Capturing Mosaic-Based Panoramic Depth Images with a Single Standard Camera

Peter Peer, Franc Solina
University of Ljubljana, Faculty of Computer and Information Science, Computer Vision Laboratory
Tržaška 25, 1000 Ljubljana, Slovenia
{*peter.peer, franc.solina*}*@fri.uni-lj.si*

Abstract. In this paper we present a panoramic depth imaging system. The system is mosaic-based which means that we use a single rotating camera and assemble the captured images in a mosaic. Due to a setoff of the camera's optical center from the rotational center of the system we are able to capture the motion parallax effect which enables the stereo reconstruction. The camera is rotating on a circular path with the step defined by an angle equivalent to one column of the captured image. The equation for depth estimation can be easily extracted from system geometry. To find the corresponding points on a stereo pair of panoramic images the epipolar geometry needs to be determined. It can be shown that the epipolar geometry is very simple if we are doing the reconstruction based on a symmetric pair of stereo panoramic images. We get a symmetric pair of stereo panoramic images when we take symmetric columns on the left and on the right side from the captured image center column. Epipolar lines of the symmetrical pair of panoramic images are image rows. We focused mainly on the system analysis. The system performs well in the reconstruction of small indoor spaces.

Key words: stereo vision, reconstruction, panoramic image, depth image, mosaicing, motion parallax effect.

1. Introduction

1.1. Motivation

Standard cameras have a limited field of view, which is usually smaller than the human field of view. Because of that people have always tried to generate images with a wider field of view, up to a full 360 degrees panorama, [12].

Under the term stereo reconstruction we understand the generation of depth images from two or more captured images. A depth image is an image that stores distances to points on the scene. The stereo reconstruction procedure is based on relations between points and lines on the scene and images of the scene. If we want to get a linear solution of the reconstruction procedure then the images can interact with the procedure in pairs, triplets or quadruplets, and relations are named accordingly to the number of images as epipolar constraint, trifocal constraint or quadrifocal constraint, [17]. We wish that the images would have the property that points and lines are visible in all images of the scene. This is the property of panoramic cameras and it presents our fundamental motivation.

In this paper we address only the issue how to enlarge the horizontal field of view of

images and we are not discussing how to enlarge the vertical field of view of images. In the future we also intend to enlarge the vertical field of view.

If we tried to build two panoramic images simultaneously by using two standard cameras which are mounted on two rotational robotic arms, we would have problems with the non-static scenes. Clearly, one camera would capture the motion of the other camera. So we decided to use only one camera. Our final goal is to develop a system for automatic navigation of a mobile robot in a room.

1.2. Basics about the system

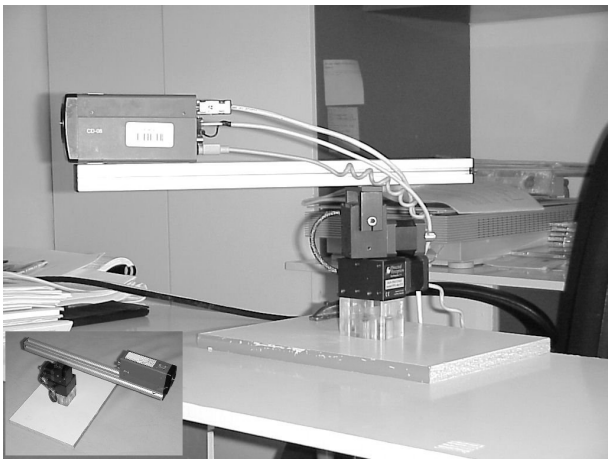


Fig. 1. Hardware part of our system.

In Fig. 1 the hardware part of our system can be seen: a color camera is mounted on a rotational robotic arm so that the optical center of the camera is offset from the vertical axis of rotation. The camera is looking outward from the system's rotational center. Panoramic images are generated by repeatedly shifting the rotational arm by the angle which corresponds to one column of the captured image. By assembling the center columns of these images, we get a mosaiced panoramic image. One of the properties of mosaic-based panoramic imaging is that the dynamic scenes are not well captured.

It can be shown that the epipolar geometry is very simple if we are doing the reconstruction based on a symmetric pair of stereo panoramic images. We get a symmetric pair of stereo panoramic images when we take symmetric columns on the left and on the right side from the captured image center column. These columns are assembled in a mosaiced stereo pair. The column from the left side of the captured image is mosaiced in the right eye panoramic image and the column from the right side of the captured

image is mosaiced in the left eye panoramic image.

1.3. Structure of the paper

In the next section we compare different panoramic cameras with emphasis on mosaicing. In Section 3 we give an overview of related work and expose the contribution of our work towards the discussed subject. Section 4 describes the geometry of our system, Section 5 is devoted to the epipolar geometry and Section 6 is about the procedure of stereo reconstruction. The focus of this paper is on the analysis of system capabilities, given in Section 7. In Section 8 we present some experimental results. At the very end of the paper we summarize the main conclusions and reveal some ideas for future work.

2. Panoramic cameras

Every panoramic camera belongs to one of three main groups of panoramic cameras: catadioptric cameras, dioptric cameras and cameras with moving parts. The basic property of a catadioptric camera is that it consists of a mirror (or mirrors ([14])) and a camera. The camera captures the image which is reflected from the mirror. A dioptric camera is using a special type of lens, e.g. fish-eye lens, which increases the size of the camera's field of view. A panoramic image can also be generated by moving the camera along some path and mosaicing together the images captured in different locations of the path.

Type of panoramic camera	Number of images	Resolution of panoramic images	Real time	References
catadioptric camera	1	low	yes	[14, 19, 22, 23]
dioptric camera	1	low	yes	[3, 5]
moving parts	a lot	high	no	[1, 6, 7, 8, 9, 10] [11, 12, 13, 15, 16, 18] [19, 20, 21, 24, 25]

Tab. 1. The comparison of different types of panoramic cameras regarding the number of standard images that are needed to build a panoramic image, the resolution of panoramic images and the capability of building a panoramic image in real time.

The comparison of different types of panoramic cameras is shown in Tab. 1.

All types of panoramic cameras enable 3D reconstruction. The camera has a single viewpoint or a projection center if all light rays forming the image intersect in a single point. Cameras with this property are also called the central cameras.

Mosaic-based procedures can be marked as non-central (we are not dealing with only one center of projection), they do not execute in real time and they give high resolution results. Thus the procedures are not appropriate for capturing dynamic scenes and consequently are less appropriate for reconstruction. The systems described in [1, 12] are exceptions because the light rays forming the mosaiced panoramic image are intersecting in the rotational center of the system. These two systems are central systems.

Dioptric panoramic cameras with wide angle lenses can be marked as non-central ([23]), they build a panoramic image in real time and they give low resolution results. Cameras with wide angle lenses are appropriate for fast capturing of panoramic images and processing of captured images, e.g. for detection of obstacles or for localization of a mobile robot, but are less appropriate for reconstruction. Please note that we are talking about panoramic cameras. Generally speaking dioptric cameras can be central.

Only some of catadioptric cameras have a single viewpoint. Cameras with a mirror (or mirrors) work in real time and they give low resolution results. Only two mirror shapes, namely hyperbolic and parabolic mirrors, can be used to construct a central catadioptric panoramic camera, [23]. Such panoramic cameras are appropriate for low resolution reconstruction of dynamic scenes and for motion estimation. It is also true that only for systems with hyperbolic and parabolic mirrors the epipolar geometry can be simply generalized, [17, 23].

3. Related work

We can generate panoramic images with the help of special panoramic cameras or with the help of a standard camera and with mosaicing standard images into panoramic images. If we want to generate mosaiced 360 degrees panoramic images we have to move the camera on a closed path, which is in most cases a circle.

One of the best known commercial packages for creating mosaiced panoramic images is QTVR (QuickTime Virtual Reality). It works on the principle of sewing together a number of standard images captured while rotating the camera, [6]. Peleg et al. ([21]) introduced the method for creation of mosaiced panoramic images from standard images captured with a handheld video camera. A similar method was suggested by Szeliski and Shum ([9]) which also does not strictly constraint the camera path but assumes that a great motion parallax effect is not present. All methods mentioned so far are used only for visualization purposes since the authors did not try to reconstruct the scene.

Ishiguro et al. ([1]) suggested a method which enables the reconstruction of the scene. They used a standard camera rotating on a circular path. The scene is reconstructed by means of mosaicing panoramic images together from the central column of the captured images and moving the system to another location where the task of mosaicing is repeated. Two created panoramic images are then used as input in a stereo reconstruction procedure. The depth of an object was first estimated using projections in two images

captured in different locations of the camera on the camera's path. But since their primary goal was to create a global map of the room, they preferred to move the system attached to the robot about the room. Clearly, by moving the robot to another location and producing the second panoramic image of a stereo pair in this location rather than producing a stereo pair in one location, they enlarged the disparity of the system. But this decision also has a few drawbacks: we can not estimate the depth for all points on the scene, the capturing time of a stereo pair is longer and we have to search for the corresponding points on the sinusoidal epipolar curves. The depth was then estimated from two panoramic images taken at two different locations of the robot in the room.

Peleg and Ben-Ezra ([15, 20]) introduced a method for creation of stereo panoramic images. Stereo panoramic images are created without actually computing the 3D structure — the depth effect is created in viewer's brain.

In [16] Shum and Szeliski described two methods used for creation of panoramic depth images, which are using standard procedures for stereo reconstruction. Both methods are based on moving the camera on a circular path. Panoramic images are built by taking one column out of a captured image and mosaicing the columns. They call such panoramic images *multiperspective panoramic images*. The crucial property of two or more multiperspective panoramic images is that they capture the information about the motion parallax effect, while the columns forming the panoramic images are captured from different perspectives. The authors are using such panoramic images as the input in a stereo reconstruction procedure.

