Model-Free Multi-Camera Calibration By Graph Pyramid Using 360° Pattern

Majid Banaeyan
Pattern Recognition and Image Processing Group 193/03, TUWien, Vienna, Austria

Abstract
Camera calibration should be easy to perform on the one hand and provide robust results on the other hand. These requirements motivate the proposed, "unusual" approach for multi-camera calibration. For camera calibration, common methods assume a geometrical model to correct distortion. Here, however, the proposed method introduces an approach that does not make any assumptions about the projection model of the lens and its parameters and thus allows nearly arbitrary distortions. This calibration procedure employs a special 360° pattern including different markers and fractal shapes in different resolutions which applicable in particular to multi-camera setups. For such multi-camera setups, not only a calibration is needed for each camera, but the individual images have to be aligned to fit together in a final image mosaic. Image alignment is usually considered a separate problem from distortion correction and thus handled separately. The proposed approach, however, combines distortion correction and image alignment in a single procedure.

This new approach employs graph pyramids to perform this registration problem and entails several advantages. First, it does in one procedure both distortion correction and image alignment using a specific 360° pattern. Second, it includes highly parallelized processes meant that individual images are prepared by the cameras themselves and the computational cost at a central processor is reduced. Finally, there is no need to detect key points and solve the correspondence problem in order to do image stitching.

Keywords: Multi-Camera Calibration, Model-free, Graph Pyramid, 360° Pattern, Image Processing.

---

1- Majid Banaeyan is supported by Pattern Recognition and Image Processing (PRIP) group via PRIP-Club.
1. Introduction

Camera calibration establishes a mapping between 3D real-world coordinates of an observed scene and the corresponding 2D coordinates in the acquired image. This mapping may be defined by parameters of a projection model. Generally, we distinguish extrinsic parameters (defining the pose of camera) and intrinsic parameters (depending on the internal nature of the lens and the sensor) [1].

1.1. Camera Models

The pinhole camera model is the most basic camera model. It is used as an approximation within various computer vision applications [2]. In order to improve the calibration accuracy, the pinhole model is combined with lens distortion models [3, 4]. However, for lenses that produce a high visual distortion, e.g. wide-angle lenses with a field of view (FOV) larger than 180°, the pinhole model is not suitable. For instance, cameras equipped with fish-eye lenses with extremely short focal length provide hemispherical images. Their geometry cannot be described based on central perspective projection [5]. Straight lines in the scene appear as curves in images captured using such lenses [3] and require non-linear transformations in addition to the projective geometry [6]. Therefore, alternative projection models such as equidistant, equisolid-angle, orthographic and stereographic projection geometries have been used to compensate this deficiency [7, 4]. The main idea here is to first find an ideal mapping using a cylindrical projection and then apply a subsequent correction. In [8] a similar model based on the spherical projection is created to develop the epipolar geometry for fish-eye lenses and to solve the triangulation problem using the least-square method. In [9] the mapping between pixel coordinates and 3D rays in camera coordinates is considered a linear combination of non-linear functions of the image coordinates. The calibration procedures presented in [10] and [11] assume a radially symmetric distortion to calibrate fish-eye lenses, whereas the procedures presented in [12] and [13] use the knowledge about straight lines in the scene and force them to be straight in the image as well. Further techniques are based on Hough transform [14] and circle fitting [4] or use first order Taylor expansion to derive an inverse function for radial distortion [15]. All parametric methods depend on the accuracy of the chosen projection model and the degree to which this model approximates the actual projection of the lens. For non-standard lenses, deviations from the ideal projection can be large and often lead to unstable approximations [16].

Contrary to the calibration procedures with parametric models, non-parametric calibration methods aim at providing procedures independent of the type of camera or lens. While parametric camera models are state-of-the-art for many applications, non-parametric camera models have been much less discussed in literature. The main idea of these approaches is to assign a line of sight in 3D to every camera pixel [17, 18]. In [18] a set of virtual sensing elements called raxels is used to perform mapping from incoming scene rays to photosensitive elements of the image detector. In [17] a general concept for this type of calibration is described. However, the method does not yield satisfying results for non-central catadioptric cameras and the general algorithm also does not work for perspective cameras. Some authors call it 'generic' calibration [17].

