Automatic Interactive Calibration of Multi-Projector-Camera Systems

Andreas Griesser[1]
[1]Computer Vision Laboratory
ETH Zurich, Switzerland
{griesser,vangool}@vision.ee.ethz.ch

Luc Van Gool[1,2]
[2]PSI/VISICS
Katholieke Universiteit Leuven, Belgium
luc.vangool@esat.kuleuven.ac.be

Abstract

This paper presents a novel interactive Projector Calibration for arbitrary Multi-Projector-Camera environments. The method does not require any calibration rig and is not restricted to any special arrangement of the display devices. The cameras in the system need to be precalibrated so that a common world coordinate system can be defined. Each projector is sequentially activated and a series of self-identifying tags is generated. These tags are accurately and robustly detected and must be seen by a minimum subset of the cameras. This is achieved by freely moving a target object, i.e. a sheet of paper, onto which the tags are projected. Point correspondences for each tag in three-space are computed by minimizing the reprojection errors in the relevant images and eliminating potential outliers. If a tag has been seen a sufficient number of times, it is masked out in the display so that a visual feedback of the current detection state is given to the user. The resulting 3D-point cloud coupled with the known 2D tag-position in the projector frame serves as input for a nonlinear optimization of the projector's intrinsic and extrinsic parameters as well as distortion factors. We show that the overall procedure takes less than a minute and results in low reprojection errors.

1. Introduction

Multi-projector-camera systems have gained an increase of interest over the past years and stimulated the development of a variety of applications such as immersive environments [3, 11], structured light scanning systems [6, 16], and multi-projector display-walls [12, 10, 13]. While multi-camera calibration has been extensively studied, adequate calibration of active illumination devices for general purpose is still outstanding. Although some effort has been put on calibration of display-walls, it is generally assumed that the illuminated surface is at least piecewise planar [7, 8].

Unlike prior work, we do not just recover the homography between projector and camera or between several projectors, instead we estimate the full projection matrices together with distortion factors. This is achieved by applying conventional calibration techniques to a set of 3D-2D-correspondences. By projecting a series of self-identifying bi-tonal markers and robustly detecting them in the camera images, we can estimate their positions in three-space. Our method is inspired by the work of Fiala [5, 4], who used the 2D-marker position in a dedicated camera to compute homographies between several display devices in order to produce a large-scale display. Our work is an interactive 3D-extension, providing a visual feedback to the user of the current detection state. Furthermore we eliminate the necessity of the tag being placed on a piecewise planar surface. Hence, the user can freely move a calibration target, *e.g.* a sheet of paper or similar, onto which the tag is projected. The 3D-point is validated by minimizing the reprojection error in the relative camera images. Interaction with the user is given by masking off those tags which have already been detected in a sufficient number of frames.

The remainder of this paper is organized as follows: Section 2 describes the system architecture. The creation and detection of the tags is explained in section 3. Section 4 is devoted to the computation of correspondences and section 5 briefly describes the method of user-interaction by updating the displayed tags. The calibration procedure and the final validation is the focus in section 6. Experimental results are given in section 7 and section 8 concludes the paper.

2. System Architecture

The setup on which we have tested our calibration method comprises 8 modules, each consists of one camera and one projector connected to a computer, placed in a half-circle around the working volume, see figure 1. The mean distance to the working volume's center is 1.5m, whereby the baseline between the cameras is approx. 0.7m. The modules communicate over a Gigabit Ethernet network with a dedicated master computer, who controls system behaviour by triggering the cameras and performing the calibration procedure.

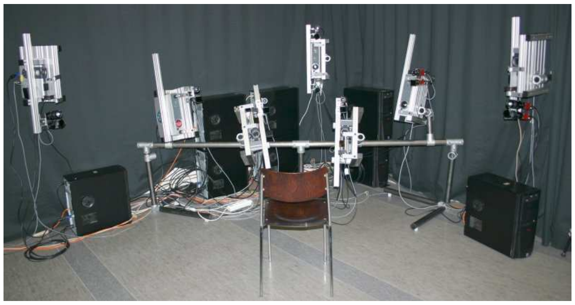

Figure 1. 3D-scanning setup for human face reconstruction: several cameras and projectors are positioned around the object of interest in a distance of approx. 1.5m.

Since the cameras need to be calibrated in order to perform our proposed projector calibration method, we use the method of Svoboda [14]: a small light source, such as a LED or a laser-pointer, which can be seen by at least 3 cameras, is swept through the observed volume. The cameras are synchronized and thus point correspondences in the images can be used to compute full projection matrices w.r.t. a common World Coordinate System. Reprojection errors of less than 0.4 pixels can be obtained by this calibration method.

In figure 2 the processing pipeline for projector calibration is shown. We start with the activation of one specific projector in the setup, connected to one of the modules. A series of tags is generated and projected onto the scene. The location of each displayed tag's image coordinates is sent to the master computer and is further used as 2D-correspondence. Now we trigger all cameras, which initiates the decoding of tags. The resulting lists of correctly detected tags together with their 2D-coordinates in the camera images are again sent to the master. After all modules finished processing, we compute the 3D-points for each tag and, if they pass the validation step, a tag-counter is increased. A tag is no longer projected and thus labelled as *observed* if its counter exceeds a given threshold. When enough tags are observed, the calibration based on 3D-2D-correspondences takes place. A final epipolar validation verifies the resulting projector parameters before switching to the next projector.

