.

[1] Notice that this choice is arbitrary, and some alternate spherical pixelization could have been selected (e.g., HEALPix [16]).

3.1 First Algorithmic Step

Given the direction $\boldsymbol{\omega} \in \mathcal{S}_d$ where the intensity $\mathcal{L}(\boldsymbol{\omega})$ of the virtual view must be estimated, determining the cameras having $\boldsymbol{\omega}$ in their AOV amounts to determine all the camera index $0 \leq i < N_{\text{cam}}$ such that

$$\omega_{t_i} = \boldsymbol{\omega} \cdot \mathbf{t}_i > \cos\frac{\alpha}{2}, \tag{7}$$

where α is the camera AOV. The angle between $\boldsymbol{\omega}$ and $\mathbf{t}_i$ is controlled to be smaller than $\frac{\alpha}{2}$.

Figure 5a illustrates an example of selecting the contributing cameras for a typical pixel direction $\boldsymbol{\omega}$ in a Panoptic device. The edges of the contributing camera faces are bold in Fig. 5a. The number of contributing cameras is seventeen in this example. Figure 5a and b will be used as a referring example in the continuation of this Section.

Having found the contributing cameras, the next step consists in translating the direction $\boldsymbol{\omega}$ in pixel coordinates of these cameras.

Using the pinhole camera model [12], the contributing two dimensional position (x_{u_i}, x_{v_i}) on the i^{th} camera image plane (which is identified by coordinate unit vectors $\mathbf{u}_i$ and $\mathbf{v}_i$) is expressed as:

$$(x_{u_i}, x_{v_i}) = -(\boldsymbol{\omega} \cdot \mathbf{u}_i, \boldsymbol{\omega} \cdot \mathbf{v}_i)\,\frac{f_L}{\omega_{t_i}}, \tag{8}$$

where f_L represents the camera focal length in (8). As an example, the contributing positions of an assumed dot on the image frames of the contributing cameras in Fig. 5a are shown in Fig. 5b with bold mark points. The image frame planar arrays shown in Fig. 5b are fictitious and are only drawn for illustrative purposes figuring the image sensor array. Considering the selected direction of observation $\boldsymbol{\omega}$, as depicted in Fig. 5a, the contribution of each imager into this direction is found at different (x_u, x_v) coordinate locations, since the imagers are not aligned. For example, the contribution of imager A in Fig. 5b into the direction of observation $\boldsymbol{\omega}$ consists of pixel A_ω.

The contributing position on the image frame of each contributing camera is likely not to coincide with an exact pixel location of a camera image frame. The light intensity of the contributing position is the desired quantity. The light intensity of the contributing position can be estimated by the light intensity of the nearest actual pixel location to the contributing position. An alternate method consists of using interpolation among light intensities of the actual neighboring pixels of the contributing position. A detail discussion of these techniques is beyond the scope of this study, and the reader is referred to related literature [17].

As a final result, the first algorithmic step estimates the values $\mathcal{L}(\boldsymbol{\omega}; \mathbf{c}_i)$ for each contributing camera i satisfying (7), which is the intensity of all light rays parallel to $\boldsymbol{\omega}$ as recorded by these cameras.

3.2 Second Algorithmic Step

Having obtained the intensity $\mathcal{L}(\mathbf{c}_i, \boldsymbol{\omega})$ of all the contributing cameras i in direction $\boldsymbol{\omega}$, a second algorithmic step is required in order to acquire the final intensity $\mathcal{L}(\boldsymbol{\omega})$ for the virtual omnidirectional vision on the hemisphere center.

Assuming that the light intensity remains constant on the trajectory of any light ray (*constant light flux* – CLF – assumption), i.e., $\mathcal{L}(\mathbf{q}, \boldsymbol{\omega}) = \mathcal{L}(\mathbf{q} + \lambda\boldsymbol{\omega}, \boldsymbol{\omega})$ for any $\lambda \in \mathbb{R}$, the *orthographic* plane is defined such that for a given direction $\boldsymbol{\omega}$, light ray intensity only varies in the plane perpendicular to $\boldsymbol{\omega}$.

The orthographic plane is indicated as the "$\boldsymbol{\omega}$-plane" for the direction $\boldsymbol{\omega}$ in Fig. 6a. For clarity, Fig. 6a is redrawn with a change of point of view in Fig. 6b. Following the CLF assumption, the $\boldsymbol{\omega}$-plane can be considered crossing the virtual camera center, which is the center of the sphere in our simplified analysis. The sphere center is marked by a bold point in Fig. 6a and b. The light rays of direction $\boldsymbol{\omega}$ recorded by each contributing camera intersect the $\boldsymbol{\omega}$-plane in points that are the projection of the camera focal points on this plane.

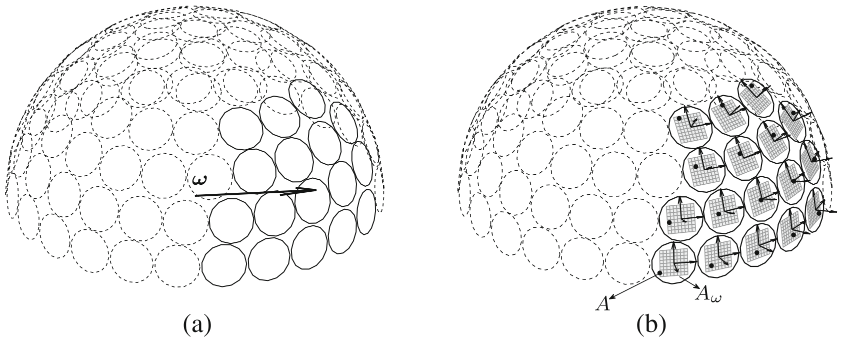


Figure 5 **a** Cameras contributing to pixel position $\boldsymbol{\omega}$, **b** contributing positions on the image frame of the contributing cameras to pixel position $\boldsymbol{\omega}$.

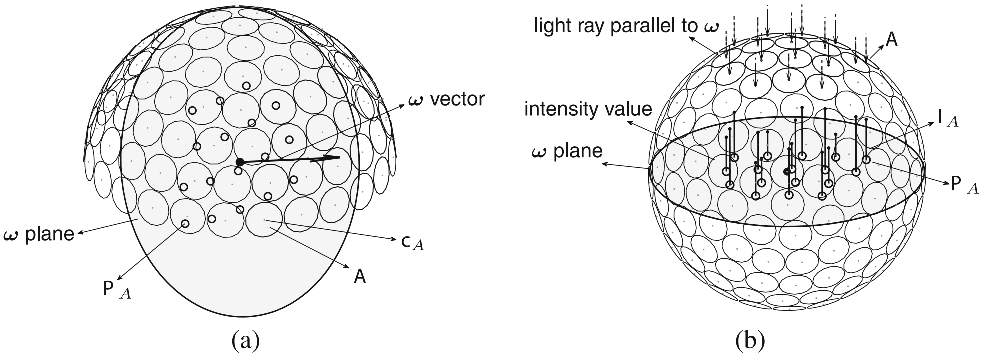


Figure 6 **a** Projection of the camera centers contributing in direction $\boldsymbol{\omega}$ onto the planar surface normal to $\boldsymbol{\omega}$, **b** another view point of the camera centers projection onto the planar surface normal to $\boldsymbol{\omega}$.

The projected points of the contributing camera positions in $\boldsymbol{\omega}$ onto the $\boldsymbol{\omega}$ are highlighted by hollow points in Fig. 6a and b. Following the CLF assumption, each projected camera position P_{c_i} on the planar surface is assigned an intensity value $\mathcal{L}(\mathbf{c}_i, \boldsymbol{\omega})$ (Fig. 6b).

As an example, the camera indicated as A of position c_A in Fig. 5b contributes in direction $\boldsymbol{\omega}$. The contributing pixel position on the image frame of the camera A is denoted as A_ω in Fig. 5b. The projection of the camera center c_A onto the $\boldsymbol{\omega}$-plane is indicated as P_A in Fig. 6a and b. The position $P_A = P_{c_A}$ on the $\boldsymbol{\omega}$-plane is assigned the intensity value $I_A = \mathcal{L}(\mathbf{c}_A, \boldsymbol{\omega})$ (Fig. 6b). The intensity value I_A is the light intensity observed by the pixel position A_ω of the image frame of camera A (Fig. 5b). The same process is applied to the sixteen other participating cameras.

For the seventeen intensity values in our illustration, the intensity value which is observed into direction $\boldsymbol{\omega}$ can be estimated by the aggregate participating cameras through a two-dimensional interpolation, i.e., using an algorithmic aggregate of the seventeen intensity values, or a subset of them to extract one unique intensity value. The easiest interpolation scheme amounts to assign the intensity of the nearest neighboring position to the spherical center $\boldsymbol{\omega}$-plane as the intensity observed by the spherical center itself.

Complex interpolation techniques for the aforementioned purpose exist but the discussion of the latter techniques is beyond the scope of this work and the reader is referred to [17] for more details.

Sections 5 and 6 focus on the introduction of the platform designed for the emulation of the Panoptic camera and the implementation of the omnidirectional vision construction algorithm on this platform.

3.3 Virtual View Construction at Any Location

In order to construct the spherical omnidirectional view scene, the intensity of the center of sphere in all $\boldsymbol{\omega}$ directions must be derived following the two algorithmic step method presented in this Section. Generalization of the omnidirectional view construction to any observing point located inside the hemisphere is straight forward. This is conducted by projecting the virtual observing point inside the hemisphere on the orthographic $\boldsymbol{\omega}$-plane, and estimating the light intensity on its projected location through a two-dimensional interpolation rather than the center point of the hemisphere.

4 Panoptic Camera Coverage Analysis

In this Section, a systematic approach is presented for analyzing the capabilities and the limitations of the Panoptic device when used as a omnidirectional imager. Although the approach is demonstrated for a Panoptic device with the specific camera positioning scheme defined by Eqs. 3 and 4, the presented method is equally applicable to other camera positioning schemes and any number of cameras as well.

As presented in Section 3 and since the Panoptic device is polydioptric, the omnidirectional view of a virtual observer located in any point inside the hemisphere can be estimated. However, the construction of an omnidirectional view using the Panoptic camera requires that all the AOVs of the cameras positioned on the hemispherical structure cover the surrounding environment of the hemisphere.

Therefore in Section 4.2 the Full-view Coverage Distance (FCD) of the Panoptic device is introduced in order to express the minimal distance which enables the Panoptic device to sample a fully covered omnidirectional view. This distance should be as close as possible to the hemisphere radius, as it is the case for an ideal hemispherical image sensor. However, the FCD is actually larger for the Panoptic device, and this distance depends on the number of cameras covering the hemisphere, their distribution on the hemisphere, and each camera AOV. The FCD is a fundamental limitation of the multi-aperture Panoptic camera, and its rigorous derivation is presented in the following. To this aim, the Panoptic Voronoi diagram and the isotropic Angle-of-View are introduced as useful geometrical tools for estimating the FCD.

4.1 Panoptic Camera Voronoi Diagram

In order to emulate an ideal hemispherical image sensor with the Panoptic device, each pixel, or each direction of the virtual spherical image sensor must be observable by at least one camera of the Panoptic device.