However, multiperspective panoramic images are not something new to vision community ([16]): they are a special case of *multiperspective panoramic images for cel animation* ([10]), they are very similar to images generated with a procedure called *multiple-center-of-projection* ([13]), to *manifold projection* procedure ([21]) and to *circular projection* procedure ([15, 20]). The principle of constructing multiperspective panoramic images is also very similar to *the linear pushbroom camera* principle for creating panoramic images, [8].

In papers closest to our work ([1, 16]) we missed two things: an analysis of system capabilities and searching for corresponding points using the standard correlation technique and the epipolar constraint. Therefore the focus of this paper is on these two issues. While in [1] authors searched for corresponding points by tracking the feature from the column building the first panorama to the column building the second panorama, the authors in [16] used an upgraded *plane sweep stereo* procedure.

4. System geometry

Let us begin this section with description of how the stereo panoramic pair is generated. From the captured images on the camera's circular path we always take only two columns which are equally distant from the middle column. We assume that the middle column

that we are referring to in this paper, is the middle column of the captured image. The column on the right side of the captured image is then mosaiced in the left eye panoramic image and the column on the left side of the captured image is mosaiced in the right eye panoramic image. So, we are building each panoramic image from only one column of the captured image. Thus, we get a symmetric pair of panoramic images.

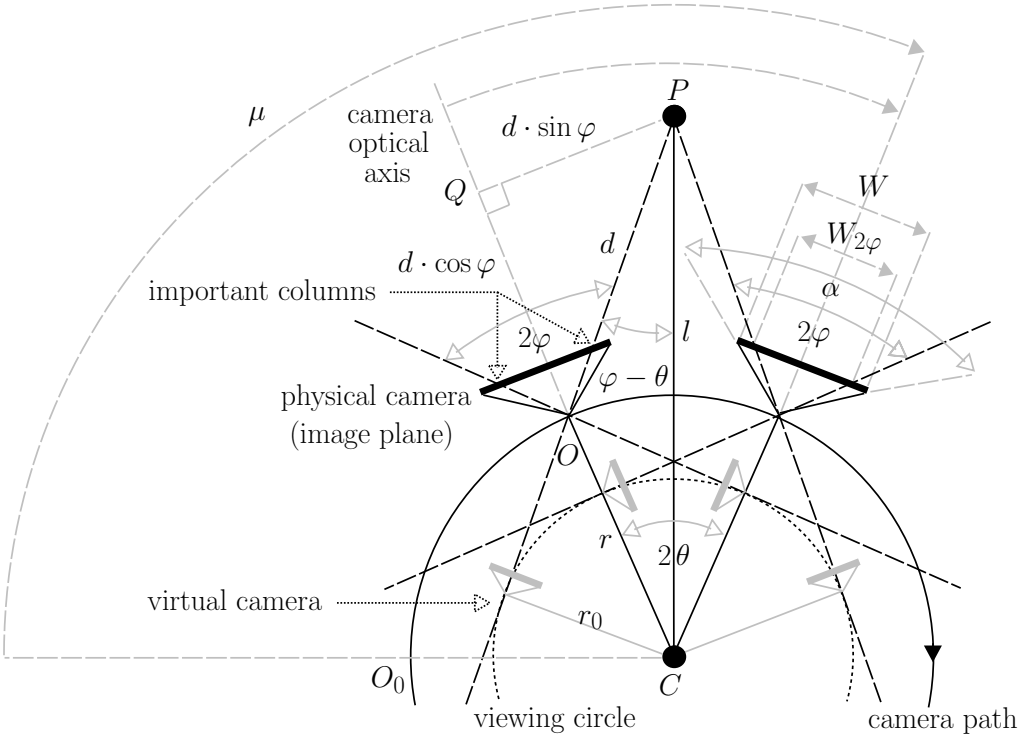


Fig. 2. Geometry of our system for constructing multiperspective panoramic images. Note that a ground-plan is presented. The optical axis of the camera is kept horizontal.

The geometry of our system for creating multiperspective panoramic images is shown in Fig. 2. Panoramic images are then used as an input to create panoramic depth images. Point C denotes the system's rotational center around which the camera is rotated. The offset of the camera's optical center from the rotational center C is denoted as r describing the radius of the circular path of the camera. The camera is looking outward from the rotational center. The optical center of the camera is marked with O. The column of pixels that is sewn in the panoramic image contains the projection of point P on the scene. The distance from point P to point C is the depth l and the distance from point

P to point O is denoted with d. θ is the angle between the line defined by point C and point O and the line defined by point C and point P. In the panoramic image the horizontal axis presents the path of the camera. The axis is spanned by μ and defined by point C, a starting point O_0 where we start capturing the panoramic image and the current point O. With φ we denote the angle between the line defined by point O and the middle column of pixels of the image captured by the physical camera looking outward from the rotational center (this column contains the projection of the point Q), and the line defined by point O and the column of pixels that will be mosaiced in panoramic image (this column contains the projection of the point P). Angle φ can be thought of as a reduction of the camera's horizontal view angle α.

The geometry of capturing multiperspective panoramic images can be described with a pair of parameters (r, φ).

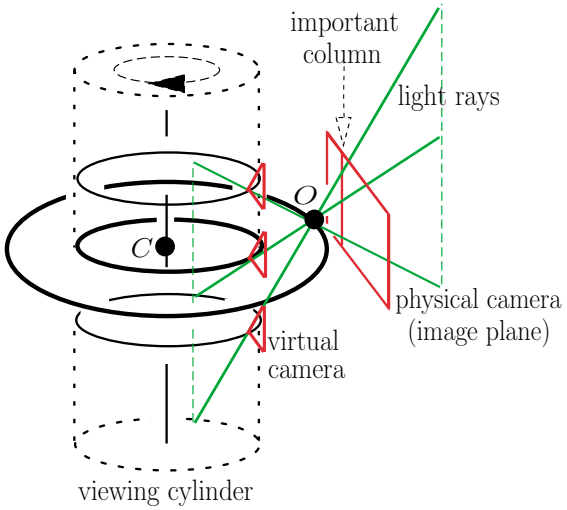
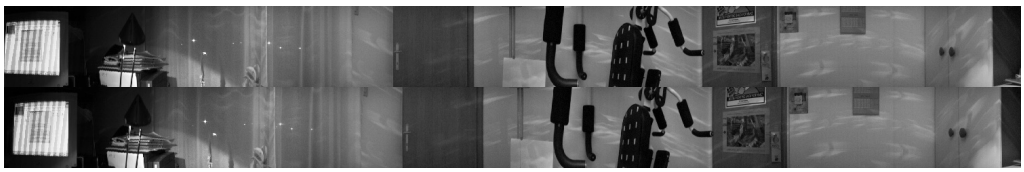


Fig. 3. All light rays forming the panoramic image are tangent to the viewing cylinder.

The system in Fig. 2 is obviously a non-central since the light rays forming the panoramic image are not intersecting in one point called the viewpoint, but instead are tangent ($\varphi \neq 0$) to a cylinder with radius r_0 called the viewing cylinder (Fig. 3). Thus, we are dealing with panoramic images formed by a projection from a number of viewpoints. This means that a captured point on the scene will be seen in the panoramic image only from one viewpoint. This is why the panoramic images captured in this way are called the multiperspective panoramic images.

$2\varphi = 29.9625^{\mathrm{o}}$



$2\varphi = 3.6125^{\mathrm{o}}$

Fig. 4. Two symmetric pairs of panoramic images which were generated with the different values of the angle φ. In Section 7.1 we will explain where did these values for the angle φ come from. Each symmetric pair of panoramic images comprises the motion parallax effect. This fact enables the stereo reconstruction.

For stereo reconstruction we need two images. If we are looking at only one circle on the viewing cylinder (Fig. 2) then we can conclude that our system is equivalent to a system with two cameras. In our case two virtual cameras are rotating on a circular path, i.e. viewing circle, with radius r_0. The optical axis of a virtual camera is always tangent to the viewing circle. The panoramic image is generated from only one pixel from the middle column of each image captured by a virtual camera. This pixel is determined by the light ray which describes the projection of the scene point on the physical camera image plane. If we observe a point on the scene P, we see that both virtual cameras which see this point, form a traditional stereo system of converging cameras.

Obviously, a symmetric pair of panoramic images used in stereo reconstruction process could be captured also with a bunch of cameras rotating on a circular path with radius r_0 where the optical axis of each camera is tangent to the circular path (Fig. 3).

Two images differing in the angle of rotation of the physical camera setup (for an example two image planes marked in Fig. 2) are used to simulate a bunch of virtual cameras on the viewing cylinder. Each column of the panoramic image is obtained from different position of the physical camera on a circular path. In Fig. 4 we present two symmetric pairs of panoramic images.

To automatically register captured images directly from knowing the camera's viewing direction, the camera lens' horizontal view angle α and vertical view angle β are required. If we know this information, we can calculate the resolution of one angular degree, i.e. we can calculate how many columns and rows are within an angle of one

degree. The horizontal view angle is especially important in our case, while we are moving the rotational arm only around it's vertical axis. To calculate these two parameters, we use an algorithm described in [12]. It is designed to work with cameras where zoom settings and other internal camera parameters are unknown. The algorithm is based on the mechanical accuracy of the rotational arm. The basic step of our rotational arm corresponds to an angle of $0.0514285\overline{5}\,^{\circ}$. In general this means that if we tried to turn the rotational arm for 360 degrees, we performed 7000 steps. Unfortunately, the rotational arm can not turn for 360 degrees around it's vertical axis. The basic idea of the algorithm is to calculate the translation dx (in pixels) between two images while the camera is rotated for a known angle $d\gamma$ in horizontal direction. Since we know the exact angle by which we move the camera, we can calculate the horizontal view angle of the camera:

$$\alpha = \frac{W}{dx} \cdot d\gamma, \tag{1}$$

where W is the width of the captured image in pixels. Now, we can calculate the resolution of one angular degree x_0:

$$x_0 = \frac{W}{\alpha}.$$

This equation enables us to calculate the width of the stripe W_s that will be mosaiced in the panoramic image when the rotational arm moves for an angle θ_0:

$$W_s = x_0 \cdot \theta_0.$$

From the above equation we can also calculate the angle of the rotational arm for which we have to move the arm if the stripe is only one column wide.

