EXTENSION OF EPIPOLAR IMAGE ANALYSIS TO CIRCULAR CAMERA MOVEMENTS

Ingo Feldmann, Peter Eisert, Peter Kauff

Fraunhofer Institute for Telecommunications, Heinrich-Hertz-Institute, Berlin, Germany
Email:{feldmann,eisert,kauff}@hhi.fhg.de

ABSTRACT

Epipolar image analysis is a robust method for 3D scene depth reconstruction that uses all available views of an image sequence simultaneously. It is restricted to horizontal, linear, and equidistant camera movements. In this paper, we present a concept for an extension of epipolar image analysis to other camera configurations like, e.g., circular movements. Instead of searching for straight lines in the epipolar image, we explicitly compute the trajectories of particular points through the image cube. Variation of the unknown depth leads to different curves. From those, the one corresponding to the true depth is selected by evaluating color constancy along the curve. In order to handle occlusions correctly an explicit occlusion compatible ordering scheme is derived for the case of circular movements. To compensate the influence of perspective projection we introduce a depth corrected epipolar image analysis algorithm which we call image cube trajectory analysis (ICT).

1. INTRODUCTION

The estimation of depth information from 2D images has received much attention in the past decade. The basic problem of recovering the 3D structure of a scene from a set of images is the correspondence search [1]. Given a single point in one of the images its correspondences in the other images need to be detected. Depending on the algorithm two or more point correspondences as well as the camera geometry are used to estimate the depth of that point [2]. However, for complex real scenes the correspondence detection problem is still not fully solved. Especially in the case of homogeneous regions, occlusions or noise, it still faces many difficulties. It is now generally recognized that using more than two images can dramatically improve the quality of reconstruction.

In Epipolar Image (EPI) analysis [3] all available image frames are considered simultaneously. The existence of EPIs is restricted to an equidistant linear camera movement parallel to the horizontal axis of the image plane. Only in this case the epipolar planes (defined by any 3D point and two arbitrary cameras) are the same for all camera positions of the sequence [1]. The corresponding epipolar lines define a single EPI. Criminisi et al. show that EPIs can be thought of being horizontal slices (or planes) in the so called *image cube* [4]. An image cube can be constructed by collating all images of a sequence (see fig. 1, left). Each EPI represents a single horizontal line (Y=constant) of all camera image planes simultaneously (see fig. 1, right).

For a linear camera movement parallel to the horizontal axis of the image plane all projections of 3D object points remain in the same EPI throughout the entire sequence. Thus, the EPI represents the trajectories of object points. If the camera is moved equidistantly, the path of an arbitrary 3D point becomes a straight

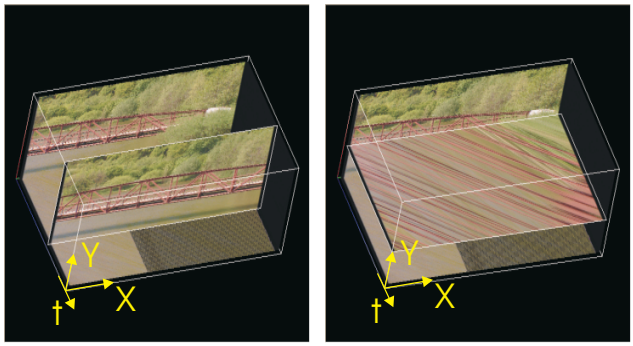

Fig. 1. Image cube representation of an image sequence, **left)** time slices (fixed t) represent images, **right)** horizontal slices (fixed Y) represent EPIs

line, called *EPI line* (see fig. 2, right) . The slope of the line represents the depth. The principle of EPI analysis is the detection of all EPI-lines (and their slopes) in all available EPIs.

The advantage of the EPI analysis algorithm is the simultaneous detection of point correspondences for all available views. Occlusions as well as homogeneous regions can be handled efficiently [4]. The big disadvantage of the algorithm is its restriction to linear camera movements. One idea to overcome this problem is the piecewise linear EPI analysis. In this case, the EPIs are constructed only from those images of the sequence where the assumption of linear equidistant camera motion is approximately fulfilled. In [5], Li et al. present an EPI construction and analysis algorithm for multi-perspective panoramas where a camera rotates on a predefined circle. In this case, the paths of arbitrary 3D points in the image cube are not longer lines on a horizontal slice as illustrated on the right hand side of fig. 3. However, small segments can still be approximated by straight lines which is utilized by the authors to determine depth of the corresponding points. Unfortunately, this approach significantly reduces the amount of reference images available for the 3D reconstruction.