In order to easily characterize this criterion, the surface of the Panoptic device hemisphere is partitioned in a set of cells centered on the camera locations. Each cell is defined as the set of all points on the hemisphere which are closer to the camera location contained in the cell than to any other camera positions. The boundaries of the cells are determined by the points equidistant to two nearest sites, and the cell corners (or nodes) to at least three nearest sites. This particular partitioning falls into the category of a well established geometry concept known as Voronoi diagram (or Voronoi tessellation) [18], which is applied on a spherical surface. Many standard algorithms exist for generating a Voronoi diagram [19].

If the virtual view to be reconstructed is centered on the hemisphere center ($\mathbf{q} = 0$ in Section 3), each point on the Panoptic hemisphere can be identified with a particular direction of the virtual view. Hence, each pixel intensity of the virtual view depends on the intensity recorded by the camera residing in its respective Voronoi cell, at least.

The *radius* of a cell is defined as the distance of the farthest point to the cell center (or site). Therefore, considering the largest cell (in term of radius), its farthest point is also the direction of the virtual view that is the less "covered" by the Panoptic cameras. Let us denote this last covering point is denoted as ω_{LCP}.

As an illustration, the Voronoi diagram of the hemisphere structure of Fig. 3a is depicted in Fig. 7. Each Voronoi cell which has the form of a spherical convex polygon encompasses a Voronoi site corresponding to one camera center position. The Voronoi nodes are indicated on the vertex of Voronoi cells with circles in Fig. 7. The segments of the largest Voronoi cell which contains the last covering point are shown with bold lines in Fig. 7b. The last covering point ω_{LCP} is highlighted with a bold circle in Fig. 7b.

Figure 7 reveals that the Voronoi sites positioning provided by Eqs. 3 and 4 does not yield a symmetric Voronoi diagram with identical Voronoi cells. In general, a symmetric Voronoi diagram on a spherical surface is not achievable [19]. Positioning schemes exist which yield a close-to-symmetry Voronoi diagram for a spherical surface. Nevertheless, the positioning scheme defined by Eqs. 3 and 4 enables the inclusion of a constant radius disk representing the occupation of a camera.

4.2 Panoptic Camera Full-View Coverage Distance

The construction of a virtual omnidirectional view from the Panoptic data requires that each pixel (direction) of the omnidirectional vision be observed by at least one of the cameras of the Panoptic device (Section 3).

This has an impact on the minimal distance below which a punctual object is not observable by the panoptic device under all circumstances. By modeling each camera in its simple form as pinhole camera [12] with an isotropic angle-of-view (as shown in an example

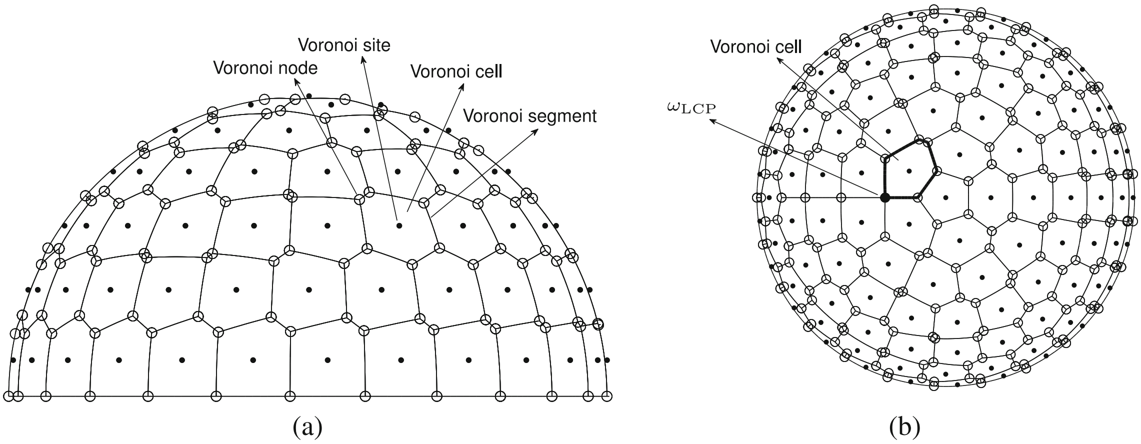


Figure 7 **a** Side view of the Voronoi diagram of the pixelized hemispherical structure with seven floors, **b** top view of the Voronoi diagram of the same structure.

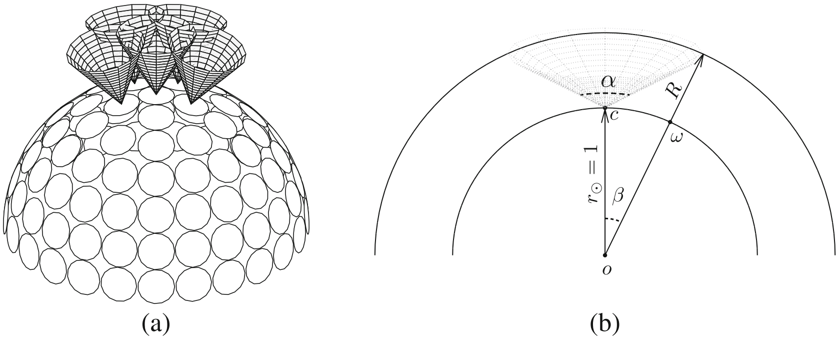


Figure 8 **a** Pinhole cameras with isotropic angle-of-view on the Panoptic device; for the sake of clarity, only five cameras are modeled in this figure, **b** intersection of the isotropic angle-of-view boundary and the spherical surface of radius R.

in Fig. 8a), a close form expression of this distance is derived.

The minimal distance R to the hemisphere center o at which a (punctual) object starts to be observed by one camera (located on the North pole for the sake of clarity) when this object makes an angle β with the camera location c is first derived (Fig. 8b). Using trigonometric relations pertaining to the triangle formed by o, c and the object location yields:

$$\frac{R}{r_\odot} = \frac{\sin(\frac{\alpha}{2})}{\sin(\frac{\alpha}{2} - \beta)}, \tag{9}$$

where $r_\odot$ is the Panoptic hemisphere radius, α the camera AOV and β assumed smaller than $\alpha/2$, that is, $\frac{\alpha}{2} - \beta \geq 0$. Since the latter expression is not dependent on the camera location, it is valid for any camera provided that β measures the angle between the object and the camera center.

When the distance R tends to infinity in the previous expression:

$$\lim_{R \to \infty} \beta = \frac{\alpha}{2}, \tag{10}$$

the value of β tends to $\frac{\alpha}{2}$. Hence, a point of interest for the spherical scene construction can only be seen by a camera if the angle between the radial vector of its position and that of the center position of the target camera is less than half of the camera AOV.

Two important quantities characterizing the Panoptic device configuration are derived using expressions 9 and 10.

First, the minimal AOV that any camera must have to guarantee that any object at an infinite distance be observed by at least one camera is derived. In Section 4.1, the Voronoi diagram of the Panoptic camera arrangement provides the last covering point ω_{LCP}, that is, the direction which is the farthest to any camera location on the sphere of the Panoptic device. Denoting $\beta_{\max}$ the angle between ω_{LCP} and its closest camera center, the minimum AOV which is required for a Panoptic device is obtained using the asymptotic behavior (Eq. 10),

$$\alpha_{\min} = 2\,\beta_{\max}. \tag{11}$$

Second, the Full-view Coverage Distance (FCD) of the Panoptic device (normalized to the case $r_\odot = 1$) is obtained utilizing Eq. 9 through Eq. 12 choosing $\alpha > \alpha_{\min}$ as the AOV for the cameras of the Panoptic device and assuming that all the cameras are identical.

$$R_{\text{FCD}} = \frac{\sin\frac{\alpha}{2}}{\sin(\frac{\alpha}{2} - \beta_{\max})}. \tag{12}$$

The last column in Table 1, named $\alpha_{\min}$, contains the calculated value of the minimum required AOV for the cameras of the Panoptic device with $N_{\text{flo}} + 1$ latitude floors and a unit radius. The second and third columns of Table 1 include the positions of the cameras for the observation of the last covering points and the last covering points positions respectively.

The minimum AOV required for full coverage versus the number of latitude floors is presented in Fig. 9a. As the number of latitude floors increase, which is equivalent to increasing the number of cameras, the minimum required AOV for full-view coverage decreases (Fig. 9a). A trade-off between the minimum required AOV $\alpha_{\min}$ and the number of cameras positioned on the Panoptic device is hence observed.

Table 1 Last coverage point and minimum angle of view for full coverage of a Panoptic device with $N_{\text{flo}} + 1$ latitude floors.

N_{flo}	c	ω_{LCP}	$\alpha_{\min}$
1	(0.27, 0.82, 0.50)	(0.81, 0.59, 0.00)	92°
2	(−0.48, 0.35, 0.81)	(−0.77, 0.00, 0.64)	56°
3	(0.55, −0.80, 0.22)	(0.35, −0.93, 0.00)	38°
4	(−0.28, 0.20, 0.94)	(−0.45, 0.00, 0.89)	32°
5	(−0.23, −0.17, 0.96)	(−0.37, 0.00, 0.93)	26°
6	(−0.19, 0.14, 0.97)	(−0.32, 0.00, 0.95)	22°
7	(0.06, 0.20, 0.98)	(−0.08, 0.26, 0.96)	19°
8	(0.06, 0.17, 0.98)	(−0.07, 0.23, 0.97)	17°
9	(0.05, 0.16, 0.99)	(−0.07, 0.20, 0.98)	15°
10	(0.05, 0.14, 0.99)	(−0.06, 0.19, 0.98)	14°

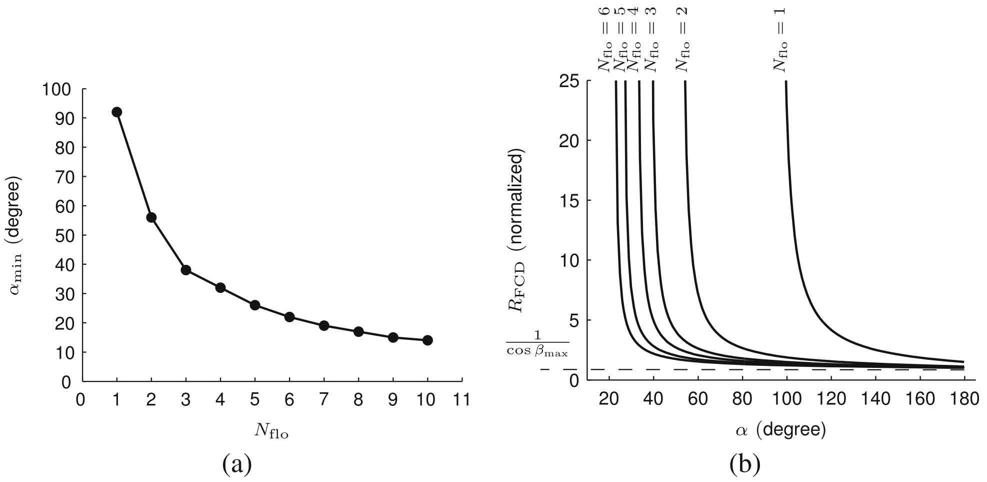


Figure 9 **a** Minimum AOV for the cameras of the Panoptic device with $N_{\text{flo}} + 1$ latitude floors, **b** R_{FCD} vs AOV of the Panoptic device.

