

Panoramic Depth Imaging with a Single Standard Camera

Peter Peer and 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 article 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 stereo pair of panoramic images the epipolar geometry needs to be determined.

We focused mainly on the system analysis. The system performs well in the reconstruction of small indoor spaces.

1. Introduction

Standard camera has 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 [9].

We wish the images would have the property that points and lines are visible on all of the scene images. This is the property of panoramic cameras and it presents our fundamental motivation.

On the small photograph within Figure 1 you can see the hardware part of our system: rotational arm is rotating around the vertical axis, a pole is fixed on the rotational arm enabling the offset of the optical center of the camera from the system's rotational center and on it we fixed one standard color camera looking outwards from the system's rotational center. Panoramic images are generated by moving the rotational arm for angle corresponding to one column of the captured image.

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 stripes on the left and on the right side from the captured image center column.

In the next section we will overview the related work and expose the contribution of our work towards the discussed subject. Section 3 will describe the geometry of our system, Section 4 epipolar geometry and Section 5 procedure of stereo reconstruction. The focus of this article is on system capabilities analysis, which is given in Section 6.

2. Related work

One of the best known commercial packages for creating mosaicked panoramic views is QTVR (QuickTime Virtual Reality). It works on the principle of sewing together a number of standard images captured while rotating the camera [1]. Peleg et al. [8] introduced the method for creation of mosaicked panoramic views from standard images captured with a handheld video camera. A similar method was suggested by Szeliski and Shum [12] which also do not strictly constrain the camera path but assume that there is no great motion parallax effect present. All the methods mentioned so far are used only for visualization purposes since the authors did not try to reconstruct the scene.

Ishiguro et al. [5] suggested a method which enables the reconstruction of the scene. They used a standard camera rotating on a circle. The scene is reconstructed by means of mosaicking together panoramic images from the central column of the captured images and moving the system to another location where the task of mosaicking is repeated. Two created panoramas are then used as input in stereo reconstruction procedure. The depth of an object was first estimated using projections in two images captured on 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.

Peleg and Ben-Ezra [6, 7] introduced the method for creation of stereo panoramic images. Stereo panoramas are created without actually computing the 3D structure — the depth effect is created in viewer’s brain.

In [11] 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 the circular path. Panoramas are build by taking one column out of captured image and mosaicking the columns. They call such panoramas *multiperspective panoramas*. The crucial property of two or more multiperspective panoramas is that they capture the information about the motion parallax effect, while the columns forming the panoramas are captured from different perspectives. The authors are using such panoramas as the input in stereo reconstruction procedure.

However multiperspective panoramas are not something entirely unknown to vision community [11]: they are a special case of *multiperspective panoramas for cel animation* [13], they are very similar to images generated with a procedure called *multiple-center-of-projection* [10], procedure *manifold projection* [8] and procedure *circular projection* [6, 7]. The principle of constructing the multiperspective panoramas is also very similar to *the linear pushbroom camera* principle for creating the panoramas [3].

In articles closest to our work [5, 11] we missed two things: system capabilities analysis and searching for corresponding points using the standard correlation technique. This is why we focus ourselves in this article on these two issues. While in [5] authors searched for corresponding points by tracking the feature from the column building on panorama to the column building the second panorama, the authors in [11] used an upgraded *plain sweep stereo* procedure.

3. System geometry

The geometry of our system for creating multiperspective panoramic views is shown in Figure 1. When created they are 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 outwards from the rotational center. Optical center of the camera is marked with O . Selected column of pixels that will be sewn in panoramic view 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 defines the angle between the line defined by point C and point O and the line defined by point C and point P . In panoramic view θ gives the hori-

zontal axis describing the path of the camera. With φ we denoted the angle between line defined by point O and the middle column of pixels of captured image and the line defined by point O and selected column of pixels that will be mosaicked in panoramic view. Angle φ can be thought of as a reduction of the camera’s horizontal view angle α .

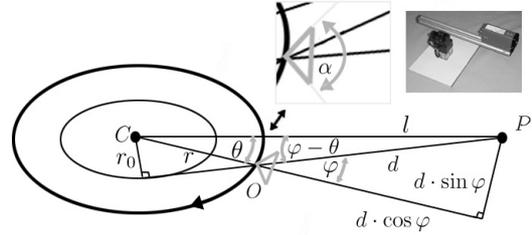


Figure 1. Geometry and the hardware part of our system for constructing a multiperspective panorama.