We were using the camera with horizontal view angle $\alpha = 34^{\circ}$ and vertical view angle $\beta = 25^{\circ}$. In the process of the construction of panoramic images we did not vary these two parameters.

5. Epipolar geometry

Searching for the corresponding points in two images is a difficult problem. Generally the corresponding point can be anywhere on the second image. That is why we would like to constraint the search space as much as possible. With the epipolar constraint we reduce the search space from 2D to 1D, i.e. to an epipolar line, [4]. In Section 7.2 we prove that in our system we can effectively reduce the search space even on the epipolar line.

In this section we will only illustrate the procedure of the proof that epipolar lines of the symmetrical pair of panoramic images are image rows, [16, 18, 24]. This statement is true for our system geometry. For proof see [18, 24].

The proof is based on radius r_0 of the viewing cylinder (Figs. 2 and 3). We can express r_0 in terms of known quantities r and φ as:

$$r_0 = r \cdot \sin\varphi \, .$$

We carry out the proof in three steps: *first*, we have to execute the projection equation for the line camera, *then* we have to write the projection equation for multiperspective panoramic image and in the *final* step we prove the property of epipolar lines for the case of a symmetrical pair of panoramic images. In the first step we are interested in how the point on the scene is projected to the camera's image plane ([4]) which has in our case, while we are dealing with a line camera, a dimension of $n \times 1$ pixels. In the second step, we have to write the relation between different notations of a point on the scene and the projection of this point on the panoramic image: notation of the scene point in Euclidean coordinates of the world coordinate system and in cylindrical coordinates of the world coordinate system, notation of the projected point in angular coordinates of the (2D) panoramic image coordinate system and in pixel coordinates of the (2D) panoramic image coordinate system. When we know the relations between the mentioned coordinate systems we can write the equation for projection of scene points on the cylindrical image plane of the panorama. Based on angular coordinates of the panoramic image coordinate system property, we can in the third step show that the epipolar lines of the symmetrical pair of panoramic images are actually rows of panoramic images. The basic idea for the last step of the proof is as follows:

> If we are given an image point in one panoramic image, we can express the optical ray defined by a given point and the optical center of the camera in 3D world coordinate system. If we project this optical ray described in world coordinate system on the second panoramic image, we get an epipolar line corresponding to the given image point in the first panoramic image.

6. Stereo reconstruction

Let us go back to Fig. 2. Using trigonometric relations evident from the sketch we can write the equation for depth estimation l of point P on the scene. Using the basic law of sines for triangles, we have:

$$\frac{r}{\sin(\varphi - \theta)} = \frac{d}{\sin\theta} = \frac{l}{\sin(180^{\mathrm{o}} - \varphi)},$$

and from this equation we can express the equation for depth estimation l as:

$$l = \frac{r \cdot \sin(180^{\mathrm{o}} - \varphi)}{\sin(\varphi - \theta)} = \frac{r \cdot \sin\varphi}{\sin(\varphi - \theta)}. \tag{2}$$

From Eq. (2) it follows that we can estimate depth l only if we know three parameters:

r, φ and θ. r is given. Angle φ can be calculated with regard to camera's horizontal view angle α (Eq. (1)) as:

$$2\varphi = \frac{\alpha}{W} \cdot W_{2\varphi}, \tag{3}$$

where W is the width of the captured image in pixels and $W_{2\varphi}$ is the width of the captured image between columns forming the symmetrical pair of panoramic images, given also in pixels. To calculate the angle θ we have to find corresponding points on panoramic images. Our system works by moving the camera for the angle corresponding to one column of captured image. If we denote this angle with θ_0, we can write angle θ as:

$$\theta = dx \cdot \frac{\theta_0}{2}, \tag{4}$$

where dx is the absolute value of difference between corresponding points image coordinates on horizontal axis x of the panoramic images.

We are using a procedure called *normalized correlation* to search for corresponding points ([4]), because it is one of the most commonly used technique for searching the corresponding point. In [2] Paar and Pölzleitner described other interesting methods than just those based on correlation.

Procedure of the normalized correlation works on the principle of similarity of scene parts within two scene images. The basic idea of the procedure is to find the part of the scene in the second image which is most similar to the given part of the scene in the first image. The procedure is using a window within which the similarity is measured with help of the correlation technique.

To increase the confidence in estimated depth we are using a procedure called *back-correlation*, [4]. The main idea of this procedure is to first find a point $\mathbf{m}_2$ in the second image which corresponds to a point $\mathbf{m}_1$ given in the first image. Then we have to find the corresponding point for the point $\mathbf{m}_2$ in the first image. Let us mark this corresponding point with $\mathbf{m}_1'$. If the point $\mathbf{m}_1$ is equal to the point $\mathbf{m}_1'$ then we keep the estimated depth value. Otherwise, we do not keep the estimated depth value. This means that the point $\mathbf{m}_1$ for which the back-correlation was not successful has no depth estimation associated with it in the depth image.

With the back-correlation we are also solving the problem of occlusions.

7. Analysis of system capabilities

7.1. Time complexity of creating a panoramic image

The biggest disadvantage of our system is that it can not produce panoramic images in real time since we are creating them by rotating the camera for a very small angle.

Because of mechanical vibrations of the system, we also have to be sure to capture an image when the system is completely still. The time that the system needs to create a panoramic image is much too long to make it work in real time.

In one circle around the system's vertical axis our system constructs 11 panoramic images: 5 symmetric pairs and a panoramic image from the middle columns of the captured images. It captures 1501 images with resolution of 160×120 pixels, where radius is $r = 30$ cm and the shift angle is $\theta_0 = 0.2^o$. We can not capture 360/0.2=1800 images because of the limitation of the rotational arm. The rotational arm can not turn for 360 degrees around it's vertical axis.

The horizontal view angle of our camera was 34^o. The middle column of the captured image was in our case the 80th column. The distances between the columns building up symmetric pairs of panoramic images were 141, 125, 89, 53 and 17 columns. These numbers include two columns building up each pair. This means that the value of the angle 2φ (Eq. (3)) corresponds to 29.9625^o (141 columns), 26.5625^o (125 columns), 18.9125^o (89 columns), 11.2625^o (53 columns) and 3.6125^o (17 columns).

The acquisition process takes a bit more than 15 minutes on PC Intel PII/350 MHz to end. The steps of the acquisition process are as follows:

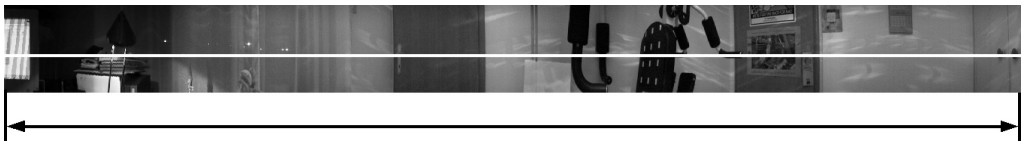
1. Move the rotational arm to it's initial position.
2. Capture the image.
3. Contribute image parts to the panoramic images.
4. Move the arm to the new position.
5. Check in the loop if the arm is already in the new position. The communication between the program and the arm is written in the file for debugging purposes. After the program exits the loop, it waits for 300 ms. This is done in order to stabilize the arm in the new position.
6. Repeat steps 2 to 5 until the last image is captured.
7. When the last image is captured, contribute image parts to the panoramic images and save them.

We could achieve faster execution since our code is not optimized. For example, we did not optimize the time of waiting (300 ms) after the arm is in the new position. No computing is done in parallel.

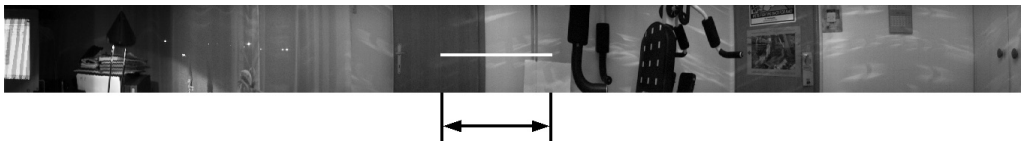
7.2. Constraining the search space on the epipolar line

Knowing that the width of the panoramic image is much bigger than the width of the captured image, we would have to search for a corresponding point along a very long epipolar line (Fig. 5a). Therefore we would like to constraint the search space on the epipolar line as much as possible. This means that the stereo reconstruction procedure executes faster. A side effect is also an increased confidence in the estimated depth.

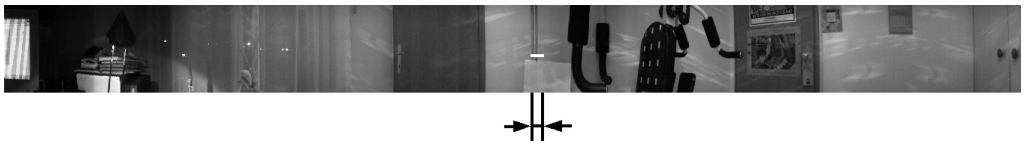
If we derive from Eq. (2) we can ascertain two things which nicely constraint the search space:



a) unconstrained length of the epipolar line: 1501 pixels



b) constrained length of the epipolar line: 149 pixels, $2\varphi = 29.9625^{\circ}$



c) constrained length of the epipolar line: 18 pixels, $2\varphi = 3.6125^{\circ}$

Fig. 5. We can effectively constraint the search space on the epipolar line.