The full-view coverage distance R_{FCD} for several Panoptic devices with $N_{\text{flo}} + 1$ latitude floors versus the AOV of the cameras are depicted in Fig. 9b. As the number of latitude floors increase and hence the number of cameras, identical full-view coverage distance is reached at smaller AOVs. A similar trend is observed for all Panoptic devices, as the AOV of their cameras increase their full-view coverage distance decrease.

The minimum full-view coverage is obtained if $\alpha = 180° = \pi$, which corresponds to a Fish-Eye camera AOV. In that case, the minimum FCD is:

$$R_{\text{FCD, min}} = \frac{1}{\sin(\frac{\pi}{2} - \beta_{\max})} = \frac{1}{\cos\beta_{\max}}. \qquad (13)$$

5 FPGA Development Platform

The Panoptic system is designed with the aim of having its own custom ASIC imagers with integrated intra and inter imager signal processing features and integrated signal processing ASIC cores dedicated for omnidirectional imaging and its applications. As a *priori* for the aforementioned purpose, a hardware emulation platform has been designed and developed based on Field Programmable Gate Array (FPGA) for the practice of implementing, containing the Panoptic device and its applications in a real-time environment, and qualifying it for an ASIC solution.

A FPGA based system which supports a Panoptic camera with up to 100 imagers generating 16 bit common intermediate format (i.e., CIF 352×288 pixels) images at 25 frame per second rate is devised. This system receives an aggregate bit rate of 3.8 Gb/s.

Prior to the development of the system, a careful feasibility study has been carried out, focusing on the system level required hardware specifications in terms of image acquisition rate, data transmission bandwidths, image processing rate, memory bandwidth and capacity, required level of architectural parallelism, FPGA pin count, connectivity, which conducted to the development of a layered system architecture, shown in Fig. 10a.

The system consists of four layers, (i) layer A: 100 imagers with programmable resolution, up to CIF, (ii) layer B: five concentrator FPGAs, handling local image processing over 20 imagers in parallel, each, (iii) layer C: one central FPGA which processes the final omnidirectional image construction, based on data transmitted in parallel from the concentrators, (iv) layer D: a PC is in charge of the applicative layer consisting of displaying the operation results transmitted from the central FPGA. The PC is not a mandatory block in the system which is autonomous; it is only used to display results in the prototype implementation. In the final application embedding the Panoptic camera, real time display capability or data communication capabilities shall be provided.

5.1 Concentrator FPGA

An FPGA board has been designed utilizing Xilinx Virtex5 XC5VLX50-1FF1153C as the concentrator FPGA module. Each concentrator FPGA board supports up to 20 imagers with 16 input/output lines, each. The concentrator FPGA board contains two zero bus turn around (ZBT) SRAMs with minimum capacity to hold twenty 16-bit color images with CIF frame size, and an operating bandwidth of 166 MHz. High-speed LVDS connections are provided for the concentrator FPGA board as a mean for data and control signal communication with the central FPGA module.

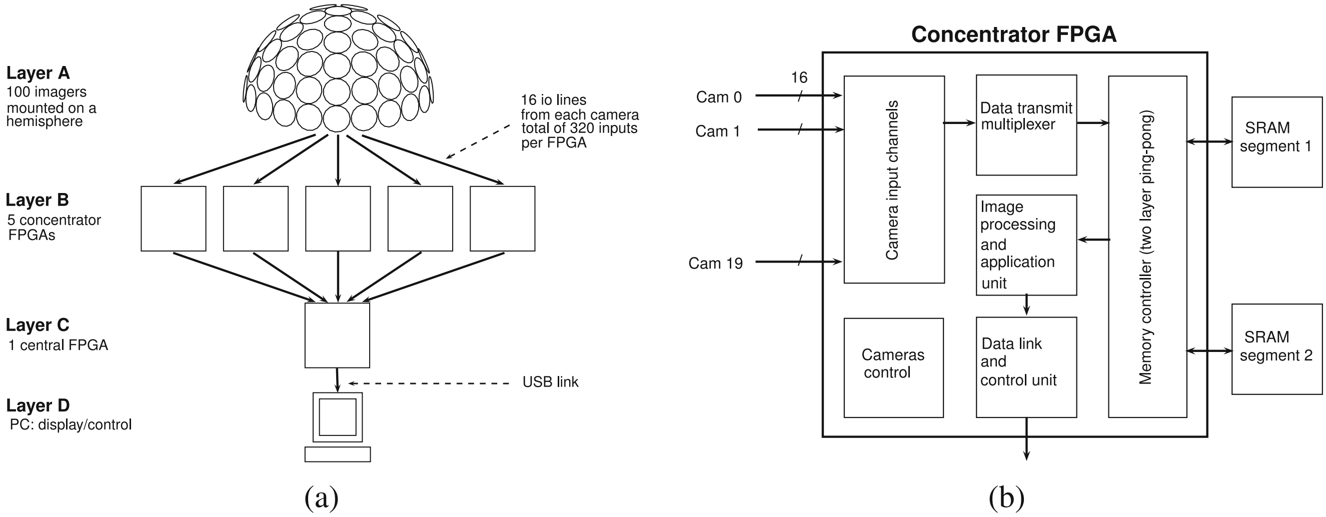

Layer A
100 imagers mounted on a hemisphere
16 io lines from each camera total of 320 inputs per FPGA
Layer B
5 concentrator FPGAs
Layer C
1 central FPGA
USB link
Layer D
PC: display/control
(a)
Concentrator FPGA
Cam 0
Cam 1
Cam 19
16
Camera input channels
Data transmit multiplexer
Image processing and application unit
Memory controller (two layer ping-pong)
SRAM segment 1
SRAM segment 2
Cameras control
Data link and control unit
(b)

Figure 10 **a** Architecture of the full hardware system and, **b** architecture of a concentrator FPGA.

The architecture of the concentrator FPGA system is depicted in Fig. 10b. The concentrator FPGA consists of five blocks. The arrow lines depicted in Fig. 10b demonstrate the image data flow inside the concentrator FPGA. Image data originating from the cameras enters the concentrator FPGA via the camera channel input block. The data transmit multiplexer block multiplexes the 20 input camera channels and passes the timed multiplexed data to the memory controller block. The memory controller block stores the incoming image frame data from the 20 cameras inside one of the SRAMs; at the same time it also retrieves the previously stored images from the other SRAM and hands it over to the image processing and application unit block. The SRAMs swap their role (i.e., one being written and one being read) each time a new image frame data is fully received. The image processing and application unit block is in charge of anticipated signal processing. In addition, some basic functionalities such as real-time single-channel image capture, simultaneous capture of twenty images, single-channel video display, etc are also considered for this block. The image processing and application unit block hands over its processed image data to the data link and control unit block. The data link and control unit block is in charge of transmitting the processed image data to the central FPGA module and servicing the control data requests received from the central FPGA module. To support the programmability feature of the cameras, a camera control block is also considered. The central FPGA can access this block through the data link and control unit block.

5.1.1 Concentrator FPGA Performance

The concentrator FPGA functionality is categorized into two major tasks, regarding the captured image data. One is related to the multiplexing of the camera input channels and the other to the image processing application. Each of these operations imposes a minimum performance limit to the concentrator FPGA. The maximum of the two is considered as the minimum performance limit of the concentrator FPGA.

The concentrator FPGA must multiplex the incoming image data from 20 cameras. The cameras output their image data on a per-line basis, assuming the synchronization of all the 20 cameras connected to the concentrator FPGA. The concentrator FPGA first captures the incoming line from all the 20 cameras. While receiving the next line from the cameras, the concentrator FPGA also transmits the multiplexed version of the received previous line to one of the SRAMs. Thus the amount of time taken by the concentrator FPGA to transmit the multiplexed version of the received image data lines must be equal or less than the amount of time it takes for a single camera (assuming all the cameras to be the same) to transmit one line of image data. In mathematical form, this is expressed as:

$$N_{\mathrm{cam}} \times \frac{I_w}{F_{\mathrm{fpga}}} \leq \frac{C_w}{F_{\mathrm{cam}}}. \tag{14}$$

In Eq. 14, I_w represents the frame width of the image, F_{fpga} the concentrator FPGA clock frequency, N_{cam} the number of cameras interfaced to the concentrator FPGA, C_w the cameras frame width and F_{cam} the rate at which the cameras transmit their pixel data to the outside world. The first minimum required performance of the concentrator FPGA is obtained by solving the inequality:

$$N_{\text{cam}} \times \frac{I_w}{C_w} \times F_{\text{cam}} \leq F_{\text{fpga}}. \qquad (15)$$

Another performance criterion reflects the amount of time a concentrator FPGA spends to conduct an image processing application. Irrespective of the type of the application, the real-time feature of the system requires that the image processing time be less than or equal to the amount of time a single camera spends to generate one full frame. The amount of time needed for a typical camera to generate one full frame is obtained from the frame rate. Hence the second performance requirement is obtained from

$$T_{pc} \leq \frac{1}{f_{\text{ps}}}, \qquad (16)$$

where f_{ps} is the camera frame per second rate, and T_{pc} is the image processing application process time. The value of T_{pc} is dependent on the concentrator FPGA operating clock frequency F_{fpga} and the architecture designed to conduct the image processing.

5.1.2 Concentrator FPGA implementation

The proposed architecture with only basic functionality support for its image processing and application unit block has been implemented and functionally confirmed on the concentrator FPGA. The latter concentrator FPGA must support 20 CIF imagers with a frame rate of 25 per second, translating to an aggregate bit rate of 0.75 Gb/s. The imagers frame width is 464 pixels and their required operating frequency to produce a 25 frame per second output is 7.5 MHz. The minimum required operating frequency to support the multiplexing stage is derived using Eq. 15 at 114 MHz. The maximum operating frequency estimated by Xilinx Synthesis Tool (i.e., XST) for this architecture is 212 MHz. Table 2 summarizes the device utilization for the concentrator FPGA. Sixty five percent (65%) of logic resources and 80% of the allocated memory blocks remains free for further development of the image processing and application unit block.

Table 2 Concentrator FPGA device utilization summary.

Resources	Used	Available	Utilization
Occupied slices	2445	7200	34%
Slice registers	4608	28800	16%
BlockRAM/FIFO	10	48	20%

5.2 Central FPGA

The main task devoted to the central FPGA consists of receiving data processed by the concentrator FPGAs, apply the final image processing stage, and transfer the final results to a PC through a USB link for displaying. The central FPGA board has been developed based on the concentrator board architecture, thus forming a modular system. The performance requirement of the central FPGA module depends on the rate of the processed data which it receives from the concentrator FPGAs and the maximum local processing time (which is essentially expressed as Eq. 16) needed to conduct the final image processing stage.

6 Hardware Implementation

This Section elaborates the hardware design of the architectural blocks presented in Fig. 10b. For the purpose of clarity and demonstration of the scalability of the system, the hardware design concept is primarily elaborated for a single FPGA board with 20 camera support, assuming that one single FPGA board acts as concentrator and central FPGA at the same time. A methodology is presented to scale the design on multiple FPGA boards to support higher number cameras for the Panoptic system.