2. OUR APPROACH

In this paper, we propose a new concept for the extension of EPI analysis to other parameterized camera movements. Without loss of generality we will discuss our approach for the case of a circular camera path. Similar to EPI analysis we analyse the path structure of the projected 3D points in the image cube. We call such paths *Image Cube Trajectories* (ICT).

We suggest an inverse approach to the conventional way of EPI analysis where usually in a first step the EPI lines are detected in the EPI using some kind of segmentation algorithm [3, 5]. In a second step, the corresponding depth is reconstructed from the

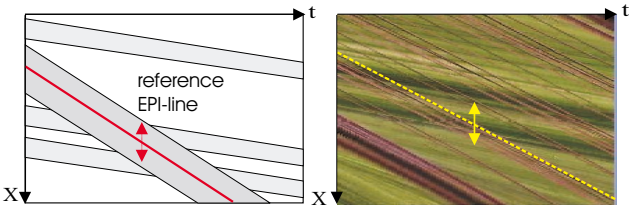

Fig. 2. Linear camera movement, **left)** occlusion handling in EPIs, **right)** principle of line matching in EPIs

slopes of the lines. In contrast, our idea is to specify an ICT for an assumed 3D point by determining the slope and intercept from its image position and assumed depth. We will call this the *reference path*. In a second step we are looking in the image cube if this assumption might be valid and such a path exists. If not, the depth will be changed until the resulting ICT fits to the image cube. At a first glance, this approach seems to be very similar to EPI analysis. But we will see in chapter 4 that especially for non-linear camera movements like circular motion, our approach has many advantages. To create the above mentioned reference paths we need to calibrate the camera system such that all camera positions are known in advance. Robust camera self calibration systems are well known in the literature [6].

One of the benefits of the conventional EPI analysis is that the structure of the EPI-lines (i.e. a straight line) is known in advance. Therefore, it is possible to design precisely adapted segmentation algorithms with powerful edge detection filters [3]. Moreover, the occlusion handling becomes very easy and efficient (see chapter 3). However, we will show in chapter 4 that the ICTs have a well defined structure in the image cube for circular camera movements as well. Therefore, it is possible to adapt the path detection algorithms to the expected shape of those paths, too. Similar to existing EPI methods, it is also possible to define an occlusion compatible ordering for this case to efficiently handle occlusions in the image sequence.

In the next chapters we will first describe our approach for the case of a linear horizontal equidistant camera movement. Afterwards, we will show its extension to the concentric circular movement for orthographic and perspective cameras. Our main goal is to discuss in which way the properties of conventional EPI analysis can be extended to circular camera movements. This method of extension can also be applied to the ICT analysis of other parametrized camera paths, such as ellipses, parabolas etc. as well.

3. LINEAR CAMERA PATH

In this chapter we will consider the restrictions of the conventional EPI analysis (linear equidistant camera movement parallel to the horizontal axis of the image plane). In this case the projected path of any 3D point in the image cube is a straight line as illustrated on the right hand side of Fig. 2. To reconstruct the 3D scene structure, we perform a full search over the entire visible 3D space. For each pixel in a particular image the corresponding depth is varied between a near and far clipping plane and the two straight line parameters slope and intercept are determined from depth and pixel location. The closer the point is to the camera the larger is the slope of the EPI line. The line that matches best in the EPI image is considered to represent the true depth.

At this point, it is important to consider the occlusion compatible ordering of the EPI-lines described in [4]. 3D points which are closer to the cameras "move" faster (i.e. have larger disparities in the camera image planes) than those which are further away. Therefore, an EPI-line of a point with a small depth will have a larger slope than that of a more distant point. Because closer 3D points always hide those which are further away, the EPI-lines with larger slopes always hide those with smaller slopes. The left side of fig. 2 gives an example of this ordering.

Due to the occlusion ordering the EPI-lines with the largest slope need to be detected first. Successfully detected EPI-line are masked and excluded from subsequent analysis steps. In this way, our algorithm tries to detect 3D points in the scene scanning it from front to back. Because 3D points at the same depth but different horizontal and vertical positions do not occlude each other they can be detected simultaneously without disturbing the occlusion compatible search order.

4. CIRCULAR CAMERA PATH

The goal of this chapter is to extend the EPI analysis algorithm to the case of circular camera movements. Because EPIs are not defined for this case the proposed ICT algorithm is used. Without loss of generality we will restrict our discussion to pure horizontal camera movements. The camera always points to the midpoint M of the circular camera path. For easier understanding we will consider a fixed camera and a rotating scene instead of the moving camera configuration. Our discussion is split into two parts: Firstly, we will explain the general algorithm and the handling of occlusions for the theoretical case of an orthographic camera. Secondly, we will discuss the influence of the perspective projection to the path structure.