1.2. Multi-Camera Setups
Multi-camera calibration denotes the process of determining the relative poses of all cameras in the setup in a global coordinate system. There are different multi-camera setups which vary in configuration. Generally, there are three categories: inward-looking cameras, outward-looking cameras and camera networks: Inward-looking cameras share a common FOV, for outward-looking cameras the FOVs of the individual cameras do not overlap and in camera networks, some cameras share a FOV, but not all FOVs overlap [19]. Camera pose determination can be performed employing a family of linear methods [20]. Moreover, object silhouettes (Shape-from-Silhouette) may be used to estimate the epipolar geometry of the cameras [21].

For the calibration of a network of cameras, objects are used to first determine the relative transformation between cameras with shared FOV. By utilizing bundle adjustment [22], the globally consistent localization is obtained. For this purpose, many methods are based on the “wand dance procedure” [23, 24] which uses a wand with one, two [25] or more [26] LEDs as a calibration object. Usually, the wand is moved to obtain more point correspondences [25]. For accurate results, the cameras must be precisely synchronized to capture images at the same time [27, 25]. In [27] a laser pointer was used to perform the self-calibration. Direct Linear Transformation (DLT) methods [28] can be used to estimate initial parameters for the first step of calibration [24]. Distributed camera networks may also be calibrated by defining the vision graph and using an algorithm based on belief propagation [29].

Outward-looking cameras do not have a common FOV [30, 31, 32]. However, for cameras with wide angle lenses, an overlap in the FOV may appear, which is the case for the setup is proposed in this paper. In case parts of the FOV overlap, the pose between cameras can be estimated using the rigidity constraint of the rig [30] or the rotation averaging approach [33] which uses bundle adjustment to improve accuracy [22, 31]. Furthermore, a planar mirror may be used to describe the camera pose [32]. Because of these variations in configurations of multi-camera setups, different calibration methods are commonly used for these categories and suitable applications may vary. Inward-looking setups are useful in 3D reconstruction, whereas outward-looking configurations are suitable for image stitching [34, 35] and thus the creation of panoramic images [36].

1.3. Stereo Cameras

Stereo cameras are considered inward-looking setups as well. In stereo vision, the aim is to determine depth information from at least two images with overlapping FOV. For this purpose, corresponding points are determined and a disparity map is computed. Image rectification reduces the correspondence problem from a search in 2D space to a search in 1D along epipolar lines. Rectification needs re-sampling and consequently comes with a loss in resolution. Stereo disparities can be determined based on either local or global constraints [37, 38]. Approaches such as block matching [39], gradient-based optimization [40] and feature matching [41] are considered local methods whereas dynamic programming [11], intrinsic curves [43], and graph cuts [44] represent global methods. Moreover, window-based approaches [38] may use fixed [45] or adaptive [46] search windows.

Contrary to the calibration methods for multi-camera setups described above, the paper here proposes a camera network calibration approach for which correspondence problems are not needed to be solved. Instead, it establishes a mapping to a global target coordinate system using image segmentation. final panorama image is handled by extrapolating a weighted warp from overlap regions of the image to non-overlap regions and by applying a constrained relaxation step of the full panorama frame to a reference projection.
2. Model-free Single Camera Calibration

In our proposed calibration method, the target coordinate system is defined by the calibration pattern used. This method therefore allows for many different target coordinate systems including non-standard ones, since their patches may be arranged in an arbitrary order as long as a certain coordinate may still be associated with every region (patch) of the calibration pattern. At the current stage a rectilinear coordinate system configured cylindrically around a multicamera setup is considered, but a spherical configuration may as well be an option.