6.1 Camera Input Channel Block

The camera input channel block is in charge of interfacing the camera modules of the Panoptic system with the FPGA. A camera module is a packaged camera with an embedded lens and an external connector interface. The interface block diagram of the camera module is shown in Fig. 11a. The master clock and the system reset of the camera module are denoted as CLK and RST respectively in Fig. 11a. The camera

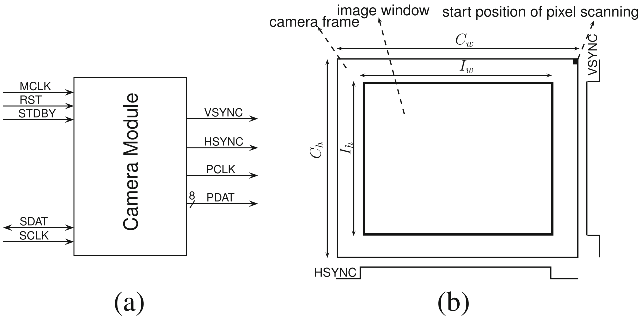


Figure 11 **a** Camera module interface, **b** camera frame.

module enters the power standby mode by activating the STDBY input. A two wire serial interface (i.e., I^2C) is available for writing and reading the internal register settings of the camera module. The serial interface data and clock buses are denoted SDAT and SCLK respectively. The PDAT is the 8 bit pixel data output of the camera module. The PCLK is the pixel clock output of the camera module. The PDAT can be latched by external devices (FPGA in our case) at the rising or falling edge of PCLK. The polarity of the PCLK can be controlled. The VSYNC outputs the vertical synchronization pulse which indicates the start of a new frame. The HSYNC outputs the horizontal synchronization pulse. The HSYNC is active for the horizontal window of interest.

The frame of the camera module has a rectangular geometry with C_h pixel rows and C_w pixel columns. A portion of camera frame pixels are effective image pixels. The effective pixels constitute the image capture window of the camera. The image window of a camera frame has I_h pixel rows and I_w pixel columns. A conceptual camera frame is depicted in Fig. 11b. Standard image window sizes (i.e., $I_w \times I_h$ pixels) are available for camera modules such as VGA (i.e., 640×320), QVGA (i.e., 320×240) and CIF (i.e., 352×280). The frame sizes of C_w and C_h are camera-dependent and vary from one camera model to another camera model. The camera module used in the presented Panoptic device is a PixelPlus PO4010N single chip CIF camera. This camera has a frame size of 464×324 (i.e., $C_w \times C_h$ pixels).

In the PO4010N camera module, pixel scanning starts from the top right corner of the camera frame and proceeds row by row downward; for each line, the scan direction is from right to the left. The VSYNC signal indicates the start of the image window region. The HSYNC signal indicates whether the output pixel data belongs to the image window region or not. The timing diagram of HSYNC with respect to VSYNC is provided in Fig. 12a. The timing diagram of the output pixel data with respect to the camera module pixel clock, master clock and horizontal synchronization strobe is provided in Fig. 12b. The activity of the HSYNC and VSYNC strobes in Fig. 12a and Fig. 12b are assumed high.

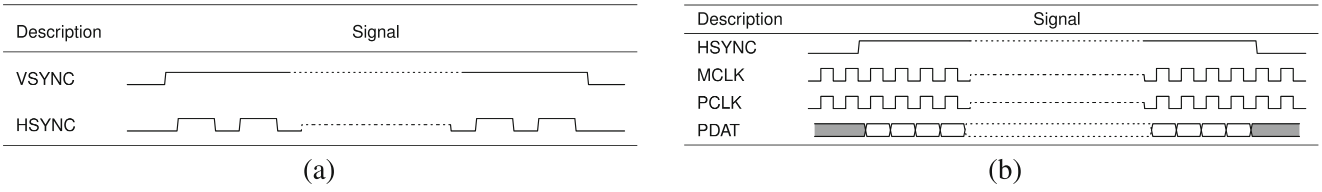
The PO4010N camera supports an 8 bit mono color (i.e., grayscale) and several 16 bit color formats such as RGB565 and RGB Bayer for the presentation of the pixel data. The frame generation rate (i.e., f_{ps}) of the camera module is directly controlled by the master clock rate (i.e., F_{mclk}). The frame rate of the camera module is derived from

$$f_{\mathrm{ps}} = \frac{F_{\mathrm{mclk}}}{2 \times C_w \times C_h}, \qquad (17)$$

where F_{mclk} stands for the frequency rate of the master clock of the camera module. The number of pixels in the column and the row of the camera frame in Eq. 17 are indicated with C_w and C_h, respectively. The maximum frame rate (i.e., f_{ps}) of the PO4010N camera module is 30 frames per second which is achieved with a master clock of 9 MHz. A frame rate of 25 per second is achieved for the PO4010N camera module with a master clock of 7.5 MHz.

6.1.1 Camera Interface Module

The input channel block of the concentrator FPGA contains twenty camera interface modules. Each camera interface module is connected to a camera module. The architecture of the concentrator FPGA is shown in Fig. 13 with additional details. The camera interface module captures the pixel data (i.e., PDAT) generated by the camera module on the rising edge of the pixel clock (i.e., PCLK) upon the condition that the horizontal (i.e., HSYNC) and vertical (i.e., VSYNC) synchronization strobes are active. When a horizontal line of pixel data is received, the camera interface module signals a transmission request via the REQ strobe to the data transmit multiplexer block. Upon reception of the acknowledge of the transmission request via the ACK strobe, the camera interface module transmits the captured line of pixel data to the data transmit multiplexer block. The capture of the next line of pixel data is not halted while the previous line of pixel data is awaiting the transmission request response, or while it is transmitted to the next block. The camera interface module preambles the pixel data sent to the data multiplexer block with additional information concerning



(a) (b)

Figure 12 **a** Timing diagram for VSYNC and HSYNC, **b** timing diagram for HSYNC, MCLK, PCLK and pixel data.

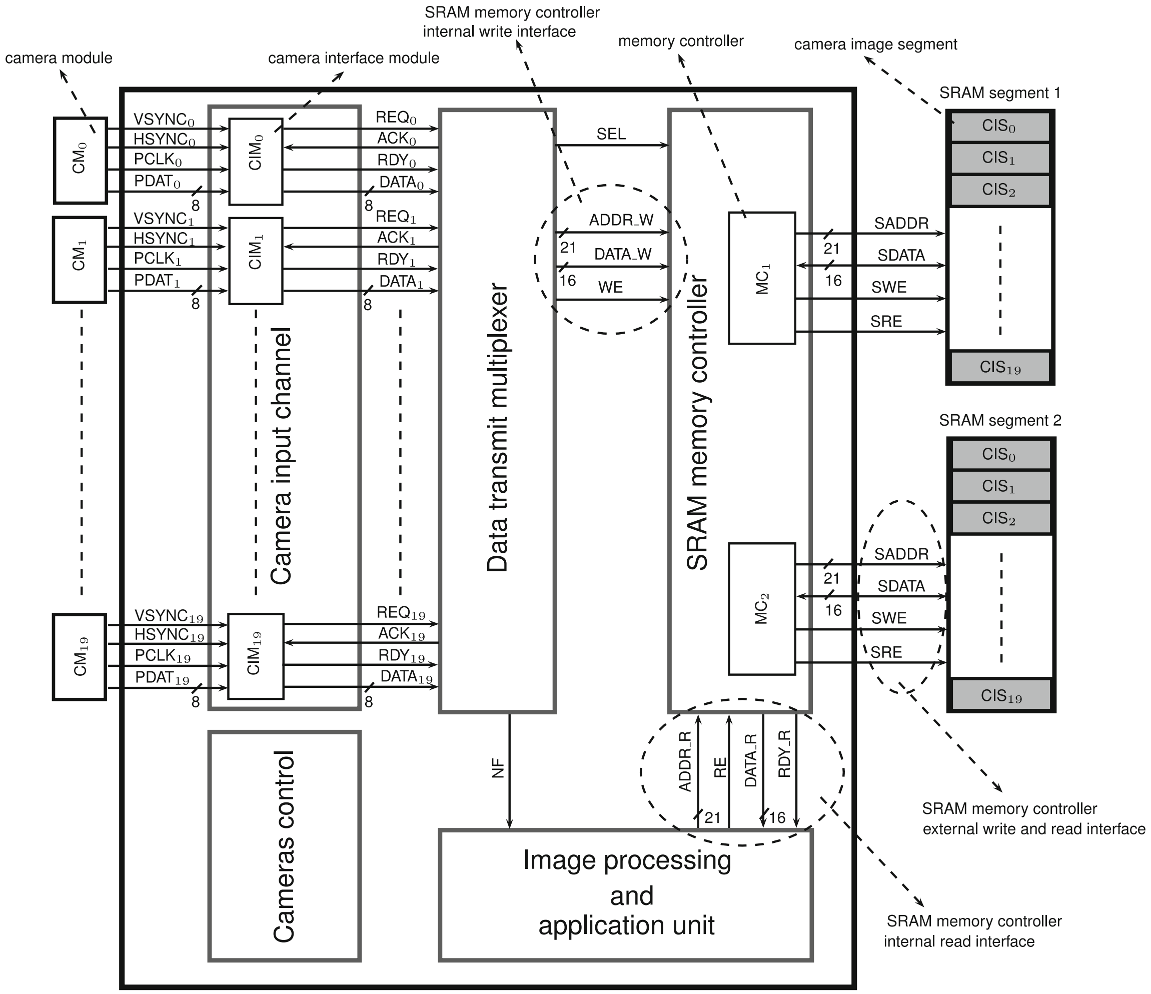


Figure 13 Architecture of the concentrator FPGA. For the sake of clarity the camera module connections to the cameras control unit and the data link and control unit are not shown in this picture.

the camera number and the line number of the camera frame. These informations are used by the data multiplexer block, in a later step. The pixel data path from the camera interface module to the data multiplexer module is indicated as the DATA bus in Fig. 13. The RDY strobe indicates the availability of a new data on the DATA bus. The camera interface module incorporates a single FPGA built-in dual-clock dual-port block RAM [20]. In addition to the temporary storage of the incoming pixel data from the camera module, the block RAM also accommodates the clock domain change from the camera module pixel clock (i.e., F_{cam}) to the FPGA system clock (i.e., F_{fpga}) [20].

6.1.2 Camera Synchronization

The proper operation of the Panoptic device applications demand; that all the cameras of the Panoptic device be synchronized. Camera synchronization for the Panoptic device is achieved by guaranteeing the same frame generation rate and the same vertical and horizontal timing diagram for all the cameras. Identical frame generation rate for all the cameras of the Panoptic device is obtained by applying the same master clock to all the cameras. Identical horizontal and vertical timing diagrams for the cameras are achieved by resetting and initializing (i.e., internal register setting) all the camera modules simultaneously.