4.1. General Algorithm and Occlusion Handling

Fig. 4 left shows the image plane IP of an orthographic camera and 3 different scene points A, B, and C rotating circularly at radius R_i around the midpoint M at the same rotational speed. Fig. 4 right illustrates the projection of these points into the image plane IP (i.e. their ICTs) for a full circle. For orthographic projection they are located on a horizontal plane within the image cube similar to EPI analysis. Fig. 3 illustrates this for the example of the real sequence "Flower". The left image shows the first frame of the sequence and the right one an example for the ICTs constructed from row Y=180. It can be easily shown that the ICT of a point P can be described by the sinusoidal function $X_P = R_P \sin(\phi + \Delta\phi_P)$ where $\Delta\phi_P$ is the initial phase of that point and $\phi = \omega t$ the phase varying between $-\pi < \phi \leq \pi$. Any of the rotating points has two degrees of freedom: The radius R_P and the initial phase $\Delta\phi_P$. The radius corresponds to the amplitude of the ICT whereas $\Delta\phi_P$ determines the phase of the curve. In this way, it is possible to reconstruct the 3D position of any point from its ICT. Vice versa, one can construct for each 3D point the corresponding ICT in the image cube.

Another task is the handling of occlusions and disocclusions. Due to the parameterized circular movement of the points it is possible to derive rules for an occlusion compatible ordering of the paths. For the following discussion it is necessary to split each of the ICTs into four subsections which correspond to their quadrants. A first rule is that for a circular movement all points in the quadrants I and IV will occlude those in the quadrants II and III if their projection rays are equal, i.e. their ICTs intersect each other (see figs. 3 and 4). The path sections in both cases have an inverse slope to each other which depends on the orientation of the coordinate system as well as the direction of rotation. The figures show

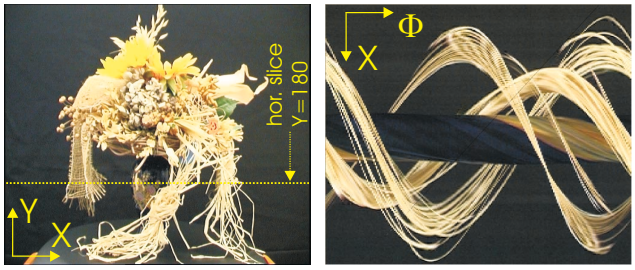

Fig. 3. "Flower" sequence, circular moving camera, **left)** first frame and, **right)** example for ICT representation for row Y=180

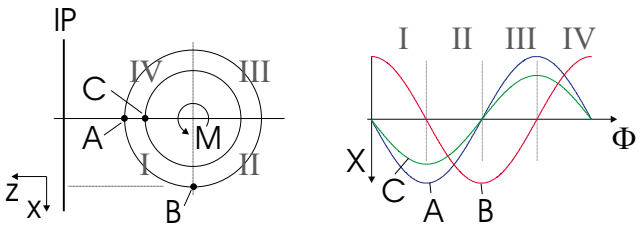

Fig. 4. Orthographic camera, **left)** projection of points A,B,C to image plane IP, **right)** ICTs of the points in the image cube

a mathematically positive rotation around the y-axis. Therefore, its projection, i.e. the ICT sections will have an increasing slope in quadrants I and IV and, vice versa, a decreasing slope in quadrants II and III. In fig. 3 right, we can see that all sections of all ICTs with increasing slopes occlude those with decreasing slopes in case of intersection. Note, that in the figures increasing slopes point downwards due to the chosen coordinate systems.

The second rule is that in quadrants I and IV all points with a larger radius will occlude those with a smaller one if they have the same phase and their projection rays are equal. The reason is that they are closer to the camera. Vice versa, in the quadrants II and III points with a small radius occlude those with a larger one. In the example in fig. 4, point A occludes point C. Vice versa C would occlude A if the points were shifted by a phase of π.

Due to both rules it is necessary to split the occlusion compatible ICT detection into two parts: In a first step only quadrants I and IV are considered. The algorithm starts with the maximum expected radius and tries to detect (by varying the phase) if there are ICTs in the image cube which correspond to points on that radius. During this operation only those sections of the considered ICTs are analyzed which correspond to quadrants I and IV. In other words, the algorithm would search only for path sections with an increasing slope (for mathematical positive rotation). Note that those path sections do not intersect each other. Similar to the linear approach each successfully detected trajectory section will be masked and excluded from further processing. If the search for the given radius is finished, the algorithm chooses the next smaller radius and repeats the whole procedure.