1. Theoretically, the minimal possible estimation of depth is $l_{\min} = r$. This is true for $\theta = 0^{\circ}$. But practically this is impossible since the same point on the scene can not be seen in the column that will be mosaiced in the panorama for the left eye and at the same time in the column that will be mosaiced in the panorama for the right eye. If we observe horizontal axis of the panoramic image regarding the direction of the rotation, we can see that every point on the scene that is shown on both panoramic images (Fig. 4) is first imaged in the panorama for the left eye and then in the panorama for the right eye. Therefore we have to wait until the point imaged in the column building up the left eye panorama moves in time to the column building up the right eye panorama. If θ_0 presents the angle for which the camera is shifted, then $2\theta_{\min} = \theta_0$. This means that we have to make at least one basic shift of the camera to get a scene point projected in a right column of the captured image forming the left eye panorama, to be seen in the left column of the captured image forming the right eye panorama.

   Based on this fact, we can search for the corresponding point in the right eye panorama starting from the horizontal image coordinate $x + \frac{2\theta_{\min}}{\theta_0} = x + 1$ forward, where x is the horizontal image coordinate of the point on the left eye panorama for which we are searching the corresponding point. Thus, we get value $+1$ since the shift for angle θ_0 describes the shift of the camera for one column of the captured image.

   In our system the minimal possible depth estimation $l_{\min}$ depends on the value of the

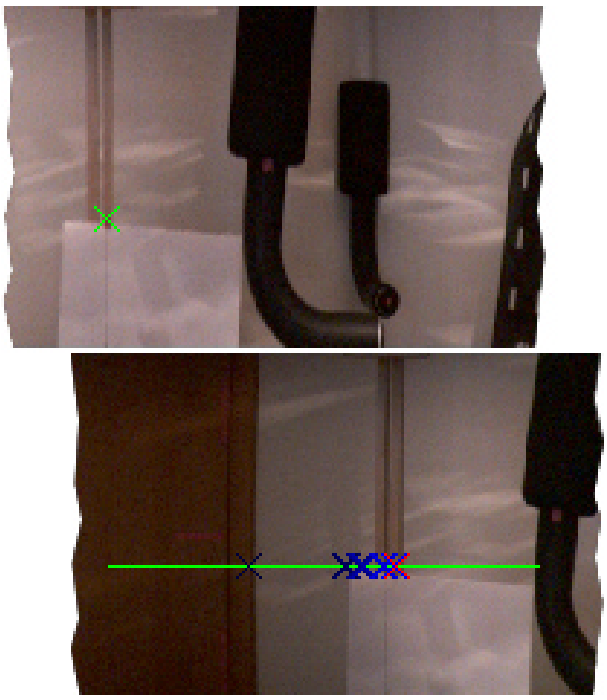


Fig. 6. Constraining the search space on the epipolar line in the case of $2\varphi = 29.9625^{\circ}$. On the left eye panorama (top image) we denoted the point for which we are searching the corresponding point with the green cross. On the right eye panorama (bottom image) we used green color to mark the part of the epipolar line on which the corresponding point must lie. The best corresponding point is marked with the red cross. With blue crosses we marked a few points which presented momentary the best corresponding point before we actually found the point with the maximal correlation.

angle φ:

$$\begin{aligned} l_{\min}(2\varphi = 29.9625^{\circ}) &= 302 \text{ mm} \\ &\ldots \\ l_{\min}(2\varphi = 3.6125^{\circ}) &= 318 \text{ mm}. \end{aligned}$$

2. Theoretically, the estimation of depth is not constrained upwards, but from Eq. (2) it is evident that the denominator must be non-zero. Practically, this means that for the maximal possible depth estimation $l_{\max}$ the difference $\varphi - \theta_{\max}$ must be equal to the value on the interval $(0, \frac{\theta_0}{2})$. We can write this fact as: $\theta_{\max} = n \cdot \frac{\theta_0}{2}$, where $n = \varphi \text{ div } \frac{\theta_0}{2}$ and $\varphi \bmod \frac{\theta_0}{2} \neq 0$.
   If we write the constraint for the last point, which can be a corresponding point on the

epipolar line, in analogy with the case of determining the starting point that can be a corresponding point on the epipolar line, we have to search for corresponding point in the right eye panorama to including horizontal image coordinate $x + \frac{2\theta_{\max}}{\theta_0} = x + n$. x is the horizontal image coordinate of the point on the left eye panorama for which we are searching the corresponding point.

Equivalently like in the case of minimal possible depth estimation $l_{\min}$, the maximal possible depth estimation $l_{\max}$ also depends upon the value of the angle φ:

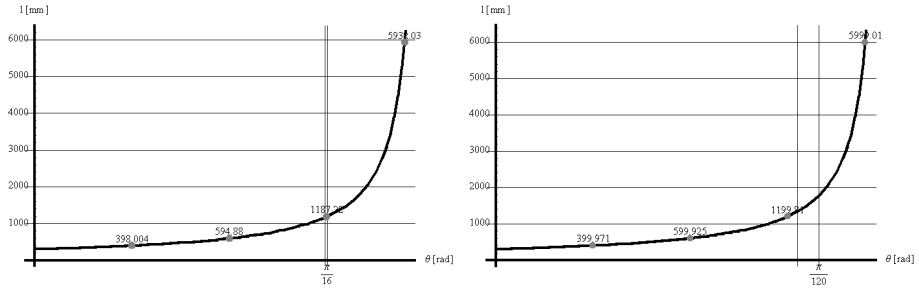
$$\begin{aligned} l_{\max}(2\varphi = 29.9625^{\circ}) &= 54687 \text{ mm} \\ &\dots \\ l_{\max}(2\varphi = 3.6125^{\circ}) &= 86686 \text{ mm}. \end{aligned}$$

In the following sections we will show that we can not trust the depth estimates near the last point of epipolar line search space, but we have proven that we can effectively constraint the search space.

To illustrate the use of specified constraints on real data, let us write the following example which describes the working process of our system: while the width of the panorama is 1501 pixels, we have to check only $n = 149$ pixels in case of $2\varphi = 29.9625^{\circ}$ (Figs. 5b and 6) and only $n = 18$ in case of $2\varphi = 3.6125^{\circ}$ (Fig. 5c), when searching for a corresponding point.

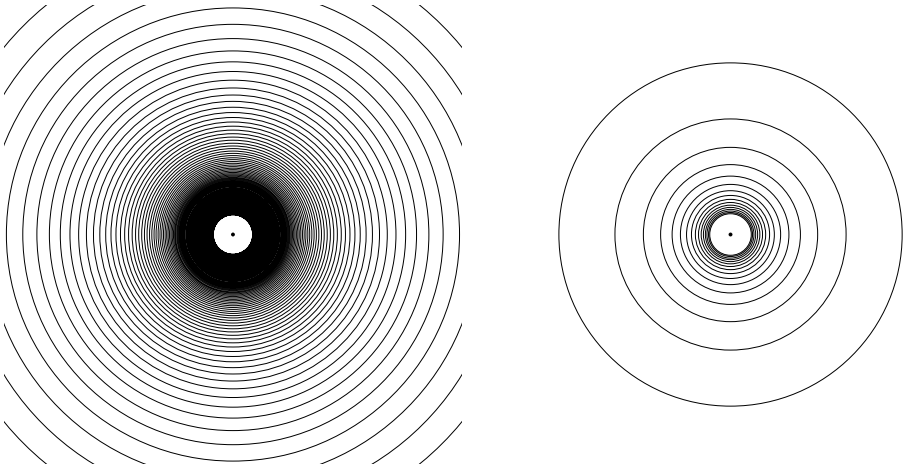
From the last paragraph we could conclude that the stereo reconstruction procedure is much faster for a smaller angle φ. But we will show in the next section that a smaller angle φ, unfortunately, has also a negative property.

7.3. Meaning of the one-pixel error in estimation of the angle θ



a) $2\varphi = 29.9625^{\circ}$ b) $2\varphi = 3.6125^{\circ}$

Fig. 7. Graphs showing dependence of depth function l from the angle θ while radius $r = 30$ cm and using different values of the angle φ. To ease the comparison of the one-pixel error in estimation of the angle θ we showed the interval of width $\frac{\theta_0}{2} = 0.1^{\circ}$ between the vertical lines around the third point.



a)
$2\varphi = 29.9625^{\circ}$
149 possible depth estimations

b)
$2\varphi = 3.6125^{\circ}$
18 possible depth estimations

Fig. 8. The number of possible depth estimation values is proportional to the angle φ. Each circle denotes possible depth estimation value.

Let us first define what we mean under the term one-pixel error. The images are discrete. Therefore, we would like to know what is the value of the error in the depth estimation if we miss the right corresponding point for only a pixel. And we would like to have this information for various values of the angle φ.

Before we illustrate the meaning of the one-pixel error in estimation of the angle θ, let us take a look at graphs in Fig. 7. Graphs are showing the dependence of depth function l from the angle θ while using different values of the angle φ. It is evident that the depth function l is rising slower in case of a bigger angle φ. This property decreases the error in depth estimation l when using bigger angle φ, but this decrease in the error becomes even more evident if we know that the horizontal axis is discrete and the intervals on the axis are $\frac{\theta_0}{2}$ degrees wide (see Fig. 7). If we compare the width of the interval on both graphs with respect to the width of interval that θ is defined on ($\theta \in [0, \varphi]$), we can see that the interval whose width is $\frac{\theta_0}{2}$ degrees, is much smaller when using bigger angle φ. This subsequently means that the one-pixel error in estimation of the angle θ is much smaller when using bigger angle φ, since a shift for the angle θ_0 describes the shift of the camera for one column of pixels.