6.2 Data Transmit Multiplexer Block

The simultaneous storage of the image frames generated by the camera modules in the single-channel access SRAM segments of the FPGA boards requires time multiplexing. The data transmit multiplexer block receives the output ports of the camera interface modules as depicted in Fig. 13. The write internal interface of the SRAM memory controller block is controlled by the data transmit multiplexer block. The SRAM controller internal write interface has a 21 bit address line (i.e., ADDR_W) and a 16 bit data line (i.e., DATA_W) with an additional write enable (i.e., WE) strobe. This interface provides the write access to the memory space of the SRAM segments. Only one SRAM segment is available for read access and one is available for write access, at any time. The selection of which SRAM segment is available for the write access is done via SEL signal strobe. The SEL signal strobe is controlled by the data transmit multiplexer block. The data transmit multiplexer block also informs the image processing and application unit block of the arrival of the new frames of the cameras through the NF signal strobe.

6.2.1 Data Transmission Request Servicing

The REQ strobes of the camera interface module is activated after the reception of each horizontal line of the camera image frame. The data transmit multiplexer block parses the REQ strobes of the camera interface modules in a rotary fashion. A priority-based parsing of the REQ strobes is not needed since all the camera modules are synchronized and activate their REQ strobes at the same time. Hence, as long as all the requests are acknowledged and serviced prior to the arrival of the next requests, no pixel data is lost. The required criterion for this matter is elaborated in the previous section and expressed in Eq. 15.

6.2.2 Camera Image Memory Storage

The memory space of the two SRAMs of the FPGA board is equally partitioned into twenty camera image segments, each having the storage size of a sixteen bit CIF image. The camera image segments are identified as CIS_i for each i^{th} camera in Fig. 13. Pixel data of camera image frames are stored starting from the top memory position and finishing at the bottom memory position of each camera assigned image segment. Each camera interface module preambles the row pixel data sent to the data transmit multiplexer block with the camera number and the row number of the frame being transmitted. The camera number is used to select the address offset of the target image segment and the row number is used to calculate the address offset position within the camera image segment. Hence the starting address of the i^{th} camera's j^{th} row pixel data transfer is derived from:

$$a(i, j) = (I_w \times I_h) \times i + I_w \times j, \tag{18}$$

where I_w and I_h are the image frame width and frame height sizes in terms of number of pixels. The camera's number (i.e., i) ranges from 0 to 19, and the row number (i.e., j) ranges from 0 to I_h-1. The I_w and I_h values for a CIF camera are 352 and 288 respectively. Expression (18) is rewritten as

$$\begin{cases} a(i, j) = b(i) + c(j), \\ b(i) = (I_w \times I_h) \times i, \\ c(j) = I_w \times j. \end{cases} \tag{19}$$

The address offset generation mechanism is implementable through addition of $b(i)$ and $c(j)$ solely. $b(i)$ in Eq. 19 is implementable with a look-up table with twenty selectable entries, each representing the address offset of a camera segment. $c(j)$ is implementable with a counter that counts at the pace of I_w and resets to zero for j=0. This is due to the fact that the row number j increments from 0 to I_h-1 synchronously, for all the cameras. Hence, the update of $c(j)$ should only occur with the arrival of the next row of pixel data.

6.2.3 Frame Update

The transmission of row number zero (i.e., $j = 0$) from the camera interface modules indicates the arrival of a new frame. Upon the arrival of a new frame the data transmit multiplexer block changes the target SRAM segment in charge of supporting the write access through the SEL strobe and informs the image processing and application unit block of the arrival of a new frame via the NF strobe.

6.3 SRAM Memory Controller Block

The SRAM memory controller block contains two zero bus turn around (ZBT) SRAM controllers which interface through an external read and write access with two 36 Mb ZBT SRAMs. This block provides an internal write access port for the data transmit block and a read access port for the image processing and application unit block. The peripheral blocks can access the memory space of the SRAM segments using these internal ports. The configuration of this block only permits the simultaneous access of one SRAM as write and the other SRAM as read. The access type can switch between the two SRAMs via the SEL strobe. For more details regarding the ZBT SRAMs and their controller design in FPGA, the reader is referred to [21].

6.4 Image Processing and Application Unit Block

This Subsection elaborates the implementation of the omnidirectional vision construction using the Panoptic camera at the architectural level. The reference algorithm used for the implementation of the omnidirectional vision construction is based on the detection of the nearest neighboring position. Hence the construction algorithm is limited to finding the closest camera position to each pixel direction $\boldsymbol{\omega}$ and the contributing pixel position on the image frame of this camera. The hardware implementation of the latter algorithm illustrates the minimum hardware requirements for the implementation of a real-time omnidirectional vision construction system using the Panoptic camera.

6.4.1 Angle Generation Module

Each virtual pixel direction $\boldsymbol{\omega}$ on the surface of the Panoptic camera is identifiable with a longitude angle ϕ_ω and a latitude angle θ_ω as follows

$$\boldsymbol{\omega} = \sin\theta_\omega \cos\phi_\omega \mathbf{x} + \sin\theta_\omega \sin\phi_\omega \mathbf{y} + \cos\theta_\omega \mathbf{z}. \tag{20}$$

The sphere surface pixelization scheme shown in Fig. 4 provides N_θ latitude and N_ϕ longitude positions. The longitude and latitude angles of each pixel direction $\boldsymbol{\omega}$ in the scheme shown in Fig. 4 is derived from

$$\begin{cases} \phi_\omega(i) = \dfrac{2\pi}{N_\phi} \times i, \quad 0 \le i < N_\phi \\ \theta_\omega(j) = \dfrac{\pi}{N_\theta} \times \left(j + \dfrac{1}{2}\right), \quad 0 \le j < N_\theta. \end{cases} \tag{21}$$

The presented scheme does not yield a homogeneous density of pixel positions over the surface of the sphere. The density of pixels is higher over areas closer to the North pole. Pixelization schemes exist which provide a homogeneous distribution of pixel positions over the surface of the sphere [16]. The latter pixelization scheme is opted over other schemes due to its regularity, ease and lower cost of implementation in terms of resources.

In order to generate the pixel direction vector $\boldsymbol{\omega}$ using Eq. 20, the ϕ_ω and θ_ω angles must be generated first. Utilizing the N bit binary presentation related to π expressed in Eq. 22 and assuming that both N_ϕ and N_θ are powers of two,

$$(a_0\, a_1\, \cdots\, a_{N-1}) = 2\pi \sum_{i=0}^{N-1} a_i \cdot 2^{-(i+1)}, \tag{22}$$

with $a_i \in \{0, 1\}$. The angles ϕ_ω and θ_ω are implementable using an accumulator for each angle. A N bit accumulator is depicted in Fig. 14a. The N bit incrementing index (i.e., step) of the accumulator is denoted as K in Fig. 14a. To generate all possible combinations of ϕ_ω and θ_ω, one accumulator is chosen to increment only when the other accumulator completes its full range cycle. This concept is illustrated in Fig. 14b. The ϕ_ω and θ_ω accumulators shown in Fig. 14b have separate incrementing indexes K_ϕ and K_θ respectively. The completion of the full range cycle of the ϕ_ω accumulator is declared to the θ_ω accumulator via an overflow detection module. The incrementing indexes K_ϕ and K_θ govern the resolution of the constructed omnidirectional vision. The highest resolution is achieved by choosing a value of one (i.e., the smallest incrementing step) for both of these indexes.

The angle generation module provides the latitude and the longitude angles of the pixel direction $\boldsymbol{\omega}$, which are denoted as PHI and THETA in Fig. 15, to the omega vector generation module. The omega vector generation module is informed of the availability of each new angle combination via the RDY_ANG strobe.

6.4.2 Omega Vector Generation Module

The omega vector generation module receives the ϕ_ω and θ_ω angles of pixel direction $\boldsymbol{\omega}$ and generates the unit vector $\boldsymbol{\omega}$ using Eq. 20. Utilization of trigonometric functions $\sin(2\pi x)$ and $\cos(2\pi x)$ is mandatory for the calculation of the $\boldsymbol{\omega}$ vector from ϕ_ω and θ_ω angles. The implementation of trigonometric functions of sine and cosine have been the focus of the direct digital frequency synthesizers (i.e., DDFS) for the past

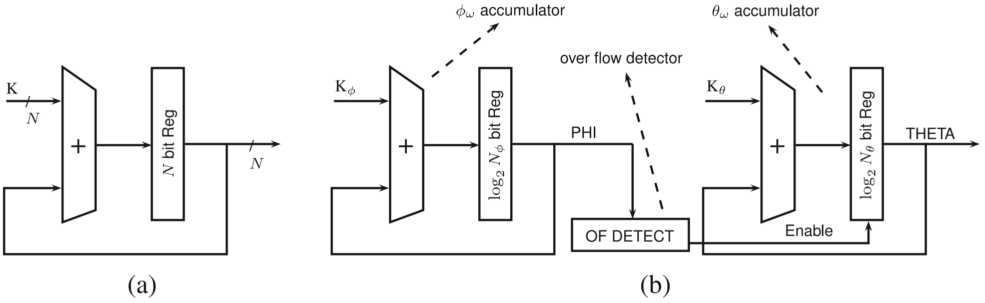


Figure 14 **a** An N bit accumulator, **b** ϕ_ω and θ_ω generation accumulators.

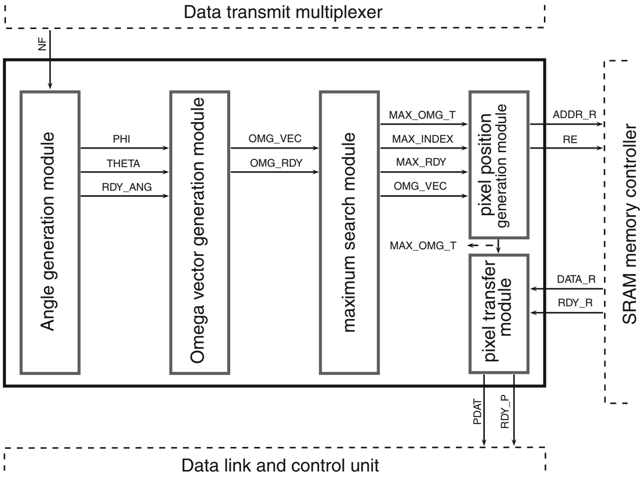


Figure 15 Image processing and application unit block diagram. For the sake of clarity, only the pertaining sub-blocks for omni-direction vision construction are shown.

decades. Hence many algorithms have been developed for the purpose of the implementation of the basic trigonometric functions. Look up table (LUT) based algorithms [22], the CORDIC algorithm [23, 24] and polynomial approximation based algorithms [25] are the major categories in this field. The selection of an algorithm for the implementation of the sine and cosine calculator (i.e., SCC) module is arbitrary and dependent on system requirements such as performance and available resources in the target platform.