For closed objects without holes it is possible to stop the searching procedure at the smallest radius. If the object is not fully closed the points from quadrants II and III are partly projected into the image plane as well. This is the case in the example in fig. 3 right. To detect those trajectory sections, we need to repeat the above procedure looking for trajectory sections with the inverse slope. The main difference is that we now start with the smallest possible radius and end with the largest one. Note, that if a scene point is visible in all quadrants then all parts of its corresponding ICTs with increasing slope are detected separately from those with decreasing slope. This is an additional condition to detect mismatches in the algorithm. Finally, the entire process must be repeated for all available ICTs, i.e. for the whole image cube.

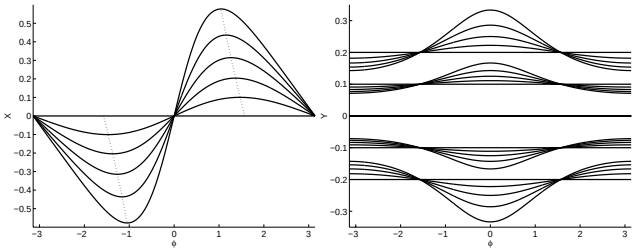

Fig. 5. ICT of a point with increasing radius **left)** X coordinate, **right)** Y coordinate as a function of phase ϕ

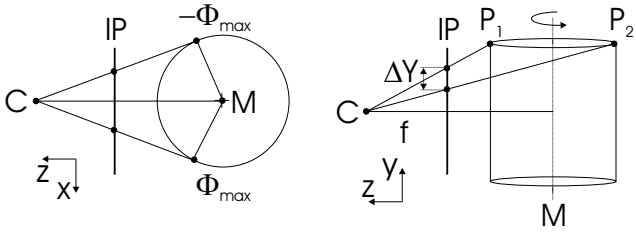

Fig. 6. Perspective camera case, **left)** X-deviation, **right)** Y-deviation

4.2. Influence of Perspective Projection

If a perspective camera with focal length f is used instead of the orthographic one of the previous section, some modifications to the ICT analysis have to be made. Due to the depth dependency in the projection, the ideal sinusoidal curves are slightly distorted. Moreover, the ICT of a particular 3D object point is no longer restricted to lie on a horizontal plane in the image cube but must be considered as a curve in the three-dimensional video cube. The shape of this curve, however, can explicitly be computed from radius R and phase ϕ resulting in the same two degrees of freedom in the ICT analysis as in the orthographic case. The basic algorithm with curve detection and occlusion handling also remains the same but the valid visible range of a point in a curve is slightly reduced. All these effects are described and quantified in the following section.

Consider the 3D object point $\mathbf{x}$

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} R\sin\phi \\ y \\ z_0 + R\cos\phi \end{bmatrix} \tag{1}$$

that moves on a circle with radius R around M (having a distance of $-z_0$ from the focal point of the camera) and height y. Perspective projection into the image plane leads to the following 2D image coordinates X and Y

$$\begin{aligned} X &= -f\frac{x}{z} = \frac{-f\frac{R}{z_0}\sin\phi}{1+\frac{R}{z_0}\cos\phi} = \frac{fq\sin\phi}{1-q\cos\phi} \\ Y &= -f\frac{y}{z} = \frac{-f\frac{y}{z_0}}{1+\frac{R}{z_0}\cos\phi} = \frac{fp}{1-q\cos\phi} \end{aligned} \tag{2}$$

with the abbreviations

$$q = -\frac{R}{z_0} \qquad p = -\frac{y}{z_0}. \tag{3}$$

The curve $X(\phi)$ in (2) is no longer a perfect sinus as in the orthographic case but slightly distorted. This is illustrated on the left hand side of fig. 5 where the function is plotted for different values of q from 0 to 0.5. Object points with larger radius R and thus larger value q deviate more from the sinusoidal curve. The phase of the maximum amplitude ϕ_{max} is given by

$$\cos(\phi_{max}) = q. \tag{4}$$

The image coordinate in vertical direction $Y(\phi)$ is also influenced for perspective projection as illustrated on the right hand side of fig. 6. Points that are closer to the camera are shifted towards the horizontal image borders. The exact curves specified in (2) are plotted in fig. 5 right for different values of q. All points, except those which are located in the horizontal plane through the optical axis of the camera, have varying Y-coordinates. Since the magnitude of these variations is also dependent on q there does not exist a common 2D surface through the video cube (like the EPI images) that contains multiple object points for the entire object rotation. Therefore, the ICT analysis for perspective projection has to be performed on the 3D video cube. For large focal lengths f and large object distances from the camera (small q and p) the Y-coordinate in (2) becomes constant again whereas $X(\phi)$ tends to the sinusoidal curve of the orthographic case.