Because of a discrete horizontal axis θ (Fig. 7), with intervals which are $\frac{\theta_0}{2}$ degrees wide (in our case $\theta_0 = 0.2^{\circ}$), the number of possible depth estimation values is proportional to the angle φ: we can calculate φ div $\frac{\theta_0}{2} = 149$ different depth values if we are

	$\theta - \frac{\theta_0}{2}$	θ	$\theta + \frac{\theta_0}{2}$
l [mm]	394.5	398	401.5
Δl [mm]	3.5		
(error)		3.5	

a) $\theta = \theta_1 = \frac{\varphi}{4}$, $2\varphi = 29.9625^{\circ}$

	$\theta - \frac{\theta_0}{2}$	θ	$\theta + \frac{\theta_0}{2}$
l [mm]	372.5	400	431.8
Δl [mm]	27.5		
(error)		31.8	

b) $\theta = \theta_1 = \frac{\varphi}{4}$, $2\varphi = 3.6125^{\circ}$

	$\theta - \frac{\theta_0}{2}$	θ	$\theta + \frac{\theta_0}{2}$
l [mm]	2252.9	2373.2	2507
Δl [mm]	120.3		
(error)		133.8	

c) $\theta = \theta_2 = \frac{7\varphi}{8}$, $2\varphi = 29.9625^{\circ}$

	$\theta - \frac{\theta_0}{2}$	θ	$\theta + \frac{\theta_0}{2}$
l [mm]	1663	2399.6	4307.4
Δl [mm]	736.6		
(error)		1907.8	

d) $\theta = \theta_2 = \frac{7\varphi}{8}$, $2\varphi = 3.6125^{\circ}$

Tab. 2. The one-pixel error in estimation of the angle θ, where $r = 30$ cm and $\theta_0 = 0.2^{\circ}$ (Eqs. (2) and (4)).

using angle $2\varphi = 29.9625^{\circ}$ (Fig. 8a) and only 18 different depth values if we are using the angle $2\varphi = 3.6125^{\circ}$ (Fig. 8b). This is the disadvantage of small angles φ.

Let us illustrate the meaning of the one-pixel error in estimation of angle θ: We would like to know what is the error of the angle θ if θ is at the beginning of the interval on which it is defined ($\theta \in [0, \varphi]$) and what is the error of the angle θ which is near the end of this interval?

For this purpose we will choose angles $\theta_1 = \frac{\varphi}{4}$ and $\theta_2 = \frac{7\varphi}{8}$. We are also interested in the nature of the error for different values of the angle φ. In this example we will use our already standard values for the angle φ: $2\varphi = 29.9625^{\circ}$ and $2\varphi = 3.6125^{\circ}$. Results in Tab. 2 give values of the one-pixel error in estimation of the angle θ for different values of parameters θ and φ.

From the results in Tab. 2 we can conclude that the error is much bigger in case of smaller angle φ than in case of bigger angle φ. The second conclusion is that the value of the error is getting bigger when the value of the angle θ is getting closer to the value of the angle φ. This is true regardless of the value of the angle φ. This two conclusions are also evident from Fig. 8: possible depth estimations lie on concentric circles centered in the center of the system and the distance between circles is increasing the further away they lie from the center. The figure nicely illustrates the fact that in case of a small angle φ, we can estimate only a few different depths and the fact that the one-pixel error in estimation of the angle θ increases if we move away from the center of the system.

We would like to get reliable depth estimates but at the same time we would like that the reconstruction procedure executes fast. Here we are faced with two contradicting requirements, since we have to make a compromise between the accuracy of the system and the speed of the reconstruction procedure. Namely, if we like to achieve the maximal

possible accuracy, then we would use the maximal possible angle φ. But this means that we would have to conduct a search for corresponding points on a larger segment of the epipolar line. Consequently, the speed of the reconstruction process would be slower. We would come to the same conclusion if we like to achieve a higher speed of the reconstruction procedure. The speed of the reconstruction process is inversely proportional to its accuracy.

By varying the parameters θ_0 and r we are changing the size of the error:

- By increasing the resolution of captured images we are decreasing the angle θ_0 and subsequently decreasing the rotational angle of the camera between two successively captured images forming the stereo panoramic images. For nearly the same factor as we increase (decrease) the resolution of captured images, we decrease (increase) the value of error Δl, while the reconstruction process takes more (less) time to end by nearly the same factor.
- By the same factor that we increase (decrease) radius r, we increase (decrease) the (biggest possible and sensible) depth estimation l and size of error Δl. If we vary the parameter r, the process of reconstruction will not be any faster or slower. In practice, bigger r means that we can reconstruct bigger scenes (rooms). The geometry of our system is adequate of reconstructing (smaller) rooms and is not suitable for reconstruction of an outdoor scene. This is due to the property of the system: we do not trust in the estimated depth l of far away objects on the scene were the size of the error Δl is too big.

7.4. Definition of the maximal reliable depth value

In Section 7.2 we defined the minimal possible depth estimation $l_{\min}$ and maximal possible depth estimation $l_{\max}$, but we did not write anything about the meaning of the one-pixel error in estimation of the angle θ for these two estimated depths. Let us examine the size of error Δl for these two estimated depths. We calculate $\Delta l_{\min}$ as an absolute value of difference between the depth $l_{\min}$ and the depth l for which the angle θ is bigger from the angle $\theta_{\min}$ by the angle $\frac{\theta_0}{2}$:

$$\Delta l_{\min} = |l_{\min}(\theta_{\min}) - l(\theta_{\min} + \frac{\theta_0}{2})| = |l_{\min}(\frac{\theta_0}{2}) - l(\theta_0)|.$$

Similarly, we calculate the error $\Delta l_{\max}$ as an absolute value of difference between the depth $l_{\max}$ and the depth l for which the angle θ is smaller from the angle $\theta_{\max}$ by the angle $\frac{\theta_0}{2}$:

$$\Delta l_{\max} = |l_{\max}(\theta_{\max}) - l(\theta_{\max} - \frac{\theta_0}{2})| = |l_{\max}(n\frac{\theta_0}{2}) - l((n-1)\frac{\theta_0}{2})|,$$

where variable n denotes a positive number in equation: $n = \varphi \text{ div } \frac{\theta_0}{2}$.

	$2\varphi = 29.9625^{\circ}$	$2\varphi = 3.6125^{\circ}$
$\Delta l_{\min}$	2 mm	19 mm
$\Delta l_{\max}$	30172 mm	81587 mm

Tab. 3. The one-pixel error (Δl) in estimation of the angle θ for the minimal possible depth estimation $l_{\min}$ and the maximal possible depth estimation $l_{\max}$ regarding the angle φ.

In Tab. 3 we gathered the error sizes for different values of angle φ. The results confirm statements in Section 7.3. We can add two additional conclusions:

1. The value of error $\Delta l_{\max}$ is unacceptably high and this is true regardless of the value of the angle φ. This is why we have to sensibly decrease the maximal possible depth estimation $l_{\max}$. In practice this leads us to define the upper boundary of allowed error size (Δl) for one pixel in estimation of the angle θ and with it, we subsequently define the maximal reliable depth value.
2. Angle φ always depends upon the horizontal view angle α of the camera (Eq. (3)). While the angle α is limited to around 40° for standard cameras, our system is limited with the angle α when estimating the depth, since in the best case we have: $\varphi_{\max} = \frac{\alpha}{2}$. Thus our system can really be used only for 3D reconstruction of small rooms.

8. Experimental results

Fig. 9 shows some results of our system. In the case denoted with b), we constructed the dense panoramic image, which means that we tried to find a corresponding point on the right eye panorama for every point on the left eye panorama. Black color marks the points on the scene with no depth estimation associated. Otherwise, the nearer the point on the scene is to the rotational center of the system, the lighter the point appears in the depth image.

In the case denoted with d), we used the information about the confidence in estimated depth (case c), which we get from the normalized correlation estimations. In this way, we eliminated from the dense depth image all the associated depth estimates which do not have a high enough associated confidence estimation. The lighter the point appears in case c), the more we trust in the estimation of the normalized correlation for this point.

In the case marked with e), we created a sparse depth image by searching only for the correspondences of feature points on input panoramic images. The feature points we used were vertical edges on the scene, which were derived by filtering the panoramic images with the Sobel filter for searching the vertical edges, [1, 4]. If we use a smaller value for angle φ, the reconstruction time would be up to eight times smaller from presented ones. All results were generated by using a correlation window of size $2n + 1 \times 2n + 1$,

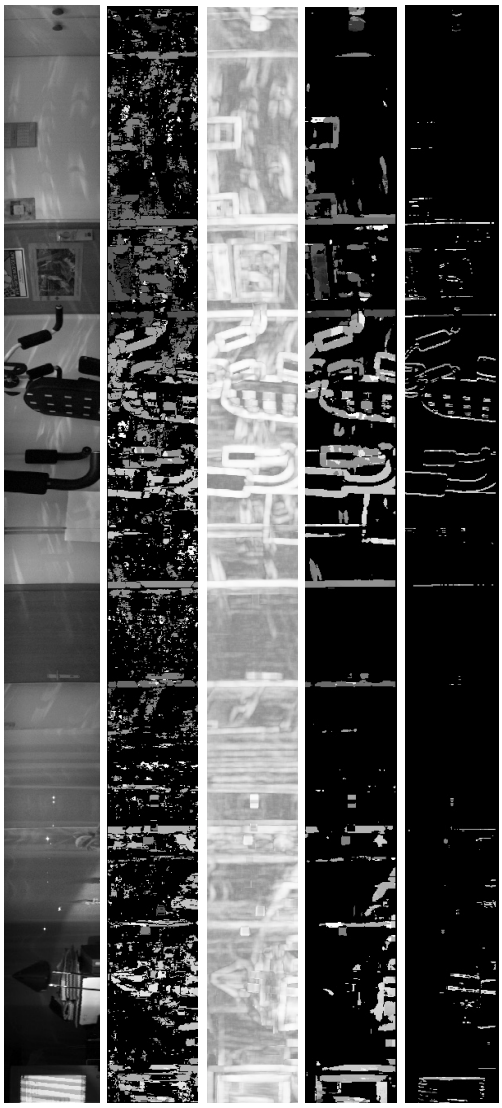


Fig. 9. Some results of stereo reconstruction when creating the depth image for the left eye while angle $2\varphi = 29.9625^{\circ}$: a) left eye panorama, b) dense depth image / using back-correlation / reconstruction time: 6 hours, 42 min., 20 sec., c) confidence of estimated depth, d) dense depth image after weighting / without back-correlation / reconstruction time: 3 hours, 21 min., 56 sec., e) sparse depth image / without back-correlation / reconstruction time: 38 seconds. The results were generated on PC Intel PII/350 MHz. The time needed for the acquisition of panoramic images is not included in the reconstruction time.