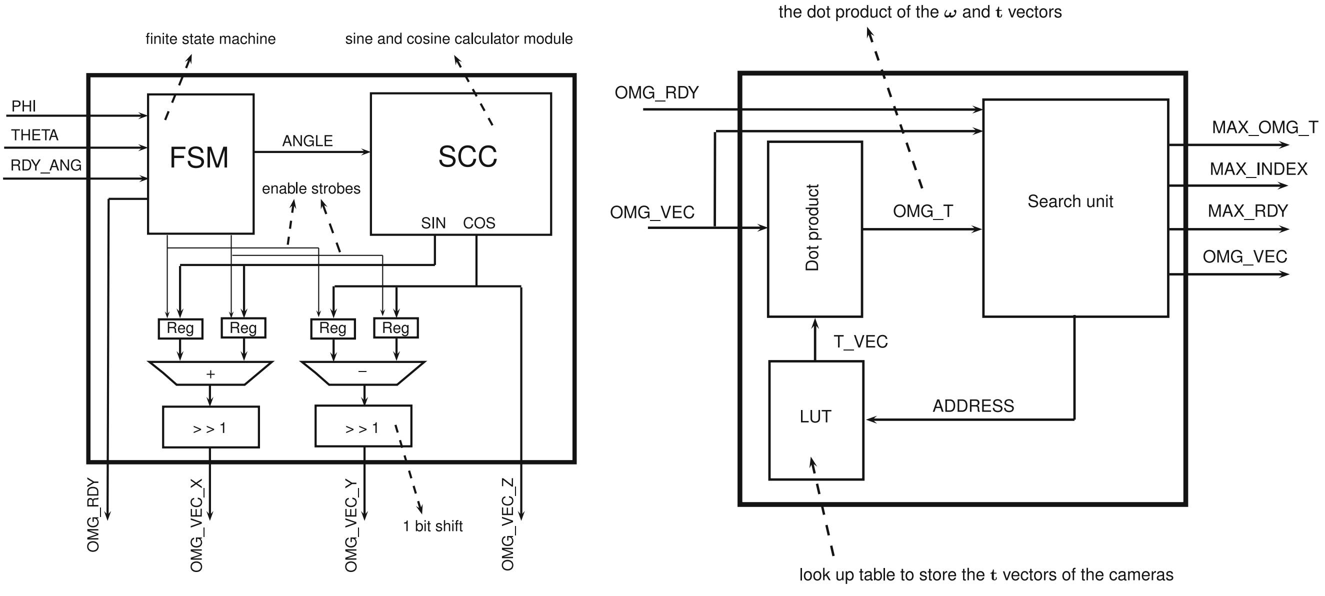
In order to reduce the implementation cost of this module, two combined approaches are proposed. The first approach replaces the multiplying trigonometric functions in $\mathbf{x}$ and $\mathbf{y}$ components of the $\boldsymbol{\omega}$ vector with their addition-based identities, that is,

$$\begin{aligned} \sin(a)\cos(b) &= \frac{1}{2}\big(\sin(a+b) \;+\; \sin(a-b)\big), \\ \sin(a)\sin(b) &= \frac{1}{2}\big(\cos(a-b) \;-\; \cos(a+b)\big). \end{aligned} \tag{23}$$

Hence, rewriting Eq. 20 yields:

$$\begin{aligned} \boldsymbol{\omega} = &\frac{1}{2}(\sin(\theta_\omega+\phi_\omega)+\sin(\theta_\omega-\phi_\omega))\,\mathbf{x} \\ &+\frac{1}{2}(\cos(\theta_\omega-\phi_\omega)-\cos(\theta_\omega+\phi_\omega))\,\mathbf{y}+\cos\theta_\omega\mathbf{z}. \end{aligned} \tag{24}$$

Thus, two multiplication operations in Eq. 20 are reduced to one addition and one subtraction operation in Eq. 23. The second approach applies resource sharing by using one SCC module and a finite state machine (i.e., FSM) to generate all the $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ components of the $\boldsymbol{\omega}$ vector. The SCC module calculates the sine and cosine of an angle at the same time. The following angle combinations are inputted to the SCC module: $(\theta_\omega+\phi_\omega)$, $(\theta_\omega-\phi_\omega)$ and θ_ω, then the respective sine and cosine outputs from the SCC modules are reordered and combined according to Eq. 24. The aforementioned concept is illustrated in Fig. 16a. The omega generation



(a) (b)

Figure 16 **a** Internal architecture of the omega vector generation module with one sine and cosine calculator module; **b** the nearest camera position finder module.

module passes out the calculated $\boldsymbol{\omega}$ vector denoted as OMEGA_VEC in Fig. 15 to the maximum search module. The advent of a new $\boldsymbol{\omega}$ vector is declared via the OMG_RDY strobe.

6.4.3 Maximum Search Module

The task of the maximum search module is to find the nearest camera position to the pixel direction $\boldsymbol{\omega}$. This task is accomplished by selecting the camera position for which the **t** vector's dot product with $\boldsymbol{\omega}$ is larger than $\cos(\frac{\alpha}{2})$ and maximum. The maximum search module contains three submodules as shown in Fig. 16b. The dot product submodule conducts the dot product operation of two vectors, each having three **x**, **y** and **z** components. The dot product operation is implementable with three signed multiplier and two adder units. The **t** vector component values of the cameras are stored in a look-up table (LUT). The search unit submodule provides all the address indexes of the LUT via the ADDRESS bus shown in Fig. 16b, to conduct the dot products of all the cameras **t** vectors with the newly presented $\boldsymbol{\omega}$ vector. At the same time, the maximum dot product value and the camera index which is the closest camera to the $\boldsymbol{\omega}$ is obtained. The search unit submodule hands over the maximum dot product value, the closest camera index and the $\boldsymbol{\omega}$ vector, which are denoted as MAX_OMG_T, MAX_INDEX and OMG_VEC in Fig. 16b, to the pixel position generation module. The transmission of a new output from this submodule is declared via the MAX_RDY strobe.

6.4.4 Pixel Position Generation Module

The pixel position generation module extracts the pixel position corresponding to the pixel direction $\boldsymbol{\omega}$ on the image frame of the camera which is closest to vector $\boldsymbol{\omega}$. The pixel position on the image frame of an observing camera in direction $\boldsymbol{\omega}$ is obtained by Eq. 8. Hence, the implementation of Eq. 8 is desired. Figure 17 depicts a block diagram for the implementation of the pixel position generation module. Both fraction terms in Eq. 8 contain a division of two vectors dot products, followed by multiplication by a constant (i.e., the focal length of the camera). The numerator of the fractions in Eq. 8 are calculated by two dot product submodules. The first input vector of the two dot product submodules are stored in two LUTs containing the **u** and **v** vectors of the cameras. The MAX_INDEX input coming from the maximum search module is used to address these LUTs. The second input of the dot product submodules is the $\boldsymbol{\omega}$ vector. The denominator of the fractions and the multiplication by constant in Eq. 8 are identical, which is taken advantage of by only computing a single division operation and multiplying its result with the calculated dot products of the numerator terms in Eq. 8.

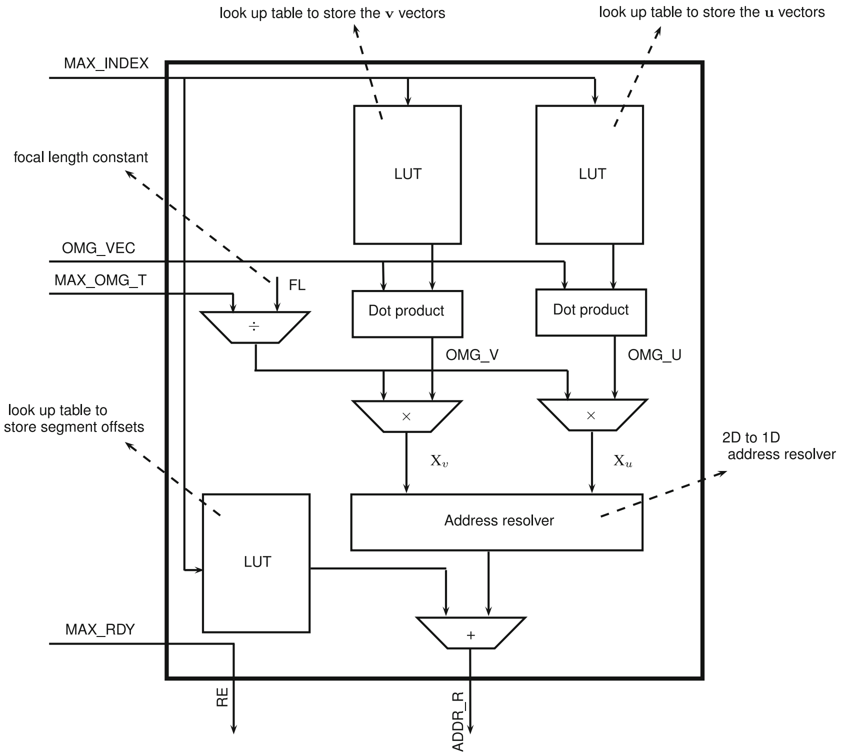

look up table to store the v vectors
look up table to store the u vectors
MAX_INDEX
LUT
LUT
focal length constant
OMG_VEC
MAX_OMG_T
FL
÷
Dot product
Dot product
OMG_V
OMG_U
look up table to store segment offsets
×
×
X_v
X_u
2D to 1D address resolver
Address resolver
LUT
MAX_RDY
+
RE
ADDR_R

Figure 17 Internal architecture of the pixel position generation module.

The latter concept is illustrated in Fig. 17. The generated two-dimensional position indexes X_u and X_v are transformed to a one-dimensional domain address presentation through an address resolver submodule. This transformation is required since the image data of the camera are stored in a single array SRAM memory. The output of the address resolver submodule is aggregated with the address offset of the camera segment closest to the pixel direction $\boldsymbol{\omega}$. The address offset of the camera segments maintained in the external SRAM memories are stored in a LUT. The addition in the last stage yields the address location of the pixel position of interest in the external SRAM memory. This address is passed to the SRAM memory controller for the retrieval of the target pixel position data.

6.4.5 Pixel Transfer Module

The pixel transfer module receives the incoming data from the SRAM memory controller which is addressed by the pixel position generation module. To accommodate the scalability of the system, this module preambles the pixel data with the maximum ω_t. This information is used by the central FPGA. The pixel transfer module hands over its data to the data link and control unit through the PDAT bus, as shown in Fig. 15. The data link and control unit block transfers the received data to the central FPGA via high speed LVDS serial links.

6.4.6 Scalability and Central FPGA

For each pixel direction $\boldsymbol{\omega}$, each concentrator FPGA board manipulates its own observation for the set of cameras that are interfaced to it. Alongside the single pixel intensity that each concentrator FPGA returns to the central FPGA, it also transfers the information of how close its nearest observing camera (if any) is to the pixel direction $\boldsymbol{\omega}$. This auxiliary information is the maximum ω_t among the set of interfaced cameras described in Section 6.4.3. The central FPGA hence receives an intensity value in addition to its corresponding ω_t, for each pixel direction $\boldsymbol{\omega}$ and from all the concentrator FPGAs. The central FPGA role consists of selecting one intensity value among the ones that are received from the concentrator FPGAs. This is accomplished by choosing the intensity value pertaining to the maximum ω_t for each pixel direction $\boldsymbol{\omega}$ originating from the concentrator FPGAs. Hence the role of the central FPGA is a selection based on a maximum criteria (i.e. closest observation).

Table 3 FPGA device utilization summary of the omnidirectional construction algorithm implementation.