The general algorithm of the ICT analysis and the occlusion handling remains the same as for an orthographic camera. Only the visible range of an object point is somewhat reduced as illustrated on the left side of fig. 6. The separation between quadrants I and II as well as between III and IV is specified by $\pm\phi_{max}$. The location of these points can be derived from (4).

5. EXPERIMENTAL RESULTS

In this section, first preliminary experiments on synthetic and real image sequences are presented. We apply the proposed inverse search strategy described in section 2 to the case of horizontally moving cameras where the trajectories can be modeled as straight lines. Following the occlusion compatible ordering mentioned in section 3, depth is varied from small to large values corresponding to decreasing slopes of ICTs that are matched with the image cube (i.e. the EPIs for this case). To detect a reference ICT in the image cube, we are using a straightforward approach: The color variation along the whole ICT is compared to a given threshold. If it is less than this value we consider the ICT as detected. Although the described matching algorithm is very primitive, it still provides reasonable results and demonstrates the general idea of our approach quite well. Fig. 7 shows the result of a 3D reconstruction of a virtual and fig. 8 for a natural scene. Both sequences have an image resolution of 360 by 288 pixels and approximately 200 frames. The average computing time on a conventional 2GHz CPU was approximately 5 minutes for a full 3D reconstruction of 200 depth maps. All depth maps are generated without any filtering, interpolation, or postprocessing. As we can see from the figures, it is possible to reconstruct the depth maps in both cases with reasonable quality. It is left to future work to apply the matching algorithm to other camera movements and to improve it using more sophisticated methods for path detection, e.g., by exchanging the simple threshold operation by a search for the best match among all slopes. This would also remove the remaining small holes in the depth map.

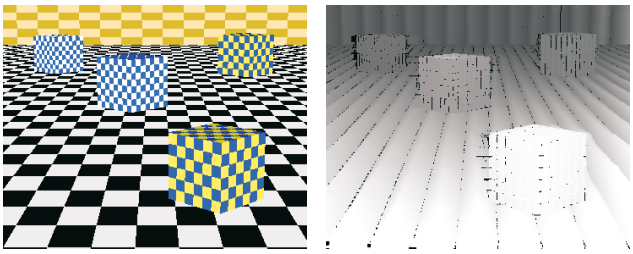

Fig. 7. "Boxes", virtual sequence with linear horizontal camera movement, **left)** original image, **right)** estimated depth

Fig. 8. "Bridge", real sequence with linear horizontal camera movement, **left)** original image, **right)** estimated depth

6. CONCLUSIONS AND FUTURE WORK

In this paper we have discussed the concept for an extension of conventional epipolar image analysis to the case of a circular moving camera which we call ICT analysis. We have shown that well defined trajectory structures in the image cube exist for this case which can be used to derive rules for a systematic occlusion compatible search algorithm. The proposed algorithm is not restricted to circular camera movements only. Our extension to EPI analysis can be used to define rules for other parameterized camera movements such as parabolic camera paths etc. as well. Therefore, we are working on more generalized rules and algorithms to solve this problem. Another topic of our research are more sophisticated path matching algorithms as well as the handling of homogeneous regions.

7. REFERENCES

[1] J. P. Mellor, S. Teller, and T. Lozano-Perez, "Dense depth maps from epipolar images," Tech. Rep. AIM-1593, 1996.

[2] P. Beardsley, P. Torr, and A. Zisserman, "3D model acquisition from extended image sequences," in *Proc. European Conference on Computer Vision (ECCV)*, 1996.

[3] R. C. Bolles, H. H. Baker, and D. H. Marimont, "Epipolar image analysis: An approach to determine structure from motion," *International Journal of Computer Vision*, pp. 7–55, 1987.

[4] A. Criminisi, S. B. Kang, R. Swaminathan, R. Szeliski, and P. Anandan, "Extracting layers and analyzing their specular properties using epipolar-plane-image analysis," Tech. Rep. MSR-TR-2002-19, Microsoft Research, 2002.

[5] Y. Li, C.-K. Tang, and H.-Y. Shum, "Efficient dense depth estimation from dense multiperspective panoramas," in *Proc. International Conference on Computer Vision (ICCV)*, Vancouver, B.C., Canada, Jul. 2001, pp. 119–126.

[6] C. Rothwell, G. Csurka, and O. D. Faugeras., "A comparison of projective reconstruction methods for pairs of views," Tech. Rep. 2538, INRIA, Jul. 1995.