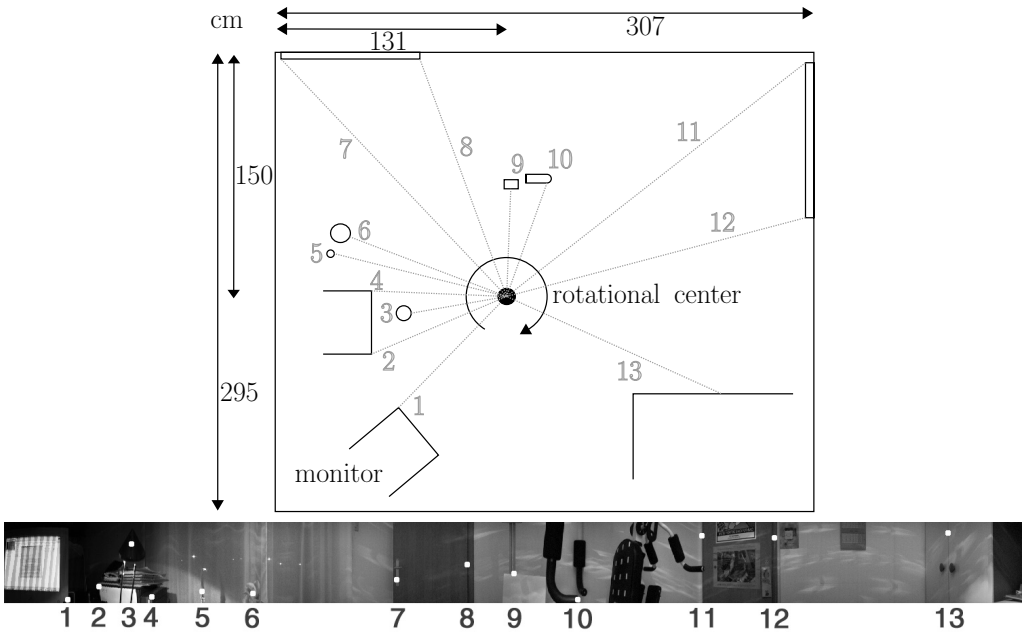


Fig. 10. On the top picture there is the plan of the room we were reconstructing. On the bottom picture we marked the features on the scene that will help us evaluate the quality of generated depth images.

n=4. We searched for corresponding points only on the panoramic image row which was determined by the epipolar geometry.

Since it is hard to evaluate the quality of generated depth images given in Fig. 9, we will present four reconstructions of the room from generated depth images. Then we will be able to evaluate the quality of generated depth images and consequently the quality of the system.

The plan of the room that we were reconstructing is given in Fig. 10. On the sketch are marked the features on the scene that will help us to evaluate the quality of generated depth images. The result of the (3D) reconstruction process is a ground-plan of the scene. The following properties are common to Figs. 11, 12, 13 and 14:

- Big dots denote features on the scene for which we measured the actual depth by hand.
- Big dot near the center of the reconstruction shows the center of our system.
- Small black dots are reconstructed points on the scene.
- Lines between black dots denote links between two successively reconstructed points.

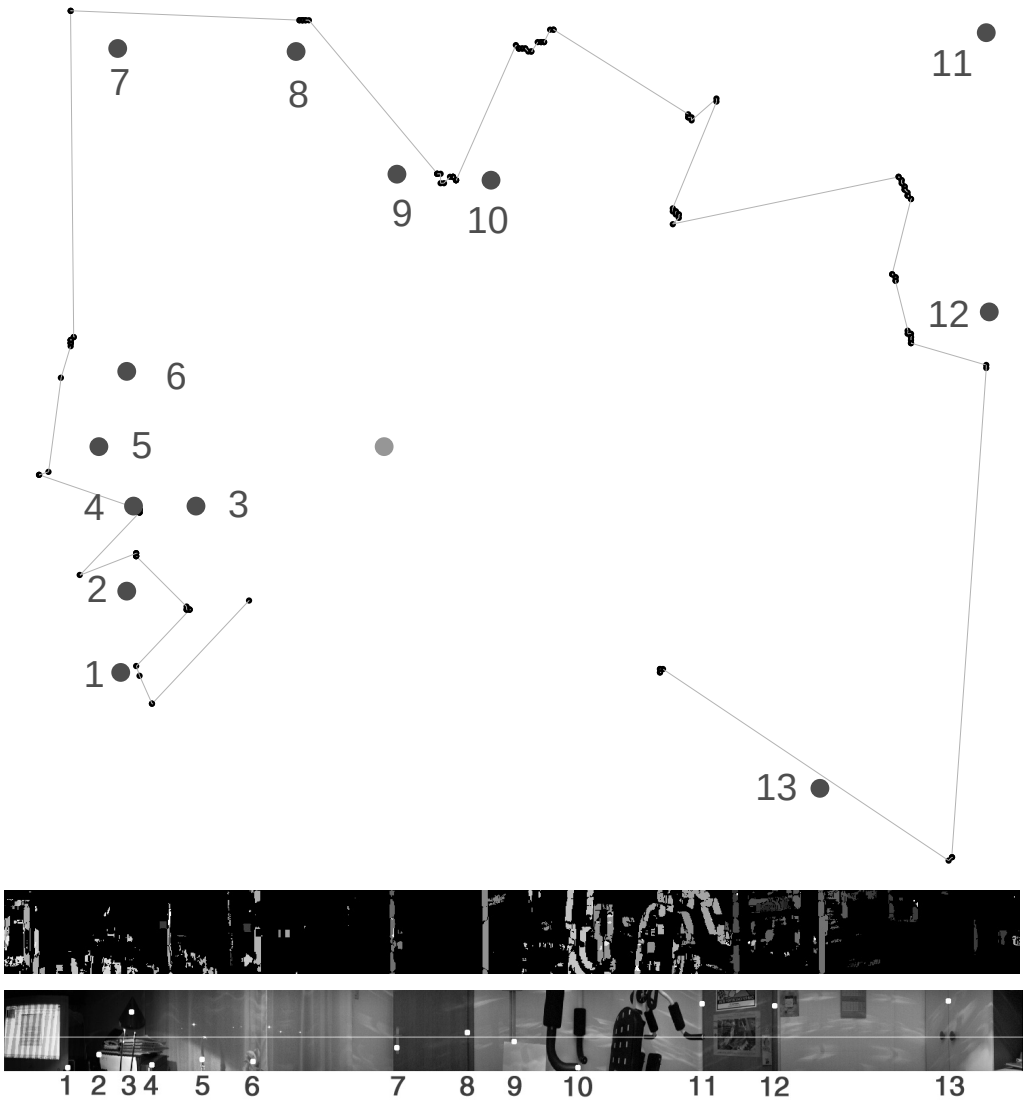


Fig. 11. On top is a ground-plan showing the results of the reconstruction process based on the 68th row of the depth image. We used back-correlation and weighting for angle $2\varphi = 29.9625^{\circ}$. The corresponding depth image is shown in the middle picture. For orientation, the reconstructed row and the features on the scene for which we measured the actual depth by hand are shown on the bottom picture. The features on the scene marked with big dots and associated numbers are not necessarily visible in this row.