Resources	Used	Available	Utilization
Occupied slices	1715	7200	24%
Slice registers	3226	28800	12%
BlockRAM/FIFO	2	48	4%
DSP48Es	13	48	27%

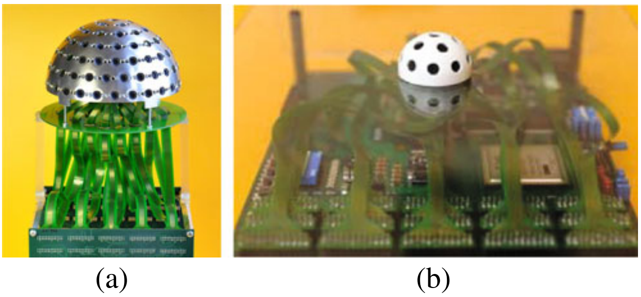


Figure 18 The built setup of FPGA platforms for Panoptic systems with, **a** 7-floor and 13 cm diameter hemisphere structure with thirty embedded cameras, **b** 3-floor and 3 cm diameter hemisphere structure with fifteen embedded cameras.

The FPGA device utilization summary of the algorithm presented in this Section is provided in Table 3. The omega vector generation module of the implemented design uses a LUT based SSC block.

A two-layer based FPGA platform for the 7-floor Panoptic camera is shown in Fig. 18a. The Panoptic camera shown in Fig. 18a includes thirty cameras. Two of the FPGA boards act as concentrators and one FPGA board functions as the central unit. The FPGA boards are connected via high speed LVDS links. Each line of the LVDS link has a 500 Mb/s transmission capacity. Figure 18b depicts a fifteen camera Panoptic system based on a single FPGA board. The FPGA board of this prototype acts both as the concentrator and the central unit. The hemisphere diameter of the fifteen camera prototype is $2\,r_{\odot} = 30$ mm.

7 Calibration

The reconstruction method described in Section 3, and the subsequent hardware realizations described in the next sections assume a perfect knowledge of intrinsic camera parameters such a AOV, lens distortion, focal length, intensity dynamics, as well as extrinsic camera parameters including camera localizations and orientations on the surface of the hemisphere.

The fabricated Panoptic camera shown in Fig. 3 intrinsically minimizes errors of extrinsic parameters as a benefit of the digital machining of a rigid aluminum structure. The use of identical CMOS cameras in all

Panoptic facets target the same goal. Nevertheless, a good estimation of the discrepancy between the theoretical and the actual camera intrinsic and extrinsic configuration is mandatory. This is the role of the calibrations methods described in the following.

7.1 Intrinsic Calibration

Intrinsic camera parameters are necessary to explain the vision of one camera. These characterize the mapping between a 3-D scene and the observed 2-D plane and are split into two classes. The first class is dedicated to the linear *homography*, that is, the 3×4 *camera* matrix mapping of 3-D points in (homogeneous) coordinates into 2-D (homogeneous) pixel coordinates [12]. The second class models the non-linear mapping effects such as the lens distortion. The reader is referred to [12] for more details of the theoretical estimation of these parameters.

For practical purposes, these parameters are extracted using the "Camera Calibration Toolbox for Matlab" from [26]. For each camera intrinsic calibration, this toolbox uses the vision of one flat checkerboard pattern of known size presented under different orientations.

7.2 Extrinsic Calibration

The toolbox [26] is also used to accurately determine the true extrinsic parameters of each camera c_i on the surface of the Panoptic sphere, which corresponds to the estimation of the camera center c_i and of the three vectors $(\mathbf{t}_i, \mathbf{u}_i, \mathbf{v}_i)$ described in Section 3.1.

The position of one camera relatively to one other is determined, provided that both observe the same checkerboard pattern and that their intrinsic parameters have previously been calibrated. The intrinsic calibration of the camera matrix provides knowledge of the mapping between the coordinate system of one camera, i.e., the one determined by the camera focal point (origin) c and the vectors set $\{\mathbf{t}, \mathbf{u}, \mathbf{v}\}$, and a coordinate system defined on the checkerboard plane. As depicted in Fig. 19, it is for instance possible to jump from the coordinate system of one camera F_1 to that of the checkerboard plane F_p, and similarly from F_2 to F_p. Therefore, using the inverse mapping from F_p to F_2, every point or vector expressed in the coordinate system of F_1 can be described in F_2. *A fortiori*, this representation in F_2 is therefore also available for the vectors $\{\mathbf{c}_1, \mathbf{t}_1, \mathbf{u}_1, \mathbf{v}_1\}$.

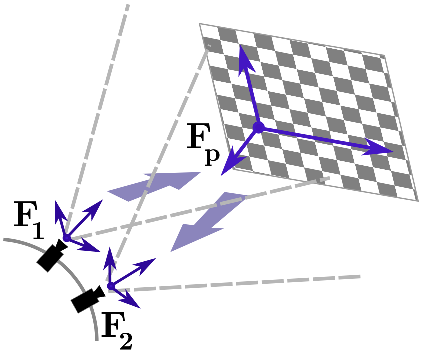


Figure 19 Extrinsic parameter calibration principles.

The full extrinsic calibration of the Panoptic device consists of, (i) arbitrarily considering one camera coordinate system as the fundamental system (e.g., the North pole camera c_0), and (ii) estimating the change of coordinate system between neighboring cameras thanks to the simultaneous observation of checkerboard planes between pair of neighboring cameras.

In the end of the process, working by overlapping camera neighborhood, the coordinates of the four vectors $\{\mathbf{c}_i, \mathbf{t}_i, \mathbf{u}_i, \mathbf{v}_i\}$ for each camera c_i are expressed in the common North pole frame $\{\mathbf{c}_0, \mathbf{t}_0, \mathbf{u}_0, \mathbf{v}_0\}$.

8 Results

The omnidirectional view construction examples achieved by the Panoptic device shown in Fig. 18a and whose internal parameters are given in Section 2.4 are presented in this section. A subset of 30 cameras among the 104 available physical locations are connected, to support mechanical constraints. All cameras shall be connected in future prototypes. This subset is selected to keep the camera arrangement in close agreement with the configuration reached with $N_{\mathrm{flo}} = 3$ (for which $N_{\mathrm{cam}} = 29$). According to Fig. 9b and considering $\alpha = 66°$, a full-view coverage distance close to $R_{\mathrm{FCD}} \simeq 3\,r_\odot$ is induced. The Panoptic device has been calibrated with the procedure described in Section 7. In all experiments presented in the following, the virtual omnidirectional views have a pixel resolution equal to $N_\phi \times N_\theta = 1024 \times 256$.

The first experiment has been realized from data recorded in Diagonal Street of the Swiss Federal Institute of Technology (EPFL). Images of all cameras have been integrated to provide an omnidirectional view in the hemisphere center ($\mathbf{q} = 0$). To this aim, the

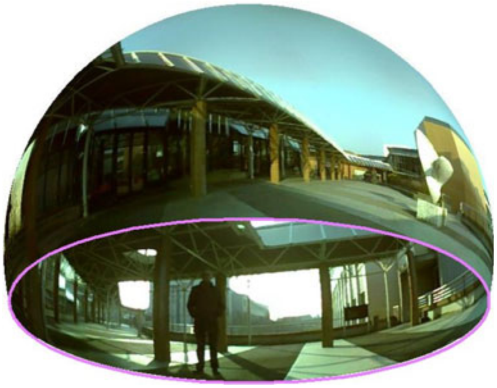


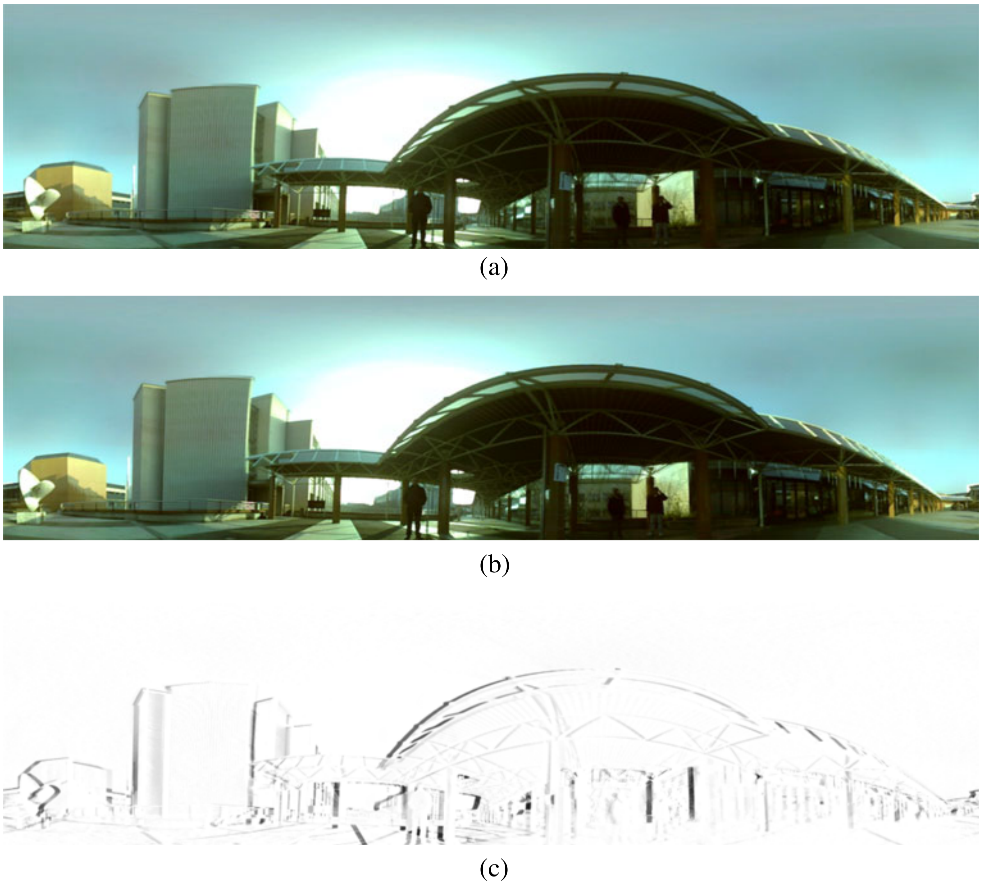
Figure 20 EPFL Diagonal street scene acquired by the camera system. Omnidirectional view represented on a sphere.

construction scheme described in Section 3 has been used. The result of the omnidirectional view mapped on the sphere of direction is depicted in Fig. 20. The scene is correctly constructed. Some intensity discontinuities occurring close to the Voronoi diagram boundaries are mainly due to the intensity dynamic of each camera which is not perfectly calibrated.

In the second experiment presented in Fig. 21, two "off-center" omnidirectional views are constructed in order to demonstrate the 3-D capability of the device. The two centers are $\mathbf{q} = (\frac{1}{2}r_{\odot}, 0, 0)$ (Fig. 21a) and $\mathbf{q} = (-\frac{1}{2}r_{\odot}, 0, 0)$ (Fig. 21b). The difference of the two images is presented in Fig. 21c which highlights the intensity discrepancies. Peripheral object motions caused by different point of views provide high intensity variations along object boundaries.

9 Conclusion and Future Work

This paper presents the theoretical analysis, the algorithmic derivations and the hardware implementation of a real-time omnidirectional camera with three-dimensional image acquisition capability. The systems is conceived in a modular manner, and the hardware resources can be adapted to carry out a wide range of image and video processing algorithms, thereby enabling the development of application-specific systems as well as consumer products.