System on Figure 1 is obviously a non-central since the light rays ($\varphi \neq 0$) forming the panoramic view are not intersecting in one point called the viewpoint, but instead are tangent to a circle with radius r_0 called the viewing circle. Thus we are dealing with panoramic views formed by projection from a number of viewpoints. This means that some point on the scene will be seen in panoramic view only from one viewpoint.

4. Epipolar geometry

Since we are limited with the length of the article we will only illustrate the procedure of the proof that the epipolar lines of the symmetrical pair of panoramic views are image rows [4, 11].

The proof is based on radius r_0 of the viewing circle (Figure 1). We can express r_0 in terms of known quantities r and φ as: $r_0 = r \cdot \sin \varphi$. We can treat value r_0 as the radius of the captured panoramic view while we get the same panoramic view in a case if we rotate a line camera on a circular path with radius r_0 and with a line camera’s optical axis tangent to this circle.

We can carry out the proof in three steps: *first*, we have to execute the projection equation for line camera, *then* we have to write the projection equation for multiperspective panoramic view and in the *final* step we can prove the property of epipolar lines for the case of a symmetrical pair of panoramic views. In the first step we are interested in how the point on the scene is projected to the camera’s image plain [2] 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 relations between different notations



of a point on the scene and projection of this point on the panoramic view: 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 view coordinate system and in pixel coordinates of the (2D) panoramic view coordinate system. When we know the relations between mentioned coordinate systems we can write the equation for projection of scene points on image plain (cylinder) of the panorama. Based on angular coordinates of the panoramic view coordinate system property, we can in the third step show that the epipolar lines of the symmetrical pair of panoramic views are actually rows of panoramas. The basic idea for the last step of our proof is as follows: If we are given an image point on one panoramic view, 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 view, we get an epipolar line corresponding to given image point on the first panoramic view.

5. Stereo reconstruction

Let us go back to Figure 1. 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:

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

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

$$l = \frac{r \cdot \sin(180^\circ - \varphi)}{\sin(\varphi - \theta)} = \frac{r \cdot \sin \varphi}{\sin(\varphi - \theta)}. \quad (1)$$

From eq. (1) follows that we can estimate the 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 as:

$$2\varphi = \frac{\alpha}{W} \cdot W_{2\varphi}, \quad (2)$$

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 views, given also in pixels. To calculate the angle θ we have to find corresponding points on panoramic views. 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:

$$\theta = dx \cdot \frac{\theta_0}{2}, \quad (3)$$

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

We are using a procedure called "normalized correlation" to search for corresponding points. To increase the confidence in estimated depth we are using procedure called "back-correlation" [2]. With the back-correlation we are also solving the problem of occlusions.

6. System capabilities analysis

6.1. Time complexity of creating a panoramic view

The biggest problem of our system is that it can not work in real time since we are creating panoramic views by rotating the camera for a very small angle. 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 view is much too long, so there is no feasibility to make it work in real time.

In one circle around the system vertical axis our system constructs 11 panoramic views, it captures 1501 images with resolution of 160×120 pixels, where radius is $r = 30$ cm and the shift angle is $\theta_0 = 0.2^\circ$. The process takes a bit more than 15 minutes on PC Intel PII./350 MHz to end.

6.2. Constraining the search space on the epipolar line

Knowing that the width of the panoramic view is much bigger than the width of the captured image, we would have to search for corresponding point along a very long epipolar line. That is why we would really like to constraint the search space on the epipolar line as much as possible. A side effect is also increased confidence in estimated depth and a faster execution of the stereo reconstruction procedure.

If we derive from eq. (1) we can ascertain two things which nicely constraint the search space:

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.

Theoretically, the estimation of depth is not constrained upwards, but from eq. (1) is evident that the denominator

must be non-zero. We can write this fact as: $\theta_{\max} = n \cdot \frac{\theta_0}{2}$, where $n = \varphi \operatorname{div} \frac{\theta_0}{2}$ and $\varphi \bmod \frac{\theta_0}{2} \neq 0$.