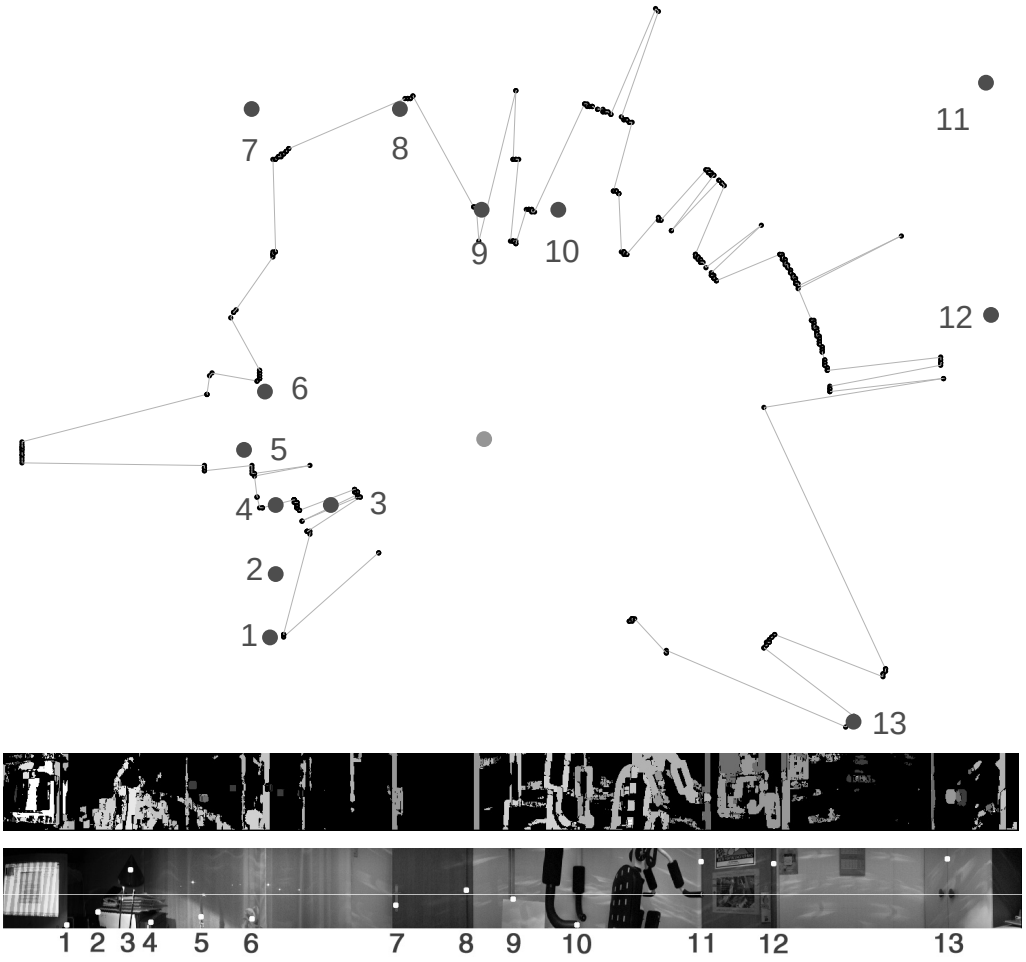


Fig. 12. On top is a ground-plan showing the results of the reconstruction process based on the 68th row of the depth image. We used back-correlation and weighting for angle $2\varphi = 3.6125^{\circ}$. The corresponding depth image is shown in the middle picture. For orientation, the reconstructed row and the features on the scene for which we measured the actual depth by hand are shown on the bottom picture. The features on the scene marked with big dots and associated numbers are not necessarily visible in this row.

The result of the reconstruction process based on the 68th row of the depth image when we used back-correlation and weighting is given in Fig. 11 for angle $2\varphi = 29.9625^{\circ}$ and in Fig. 12 for angle $2\varphi = 3.6125^{\circ}$. In Figs. 11 and 12 black dots are reconstructed on the basis of estimated depth values, which are stored in the same row of the depth image. The features on the scene marked with big dots are not necessarily visible in the same row.

In Fig. 12 we can observe two properties of the system: reconstructed points are on concentric circles centered in the center of the system and the distance between circles is increasing the further away they lie from the center. The figure nicely illustrates the fact that in case of a small angle φ, we can estimate only a few different depths and the fact that the one-pixel error in estimation of the angle θ increases if we move away from the center of the system.

We built sparse depth images by detecting first vertical edges in panoramic images. We made an assumption that points on vertical edges have the same depth which was true in the examples shown in this paper. The results of the reconstruction shown in Figs. 13 and 14 are based on information within the entire sparse depth image: at first we calculate the average depth within each column of the depth image and then we show this average depth value on the ground-plan of the scene. In Figs. 13 and 14 the results are derived from the sparse depth image using back-correlation, the result in Fig. 13 is given for angle $2\varphi = 29.9625^{\circ}$ and the result in Fig. 14 is given for angle $2\varphi = 3.6125^{\circ}$. We placed one additional constraint in the reconstruction process: each column in the depth image must contain at least four points with associated depth estimates or the average depth is not shown on the ground-plan of the scene.

Let us end this section with the demonstration of the reconstruction error. The error function on the manually measured points on the scene is evaluated in Tab. 4.

9. Summary and future work

We presented an exhaustive analysis of our mosaic-based system for construction of depth panoramic images using only one standard camera. We demonstrated the following: the procedure for creating panoramic images is very long and can not be executed in real time under any circumstances (using only one camera); epipolar lines of symmetrical pair of panoramic images are image rows; based on the equation for estimation of depth l (Eq. (2)), we can constraint the search space on the epipolar line; confidence in estimated depth is changing: the bigger the slope of the function l curve, the smaller the confidence in estimated depth; if we observe the reconstruction time, we can conclude that the creation of dense panoramic images is very expensive.

The essential conclusions are:

1. Such systems can be used for 3D reconstruction of small rooms.
2. With respect to the presented reconstruction times (Fig. 9) we could conclude that

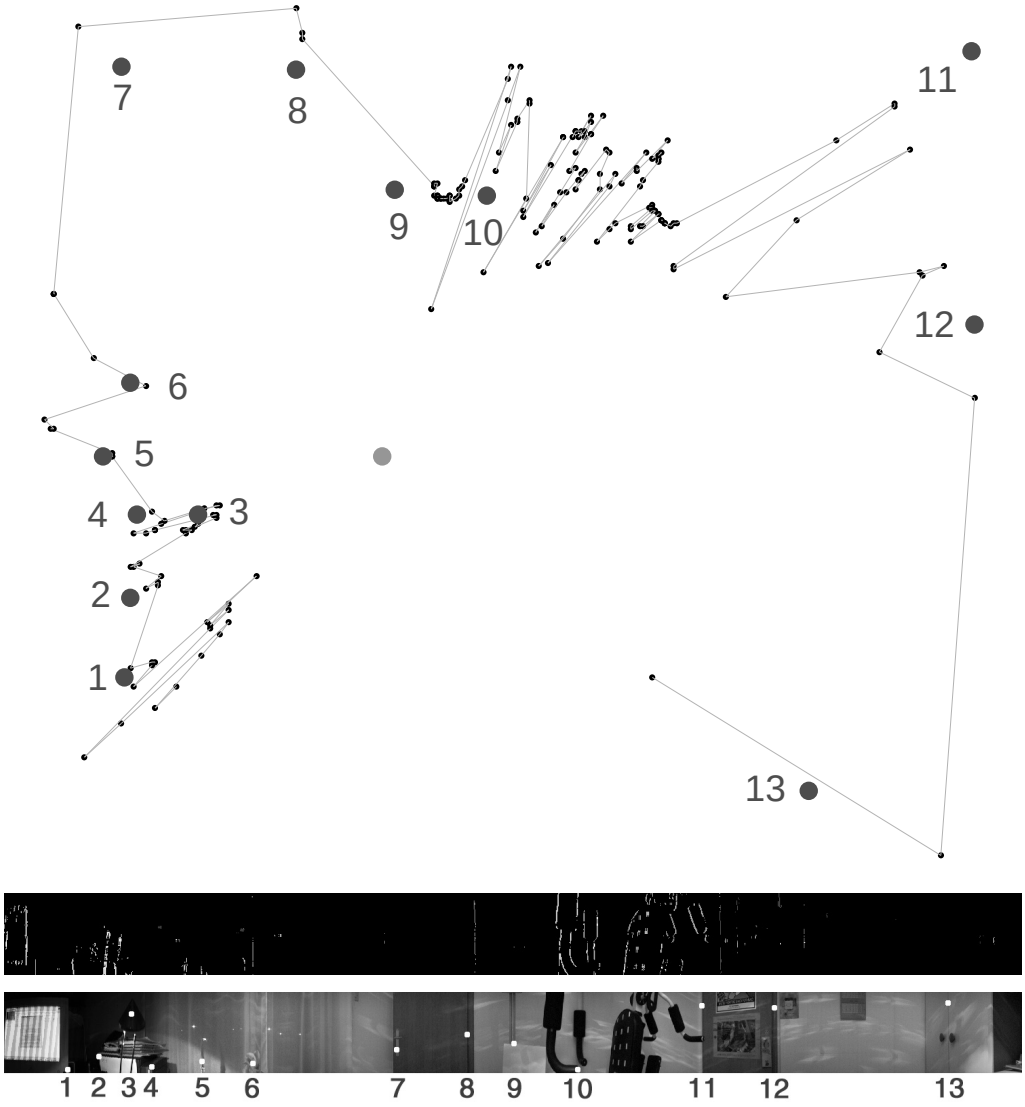


Fig. 13. On top is a ground-plan showing the results of the reconstruction process based on the average depth within each column of the sparse depth image. We used back-correlation for angle $2\varphi = 29.9625^{\circ}$. The corresponding sparse depth image is shown in the middle picture. For orientation, the features on the scene for which we measured the actual depth by hand are shown on the bottom picture.