(a)

(b)

(c)

Figure 21 EPFL Diagonal street off-center panoramic reconstruction. **a** Reconstruction using $\mathbf{q} = (\frac{1}{2}r_{\odot}, 0, 0)$. **b** Reconstruction using $\mathbf{q} = (-\frac{1}{2}r_{\odot}, 0, 0)$. **c** Image difference. For the three images, the horizontal and vertical directions correspond to φ and θ spherical coordinates respectively.

The major focus of this literature is the omnidirectional vision construction using the Panoptic camera. Future work related to the Panoptic device focuses on real-time depth map estimation, 3-D cinematography, and distributed parallel implementation of the Panoptic system in an interconnected network of cameras setup.

Acknowledgements The authors thank A. Cevrero, P. Athanasopoulos, S. Hauser, P. Bruehlmeier, D. Raboud, B. Devrim, A. Amini and the staff at the mechanical and electrical workshops of EPFL. The authors gratefully acknowledge the support of Xilinx, Inc., through the Xilinx University Program.

References

1. Richter, I. A. (1980). *The notebooks of Leonardo Da Vinci*. Oxford University Press.
2. Adelson, E. H., & Bergen, J. R. (1991). *Computational models of visual processing* (pp. 3–20). MIT Press.
3. Neumann, J., Fermuller, C., & Aloimonos, Y. (2002). In *3D data processing visualization and transmission, 2002. Proceedings. First international symposium on* (pp. 2–11).
4. Zbikowski, R. (2005). Spectrum. *IEEE, 42*(11), 46.
5. Geyer, C., & Daniilidis, K. (2001). *International Journal of Computer Vision, 45*(3), 223.
6. Adelson, E.H, & Wang, J.Y.A. (2002). Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 14*(2), 99–106.
7. Cianci, C., Raemy, X., Pugh, J., & Martinoli, A. (2006). In *Proceedings of the 2nd international conference on swarm robotics* (pp. 103–115). Springer-Verlag.
8. ViewPLUS. ASTRO Sensor Series (SOS:Stereo Omnidirectional System, BallCam). Available online at http://www.viewplus.co.jp/product/09/05.html. Accessed on 7 December 2011.
9. Lee, L. P., & Szema, R. (2005). Inspirations from biological optics for advanced photonic systems. *Science, 310*(5751), 1148–1150.
10. Jeong, K. H., Kim, J., & Lee, L. P. (2006). Biologically inspired artificial compound eyes. *Science, 312*(5773), 557–561.
11. Spherical trigonometry. Available online at http://en.wikipedia.org/wiki/Spherical_trigonometry. Accessed on 7 December 2011.
12. Hartley, R. I., & Zisserman, A. (2004). *Multiple view geometry in computer vision* (2nd Edn.). Cambridge University Press, ISBN: 0521540518.
13. Levoy, M., & Hanrahan, P. (1996). In *SIGGRAPH '96, proceedings of the 23rd annual conference on computer graphics and interactive techniques* (pp. 31–42). ACM.
14. Gortler, S., Grzeszczuk, R., Szeliski, R., & Cohen, M. (1996). In *SIGGRAPH '96, proceedings of the 23rd annual conference on computer graphics and interactive techniques* (pp. 43–54). ACM.
15. McMillan, L., & Bishop, G. (1995). In *SIGGRAPH '95, proceedings of the 22rd annual conference on computer graphics and interactive techniques* (pp. 39–46). ACM.
16. Górski, K. M., Hivon, E., Banday, A., Wandelt, B., Hansen, F., Reinecke, M., et al. (2004). HEALPix: A Framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal, 622*, 759–771.
17. Raboud, D. (2009). The panoptic camera—plenoptic interpolation in an omnidirectional polydioptric camera. Master's thesis, EPFL.
18. Aurenhammer, F. (1991). Computational geometry: Algorithms and applications. *ACM Computing Surveys, 23*, 345.
19. de Berg, M., Cheong, O., van Kreveld, M., & Overmars, M. (2008). *Computational geometry* (3rd Edn.). Springer-Verlag, ISBN: 978-3-540-77973-5.
20. Kilts, S. (2007). *Advanced FPGA design: Architecture, implementation, and optimization*. Wiley-IEEE Press.
21. Xilinx (2000). Application note 136. Available online at http://www.xilinx.com/support/documentation/application_notes/xapp136.pdf. Accessed on 7 December 2011.
22. Tierney, J., Rader, C., & Gold, B. (1971). A digital frequency synthesizer. *IEEE Transactions on Audio and Electroacoustics, 19*(1), 48–57.
23. Volder, J. E. (1959). The CORDIC trigonometric computing technique. *IRE Transactions on Electronic Computers, EC-8*(3), 330–334.
24. Walther, J. S. (1971). In *Proceedings of the 18–20 May 1971, Spring joint computer conference, AFIPS '71 (Spring)* (pp. 379–385). New York, NY, U.S.A.: ACM.
25. Meyer-Baese, U. (2007). *Digital signal processing with field programmable gate arrays* (3rd Edn.). Springer.
26. Bouguet, J. (2011). Camera calibration toolbox for matlab. Available online at http://www.vision.caltech.edu/bouguetj/calib_doc. Accessed on 7 December 2011.

Hossein Afshari received the BSc and MSc degree in electrical engineering majoring in communications and electronics respectively, from University of Tehran, Iran. He has joined the Microelectronics Systems Laboratory (LSM) at EPFL as a PhD student since February 2008. He has worked and collaborated in the implementation of several industrial projects such as DRM transmitter, base band DVBT transmitter, DVB-H ASIC receiver and MPEG-TS multiplexer. His research interests are implementation of digital transceivers, signal processing systems, image processing and vision reconstruction algorithms and embedded systems.

Laurent Jacques received the B.Sc. in Physics, the M.Sc. in Mathematical Physics and the PhD in Mathematical Physics from the Universite Catholique de Louvain (UCL), Belgium. He was a Postdoctoral Researcher with the Communications and Remote Sensing Laboratory of UCL in 2005.2006. He obtained in Oct. 2006 a fouryear $(3+1)$ Postdoctoral funding from the Belgian FRS-FNRS in the same lab. He was a visiting Postdoctoral Researcher, in spring 2007, at Rice University (DSP/ECE, Houston, TX, USA), and from Sep. 2007 to Jul. 2009, at the Swiss Federal Institute of Technology (LTS2/EPFL, Switzerland). Formerly funded by Belgian Science Policy (Return Grant, BELSPO), he is now a F.R.S.-FNRS Scientific Research Worker in the ICTEAM institute of UCL. His research focuses on Sparse Representations of signals (1-D, 2-D, and sphere), Compressed Sensing theory and application, Inverse Problems, and Computer Vision. Since 1999, Laurent Jacques has coauthored 16 papers in international journals, 30 conference proceedings and presentations in signal and image processing conferences, and three book chapters.

Luigi Bagnato received the M.S. degree in Informatics and Telecommunication Engineering from the University of Perugia in 2006. He worked as a collaborator in early 2007 in the Imaging and Vision laboratory, University of Milan-Bicocca, Italy. Since August 2007 he is working towards his Ph.D. with the Signal Processing Laboratory (LTS2 and LTS4) of the Swiss Federal Institute of Technology (EPFL) under the guidance of Prof. Pierre Vandergheynst and Prof. Pascal Frossard. His research interests include light field representation and processing, inverse problems and computer vision.

Alexandre Schmid received the M.Sc. degree in microengineering and the Ph.D. degree in electrical engineering from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1994 and 2000, respectively.

Since 1994, he has been with the EPFL, working with the Integrated Systems Laboratory as a Research and Teaching Assistant, and with the Electronics Laboratories as a Postdoctoral Fellow. In 2002, he was a Senior Research Associate with the Microelectronic Systems Laboratory, where he has been conducting research in the fields of bioelectronic interfaces, nonconventional signal processing and neuromorphic hardware, and reliability of nanoelectronic devices, and also teaches with the Microengineering and Electrical Engineering Departments of EPFL. Since 2011, he is a "Maître d'Enseignement et de Recherche" Faculty Member with EPFL. He is a coauthor and a coeditor of two books and over 90 articles published in journals and conferences.

Dr. Schmid has served as the General Chair of the Fourth International Conference on Nano-Networks in 2009 and has been serving as an Associate Editor of the Institute of Electrical, Information, and Communication Engineers Electronics Express since 2009.

Pierre Vandergheynst received the M.S. degree in physics and the Ph.D. degree in mathematical physics from the Université Catholique de Louvain, Louvain-la-Neuve, Belgium, in 1995 and 1998, respectively. From 1998 to 2001, he was a Postdoctoral Researcher with the Signal Processing Laboratory, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland. He was Assistant Professor at EPFL (2002–2007), where he is now an Associate Professor.

His research focuses on harmonic analysis, sparse approximations and mathematical data processing in general with applications covering signal, image and high dimensional data processing, sensor networks, computer vision.

He was co-Editor-in-Chief of Signal Processing (2002–2006) and is Associate Editor of the IEEE Transactions on Signal Processing (2007–present), the flagship journal of the signal processing community. He has been on the Technical Committee of various conferences, serves on the steering committee of the SPARS workshop and was co-General Chairman of the EUSIPCO 2008 conference.

Pierre Vandergheynst is the author or co- author of more than 60 journal papers, one monograph and several book chapters. He has received two IEEE best paper awards.

Professor Vandergheynst is a senior member of the IEEE, a laureate of the Apple 2007 ARTS award and of the 2009–2010 De Boelpaepe prize of the Royal Academy of Sciences of Belgium.

Yusuf Leblebici (M'90–SM'98–F'09) received his B.Sc. and M.Sc. degrees in electrical engineering from Istanbul Technical University, in 1984 and in 1986, respectively, and his Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign (UIUC) in 1990. Between 1991 and 2001, he worked as a faculty member at UIUC, at Istanbul Technical University, and at Worcester Polytechnic Institute (WPI). In 2000–2001, he also served as the Microelectronics Program Coordinator at Sabanci University.

Since 2002, Dr. Leblebici has been a Chair Professor at the Swiss Federal Institute of Technology in Lausanne (EPFL), and director of Microelectronic Systems Laboratory. His research interests include design of high-speed CMOS digital and mixed-signal integrated circuits, computer-aided design of VLSI systems, intelligent sensor interfaces, modeling and simulation of semiconductor devices, and VLSI reliability analysis.

He is the coauthor of 4 textbooks, namely, Hot-Carrier Reliability of MOS VLSI Circuits (Kluwer Academic Publishers, 1993), CMOS Digital Integrated Circuits: Analysis and Design (McGraw Hill, 1st Edition 1996, 2nd Edition 1998, 3rd Edition 2002), CMOS Multichannel Single-Chip Receivers for Multi-Gigabit Optical Data Communications (Springer, 2007) and Fundamentals of High Frequency CMOS Analog Integrated Circuits (Cambridge University Press, 2009), as well as more than 200 articles published in various journals and conferences.

He has served as an Associate Editor of IEEE Transactions on Circuits and Systems (II), and IEEE Transactions on Very Large Scale Integrated (VLSI) Systems. He has also served as the general cochair of the 2006 European Solid-State Circuits Conference, and the 2006 European Solid State Device Research Conference (ESSCIRC/ESSDERC). He is a Fellow of IEEE and has been elected as Distinguished Lecturer of the IEEE Circuits and Systems Society for 2010–2011.