The calibration procedure in the paper is a mapping between the coordinate system of the input image and the target coordinate system of an undistorted (calibrated) image. The idea is to employ image segmentation to obtain this mapping: In a graph pyramid [47], each pixel in the base level is represented by a vertex and the vertices of neighboring pixels are connected by an edge. By consecutive contraction of edges pixels with similar values are merged into regions which are represented by vertices on a higher level - each level of such a pyramid is represented by a region adjacency graph (RAG) of decreasing detail (see Figure 1).

![Fig1. Construction of an image pyramid](image)

For our purpose, the aim is to segment the image of the calibration pattern (e.g. a checkerboard pattern) using a graph-pyramid based approach until reaches the level where for each patch of the pattern, its pixels are merged into a single region represented by a vertex: this level is called the “RAG level”. The identification of corresponding regions in the input and the target pattern is straightforward since the number of patches is the same and can thus be measured by counting in order to define the mapping at the respective level of the graph pyramid. To obtain a transformation on pixel-level, the high-level information is propagated down to the base level of the pyramid using the segmentation history that is stored in the canonical pyramid representation [22]. The rest of the steps is explained in the next section.

3. Model-free multi-camera calibration

The proposed calibration algorithm works for arbitrary arrangement of the cameras which the union of their field of views (FOVs) covers the 360º space around them. For the sake of simplicity and better explaining the proposed method, a specific arrangement of six cameras locating around a circle with equal angular distance between each one (60º) is considered. The Fig.2 shows such multi-camera system. In such an arrangement, the FOV for each camera is 180º and each part of the pattern can be seen by at least two adjacent cameras. However, there are some parts of the pattern which are seen by three cameras. These overlapping areas are illustrated in Fig.2.
3.1. Correspond the image to the real world

In contrast to the common methods for calibrating a camera, here no specific model for the lens of the camera is assumed. Indeed, the distortion of the lens is assumed to be an unknown non-linear function. At this point, the task of calibration is to correspond every individual pixel of the 2D image to its correspondence in the 3D real world. Usually very wide FOV lenses result in a high distortion on the side of the image and less in the center. As a result, the objects aligned in the side of the camera become smaller and highly deformed and those objects around the center are highly magnified. Fig. 3 shows such distortion in a fish eye lens.

Having said that, no matter what is the distortion of the lens, the light beams traverse a unique line from the outer lens into the 3D world. In other words, there is a unique point for each pixel of the
image plane projected to the corresponding point in space (see Fig4.). Therefore, the task of calibration is to define such correspondence matching. Generally, in camera calibration systems a known pattern, e.g. a black and white checkerboard, is shown to the camera and then the camera identifies the matching of some robust features (usually corners) for each corresponding feature. Now, having multiple correspondences between two sets of virtual and real points, by assuming a predefined mode for the lens, the calibration is performed.

Fig4. The light beams traversing straight lines from the outer lens into the 3D space

Generally, there are two types of camera models for the lenses. The standard models which assume pinhole projection and the non-standard ones that consider the radial distortion. Normal cameras belong to former and the fish-eye ones - to latter. Now, consider, for example, the fish-eye cameras. By assuming the predefined radial distortion model, after matching the corresponding points, all the rest points employ the same formula to be projected. Therefore, in the cases that there are non-symmetrical distortions, like unwanted movement of lenses, or having non-symmetrical distortions, the non-corresponding points cannot correctly be matched and the calibration fails.

In proposed calibration algorithm, we do not assume any predefined model and instead the correspondence matching task is performed by using the specific 360º calibration pattern. Fig.5 shows the 360º black and white pattern. The proposed pattern encodes uniquely the 3D space around the set of the cameras. It includes specific markers which not only define the origin of the pattern, but also assign the relative coordinate into each individual patch of the checkerboard. In fact, each camera when sees a part of the pattern in its FOV, can immediately define where its location in relative to all other patches is.
Beside the markers, there are some patches which have special fractal pattern in their inside. These patches play the refinement procedure of calibration as are explained later.

3.2. Identifying the corresponding part of the pattern in each individual camera

Based on the arrangement of the cameras in a multi-camera system, the overlapping areas is defined. Here, since the cameras are located in a symmetrical constellation and since they have the same resolution, they divide the image plane into 6 rectangles which all together create the 360° image. By using the 360° pattern, the corresponding matching problem between the two adjacent cameras changes into the easily just counting the patches.