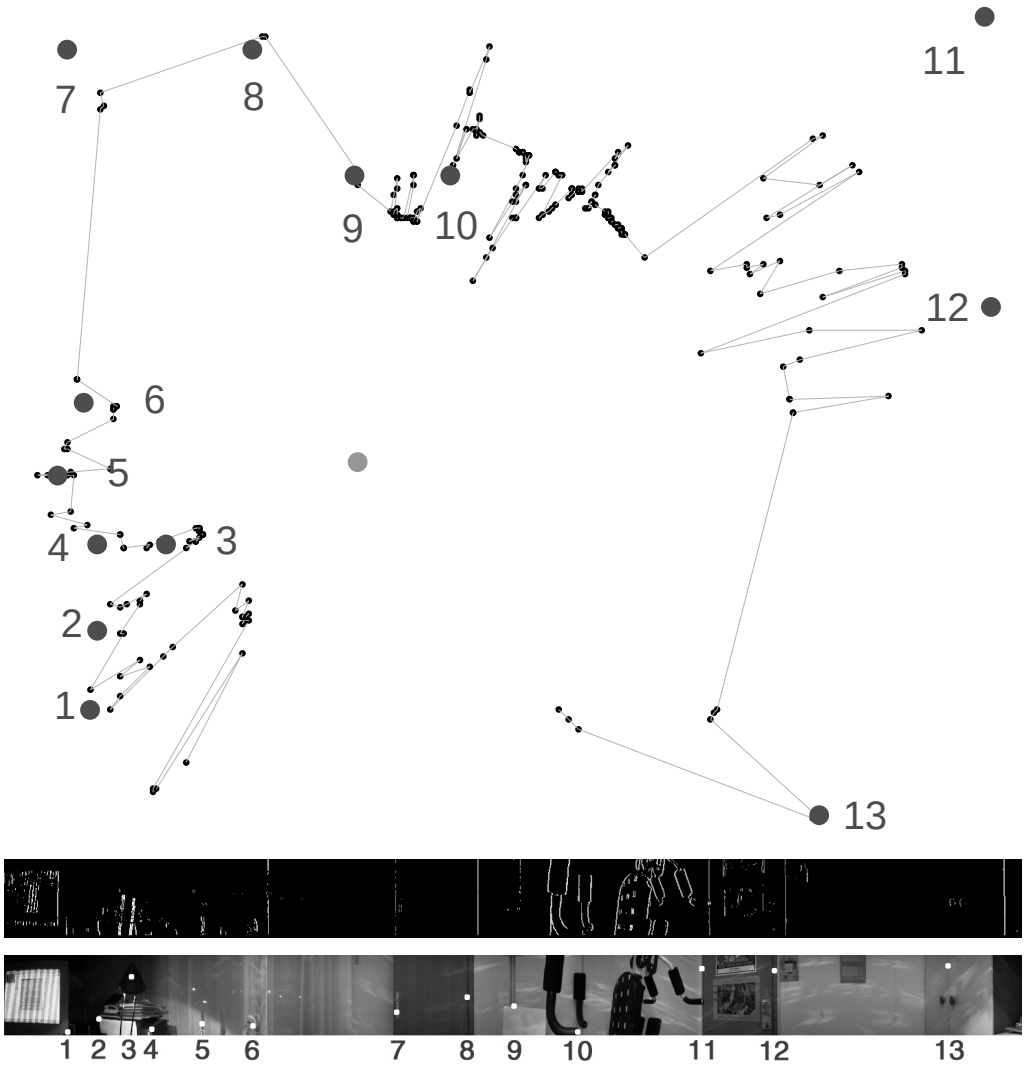


Fig. 14. On top is a ground-plan showing the results of the reconstruction process based on the average depth within each column of the sparse depth image. We used back-correlation for angle $2\varphi = 3.6125°$. The corresponding sparse depth image is shown in the middle picture. For orientation, the features on the scene for which we measured the actual depth by hand are shown on the bottom picture.

feature marked in Fig. 10 with	actual distance d [cm]	estimated depth l [cm] for $2\varphi =$		difference $l - d$ [cm (% of d)] for $2\varphi =$	
		3.6125°	29.9625°	3.6125°	29.9625°
1	111.5	89.4	109	-22.1 (-19.8%)	-2.5 (-2.2%)
2	95.5	76.7	89.3	-18.8 (-19.6%)	-6.2 (-6.5%)
3	64	53.8	59.6	-10.2 (-15.9%)	-4.4 (-6.9%)
4	83.5	76.7	78.3	-6.8 (-8.1%)	-5.2 (-6.2%)
5	92	89.4	89.3	-2.6 (-2.8%)	-2.7 (-2.9%)
6	86.5	76.7	82.7	-9.8 (-11.3%)	-3.8 (-4.4%)
7	153	133.4	159.8	-19.6 (-12.8%)	6.8 (4.4%)
8	130.5	133.4	135.5	2.9 (2.2%)	5 (3.8%)
9	88	76.7	87.6	-11.3(-12.8%)	-0.4 (-0.5%)
10	92	89.4	89.3	-2.6 (-2.8%)	-2.7 (-2.9%)
11	234.5	176.9	213.5	-57.6 (-24.6%)	-21 (-9%)
12	198	176.9	179.1	-21.1 (-10.7%)	-18.9 (-9.5%)
13	177	176.9	186.7	-0.1 (-0.1%)	9.7 (5.5%)

Tab. 4. The comparison between the manually measured actual distances and estimated depths (Eq. (2)) for some features on panoramic images. It is given for different values of the angle φ and $r = 30$ cm. Please remember that we can calculate only 149 different depth values if we are using the angle 29.9625° and only 18 different depth values if we are using the angle 3.6125° (Section 7.3).

the reconstruction procedure could work in nearly real time, if we work with 8-bit grayscale images, with lower resolution, if we create the sparse depth image of only part of the scene and/or simply if we use a faster computer. This could be used for robot navigation, [1].

A further time reduction in panorama building can be achieved: instead of building the panorama from only one column of the captured image, we could build the panorama from the wider stripe of the captured image, [12]. Thus, we would increase the speed of the building process. If we use this idea in our system, we know that within the stripe the angle φ is changing. However, the question is how this influences the reconstruction procedure.

In the future we intend to enlarge the vertical field of view of panoramic images, address the precision of vertical reconstruction and use the sub-pixel accuracy procedure.

Our future work is directed primarily in the development of an application for the real time automatic navigation of a mobile robot in a room.

Acknowledgment

This work was supported by the Ministry of Education, Science and Sport of the Republic of Slovenia (project Computer Vision 1539-506).

References

1992

[1] Ishiguro H., Yamamoto M., Tsuji S.: Omni-directional stereo. IEEE Trans. PAMI, 14(2), 257–262.

[2] Paar G., Pölzleitner W.: Robust disparity estimation in terrain modeling for spacecraft navigation. Proc. IEEE ICPR, The Hague, The Netherlands, August 30 – September 3, I:738–741.

[3] Weng J., Cohen P., Herniou M.: Camera calibration with distortion models and accuracy evaluation. IEEE Trans. PAMI, 14(10), 965–980.

1993

[4] Faugeras O.: Three-Dimensional Computer Vision: A Geometric Viewpoint. MIT Press, Cambridge, Massachusetts, London, England.

1995

[5] Basu A., Licardie S.: Alternative models for fish-eye lenses. Pattern Recognition Letters, 16(4), 433–441.

[6] Chen S.: Quicktime VR — an image-based approach to virtual environment navigation. Proc. ACM SIGGRAPH, Los Angeles, USA, August 6–11, 29–38.

1996

[7] Szeliski R.: Video Mosaics for Virtual Environments, IEEE Computer Graphics and Applications, 16(2), 22–30.

1997

[8] Gupta R., Hartley R. I.: Linear pushbroom cameras. IEEE Trans. PAMI, 19(9), 963–975.

[9] Szeliski R., Shum H. Y.: Creating full view panoramic image mosaics and texture-mapped models. Computer Graphics (ACM SIGGRAPH), Los Angeles, USA, August 3–8, 251–258.

[10] Wood D., Finkelstein A., Hughes J., Thayer C., Salesin D.: Multiperspective panoramas for cel animation. Computer Graphics (ACM SIGGRAPH), Los Angeles, USA, August 3–8, 243–250.

1998

[11] Benosman R., Maniere T., Devars J.: Panoramic stereovision sensor. Proc. IEEE ICPR, Brisbane, Australia, August 16–20, I:767–769.

[12] Prihavec B., Solina F.: User interface for video observation over the internet. Journal of Network and Computer Applications, 21, 219–237.

[13] Rademacher P., Bishop G.: Multiple-center-of-projection images. Computer Graphics (ACM SIGGRAPH), Orlando, USA, July 19–24, 199–206.

1999

[14] Nayar S. K., Peri V.: Folded Catadioptric Camera. Proc. IEEE CVPR, Fort Collins, USA, June 23–25, II:217–223.

[15] Peleg S., Ben-Ezra M.: Stereo panorama with a single camera. Proc. IEEE CVPR, Fort Collins, USA, June 23–25, I:395–401.

[16] Shum H. Y., Szeliski R.: Stereo reconstruction from multiperspective panoramas. Proc. IEEE ICCV, Kerkyra, Greece, September 20–25, I:14–21.

2000

[17] Hartley R., Zisserman A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge, UK.

[18] Huang F., Pajdla T.: Epipolar geometry in concentric panoramas. Technical Report CTU-CMP-2000-07, Center for Machine Perception, Czech Technical University, Prague, Czech Republic. Available at: ftp://cmp.felk.cvut.cz/pub/cmp/articles/pajdla/Huang-TR-2000-07.ps.gz .

[19] Nayar S. K., Karmarkar A.: 360×360 Mosaics. Proc. IEEE CVPR, Hilton Head Island, USA, June 13–15, II:388–395.

[20] Peleg S., Pritch Y., Ben-Ezra M.: Cameras for stereo panoramic imaging. Proc. IEEE CVPR, Hilton Head Island, USA, June 13–15, I:208–214.
[21] Peleg S., Rousso B., Rav-Acha A., Zomet A.: Mosaicing on adaptive manifolds. IEEE Trans. PAMI, 22(10), 1144–1154.
[22] Svoboda T.: Central Panoramic Camera Design, Geometry, Egomotion. Ph.D., Center for Machine Perception, Czech Technical University, Prague, Czech Republic. Available at: ftp://cmp.felk.cvut.cz/pub/cmp/articles/svoboda/phdthesis.ps.gz .
[23] Svoboda T., Pajdla T.: Panoramic cameras for 3D computation. Proc. Czech Pattern Recognition Workshop, Prague, Czech Republic, 63–70.

2001

[24] Huang F., Wei S. K., Klette R.: Geometrical Fundamentals of Polycentric Panoramas. Proc. IEEE ICCV, Vancouver, Canada, July 9-12, I:560–565.
[25] Peleg S., Ben-Ezra M., Pritch Y.: Omnistereo: Panoramic Stereo Imaging. IEEE Trans. PAMI, 23(3), 279–290.

Peter Peer is a research assistant at the Faculty of Computer and Information Science, University of Ljubljana, Slovenia. He received a B.Sc. and a M.Sc. in computer science from the University of Ljubljana in 1998 and 2001, respectively. Currently he is moving towards his Ph.D. in computer science. His research interests include stereo reconstruction, face detection and real time applications.

Franc Solina is professor of computer science at the University of Ljubljana, Slovenia and head of Computer Vision Laboratory at the Faculty of Computer and Information Science. He received a B.Sc. and a M.Sc. in electrical engineering from the University of Ljubljana in 1979 and 1982, respectively, and a Ph.D. in computer science from the University of Pennsylvania, USA in 1987. His research interests include range image interpretation, segmentation and 3D shape reconstruction.