If we write the constraint for the last point, that 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 on the right eye panorama to including horizontal image coordinate $x + \frac{2\theta_{\max}}{\theta_0} = x + n$, where x is the horizontal image coordinate of the point on the left eye panorama for which we are searching the corresponding point.

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 the 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 a case of $2\varphi = 29.9625^\circ$ and only $n = 18$ in the case of $2\varphi = 3.6125^\circ$, when searching for corresponding point.

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

6.3. The meaning of the error for a pixel in estimation of angle θ

Before we illustrate the meaning of the error for a pixel in estimation of angle θ , let us take a look at graphs on Figure 2. Graphs are showing dependence of depth function l from angle θ while using different values of angle φ . From the graphs it is evident that the depth function l is rising slower in a case of 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 Figure 2). 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 error for a pixel in estimation of angle θ is much smaller when using bigger angle φ , since a shift for angle θ_0 describes the shift of the camera for one column of pixels.

Because of discrete horizontal axis θ (Figure 2) 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 angle φ : we can calculate ($\varphi \operatorname{div} \frac{\theta_0}{2} =$)149 depth values if we are using the angle $2\varphi = 29.9625^\circ$ and only 18

depth values if we are using the angle $2\varphi = 3.6125^\circ$. And this is the negative property of smaller angle φ .

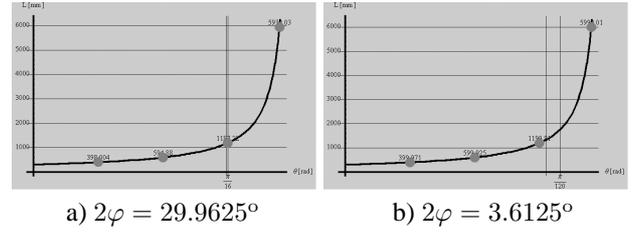


Figure 2. Graphs showing dependence of depth function l from angle θ while radius $r = 30$ cm and using different values of angle φ . To ease the comparison of the error for a pixel in estimation of angle θ we showed the interval of width $\frac{\theta_0}{2} = 0.1^\circ$ between the vertical lines around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

Results in table 1 give the values of the error for a pixel in estimation of angle θ for different values of parameters θ and φ .

From the results we can conclude that the error is much bigger in case of smaller angle φ than in case of bigger angle φ . The speed of the reconstruction process is inversely proportional to the accuracy of the process.

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

With increasing the resolution of captured image we are decreasing the angle θ_0 and subsequently decreasing the rotational angle of the camera between two successively captured images forming the panoramas. For nearly the same factor as we increase (decrease) the resolution of captured image, we decrease (increase) the value of error Δl , while the reconstruction process takes for nearly the same factor more (less) time to end. We can treat the increase in resolution of captured image as the sub-pixel accuracy procedure.

For the same factor that we increase (decrease) radius r , we increase (decrease) the (biggest possible and sensible) depth estimation l and size of the 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 fact is that our system is capable of doing the reconstruction of (small) rooms and it will not perform well in the reconstruction process of an outdoor scene. This is due to the property of the system while 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.

6.4. Definition of maximal depth in which we trust

In section 6.2 we defined the minimal possible depth estimation l_{\min} and maximal possible depth estimation l_{\max} ,

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

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

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

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

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

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

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

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

Table 1. The meaning of the error for a pixel in estimation of angle θ (equation (3)).

but we did not write anything about the meaning of the error for a pixel in estimation of angle θ for these two estimated depth. Let us examine the size of the error Δl for these two estimated depths: we calculate Δl_{\min} as absolute value of difference between the depth l_{\min} and the depth l for which the angle θ is bigger from angle θ_{\min} for 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 Δl_{\max} as absolute value of difference between the depth l_{\max} and the depth l for which the angle θ is smaller from angle θ_{\max} for 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 with variable n we denote positive number following from equation: $n = \varphi \operatorname{div} \frac{\theta_0}{2}$.

In table 2 we gathered the error sizes for different values of angle φ . The results confirm already written conclusions (section 6.3), thus we can add two additional conclusions:

The value of error Δl_{\max} is unacceptable high and this is true regardless of the value of angle φ . This is why we have to sensibly decrease the maximal possible depth estimation l_{\max} . This conclusion in practice leads us to define

	$2\varphi = 29.9625^\circ$	$2\varphi = 3.6125^\circ$
Δl_{\min} [mm]	2	19
Δl_{\max} [mm]	30172	81587

Table 2. The meaning of the error (Δl) for one pixel in estimation of angle θ for l_{\min} and l_{\max} regarding the angle φ .

the upper boundary of allowed error size (Δl) for one pixel in estimation of angle θ and with it, we subsequently define the maximal depth in which we trust.