3.3. Graph pyramid segmentation algorithm

To detect the correspondence points of each individual camera and the 3D space, the graph segmentation algorithm for the binary pattern is employed. In such segmentation, first the primal graph of the image is created at the first level of the irregular pyramid. Second, by consecutively contracting the contraction kernels step by step the next levels are built. Finally, the region adjacency graph (RAG) at the top of the pyramid is represented. Such a RAG has the property that each of its node is consider as one patch of the checker board [48].

3.4. Finding the correspondences

After segmenting the image each node of the resulted graph on top of the pyramid represents a unique patch of the checker board. Now, since there are special markers in the 360° pattern, one can simply find out the coordinate of each patch by just counting patches in between till reaches...
to the pre-defined origin. The markers and the fractal shapes are illustrated in magnified size in Fig.7.

![Fig7. (a) the fractal shapes in different resolutions. (b) the markers inside the 360° pattern](image)

### 3.5. Refining the calibration

The 360° pattern has special fractal shape patches. The main property of the fractals is that they generate the same shape in different scales or different positions. Therefore, in the pattern as it sees in Fig. two fractals in two different scales are designed. Every normal patch in such pattern has uniform region inside its patch. This means that the corresponding points of the resulted graph for each normal patch has 4 points from the world. Hence, the points between the corners have not any correspondence. Now, at this point, by rotating the pattern around the multi-camera system each time there is a new pattern in the FOV of the cameras. This means that whenever the position of the high resolution (fractal) patches are changed, a new part of the space get a high number of corresponding points. Therefore, after rotating the pattern in a few times, the space receives the corresponding matching points as the high resolution in the fractal patch. Note that, if the pattern is fixed one can use the 360° turn-table to put the camera in center and rotate the multi-camera system instead of the pattern.

### 4. Discussion and implementation.

Following an important problem exists in camera calibration explains and the solution is discussed. Then the next part the results are summerised.

#### 4.1. Solving the problem of corresponding by using DT

One of the common problem of the camera calibration is to define exactly where the location of a corner is. As it mentioned before, in a very wide FOV, the patches that are located in the side of the image receive much more distortion (see Fig.8). As the result, a single corner which is between two diagonal patches becomes two distinguish points. Therefore, the calibration may fail or becomes less accurate. In the proposed algorithm this multi-corners cause that the two adjacent one-color patches connected to each other and hence are considered as a one patch. Therefore, to avoid such situation, the algorithm computes the distance transform (DT) inside each individual patch of the pattern. This results in detecting the center of the patches instead of the corners. By using DT every patch has its local extremum which is uniquely identified its patch (Fig.9).
4.2. Implementation

The 360° cylindrical pattern is shown in Fig. 10. It covers the area around the camera rig and there are 6 cameras in the arrangement. The diameter of the pattern in cylindrical situation is 100 cm. Each fish-eye lens has 200° FOV and the resolution of each camera is 7800 by 3600 pixels. Size of the square patch in the pattern (read 3D world) is equal to 9 cm. The standard and little-planet view of the resulted stitched 360° image are shown in Fig. 11 and Fig. 12, respectively.
Fig10. The 6 fish-eye frames corresponding to 6 cameras in the multi-camera system.

Fig11. The stitched 360° image in standard view
5. Conclusion

The paper proposes a new multi-camera calibration based on the graph pyramid and the novel 360° pattern. It does not assume any geometrical model for the lens distortion of the lenses and therefore can be applied for almost any arbitrary distortion. Moreover, the specific design of the employed pattern enables it to uniquely encode the 3D world with high resolution depending on what resolution the camera set requires. The method also uses distance transform shift the corners into the center of patches and therefore solves the problem of multi-corner detection happens along the side of the highly distorted image. The future work is to design a colorful version of the pattern and to simulate the non-symmetrical arrangement of cameras.

References