Angle φ is always depending upon the horizontal view angle α of the camera (equation (2)). And while the angle α is limited with around 40° considering standard cameras, our system is limited with 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 in the case of reconstruction of small rooms.

6.5. Results

Figure 3 shows some results of our system. In the case denoted with b), we constructed the dense panoramic view, 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 normalized correlation estimations. In this way, we eliminate from the dense depth image all the associated depth estimates which do not have a high enough associated confidence estimation. In the case marked with e), we created a sparse depth image by searching only for the corresponding points of features on input panoramas. The features can be presented as vertical edges on the scene, which we can derive very fast if we filter the panoramas with the Sobel filter for searching the vertical edges [2, 5]. If we would use a smaller value for angle φ , the reconstruction times would be up to eight times smaller from presented ones.

With respect to the presented reconstruction times we could conclude that the reconstruction procedure could work in nearly real time, if we would work with 8-bit grayscale images (and not with the color images as in our case), with lower resolution and/or if we would create the sparse depth image of only part of the scene. This could be used for robot navigation [5].



Figure 3. 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: 6:42:20 (h:m:s), c) the information about the confidence in estimated depth, d) dense depth image after the weighting / not using back-correlation / reconstruction time: 3:21:56, e) sparse depth image / not using back-correlation / reconstruction time: 0:0:38.

7. Outlook

We presented an exhaustive analysis of our system for construction of depth panoramic images using only one standard camera. We proved the following: the procedure

for creating panoramic views 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 views are image rows; based on the equation for estimation of depth l , we can constraint the search space on the epipolar line; confidence in estimated depth is changing: bigger the slope of the function l curve, smaller the confidence in estimated depth; if we observe only the reconstruction time, we can conclude that the creation of dense panoramic images is too expensive.

References

- [1] S. Chen. Quicktime VR — an image-based approach to virtual environment navigation. In *ACM SIGGRAPH*, pages 29–38, Los Angeles, USA, 1995.
- [2] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, Massachusetts, London, England, 1993.
- [3] R. Gupta and R. I. Hartley. Linear pushbroom cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):963–975, September 1997.
- [4] F. Huang and T. Pajdla. Epipolar geometry in concentric panoramas. Technical Report CTU-CMP-2000-07, Center for Machine Perception, Czech Technical University, Prague, Czech Republic, March 2000.
- [5] H. Ishiguro, M. Yamamoto, and S. Tsuji. Omni-directional stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):257–262, February 1992.
- [6] S. Peleg and M. Ben-Ezra. Stereo panorama with a single camera. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 395–401, Fort Collins, USA, 1999.
- [7] S. Peleg, Y. Pritch, and M. Ben-Ezra. Cameras for stereo panoramic imaging. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 208–214, Hilton Head Island, USA, 2000.
- [8] S. Peleg, B. Rousso, A. Rav-Acha, and A. Zomet. Mosaicing on adaptive manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1144–1154, October 2000.
- [9] B. Prihavec and F. Solina. User interface for video observation over the internet. *Journal of Network and Computer Applications*, 21:219–237, 1998.
- [10] P. Rademacher and G. Bishop. Multiple-center-of-projection images. In *Computer Graphics (ACM SIGGRAPH)*, pages 199–206, Orlando, USA, 1998.
- [11] H. Y. Shum and R. Szeliski. Stereo reconstruction from multiperspective panoramas. In *IEEE 7th International Conference on Computer Vision*, pages 14–21, Kerkyra, Greece, 1999.
- [12] R. Szeliski and H. Y. Shum. Creating full view panoramic image mosaics and texture-mapped models. In *Computer Graphics (ACM SIGGRAPH)*, pages 251–258, Los Angeles, USA, 1997.
- [13] D. Wood, A. Finkelstein, J. Hughes, C. Thayer, and D. Salesin. Multiperspective panoramas for cel animation. In *Computer Graphics (ACM SIGGRAPH)*, pages 243–250, Los Angeles, USA, 1997.