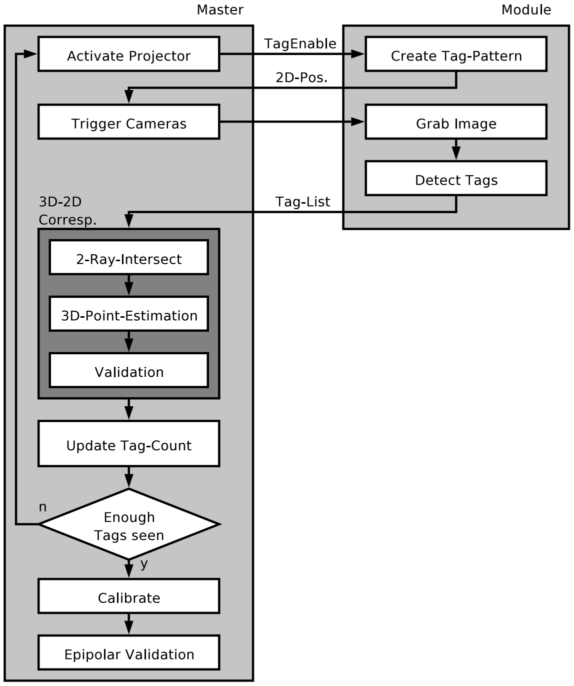

Figure 2. Processing Pipeline for one projector.

3. Tag Creation and Detection

In this section we give a brief overview of the creation and detection of the projected bi-tonal markers. The basic prinicple is quite similar to the system presented by Fiala [4], which has been shown to be very robust against inter-marker confusion, lighting changes and false detections. However, in our application the focus is primarily on the accurate recognition of marker-positions in the image instead of just finding the correct tag-ID. Moreover, we eliminate the requirement of a tag to be placed on a planar surface and allow for slightly bended targets, such as a shirt on human body or a single sheet of paper. Thus, we extended the tag-detection and provide an accuracy-qualifier, coupled with the possibly bit-corrected tag-ID.

In figure 3 the structure of such projected tag is shown. The current maximum number of tags is limited to 256 but can be increased by adapting the code-lengths. The tag-ID is encoded in the 8 data bits $d_{0...7}$ and together with 4 parity bits $p_{0...3}$ a 12-bit codeword is formed. By using a 12-8 Hamming code we can correct one single bit in case of an error or detect 2 false bits alternatively.

In order to account for rotation-invariance, the 4 corner bits are set to $r_{0...3} = [1, 0, 0, 0]$, whereby 0 indicates black and 1 indicates white. Indeed, the codeword needs to be rotated before decoding the ID.

Tag-Recognition starts with applying an adaptive threshold to the input greyscale image. This method is more robust against illumination changes than a simple static thresholding. This follows a detection of closed contours, *e.g.* the white border around each tag. The further processing is similar to ARToolkit [2]: finding potential edges and eliminate false contours. The result is a list of 4 corner points for each detected tag. Now a homography between the tag-coordinate system and the image plane can be computed, which provides image coordinates for each bit-cell of the tag.

Now we find the intensity values of each bit by summing

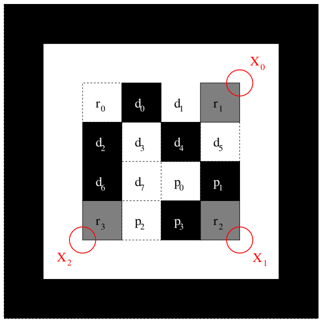

Figure 3. Structure of the projected tags: the 16 inner bit-cells together with a surrounding white border form one tag placed on black background. The 8 data bits $d_{0\dots7}$ and the 4 parity bits $p_{0\dots3}$ can be error-corrected. The 4 corner bits $r_{0\dots3}$ uniquely identify the rotation.

the weighted intensities in a 3×3 pixel neighbourhood w.r.t the distance to the center pixel. Based on the assumption that the 16 intensities form a bi-tonal histogram, we iteratively find an optimal threshold t. One can observe that the better the separation into bright pixels (above threshold, bit-value 1) and dark pixels (below threshold, bit-value 0) is, the more accurate a bit-value can be assigned to each tag-bit. As an accuracy-qualifier we define the contrast

$$d_{max} = max(I_i) - min(I_i) \tag{1}$$

as distance between the highest intensity value and the lowest. Further, the minimum distance between the dark and the bright region is given by

$$\begin{aligned} d_t &= min(I_{i,1}) - max(I_{i,0}) \qquad (2)\\ &I_{i,0} \leq t < I_{i,1}, \qquad (3)\end{aligned}$$

with I_i being the intensity for the i^{th} tag-bit and $I_{i,0}$,$I_{i,1}$ being the closest value below the threshold and above respectively. The relation between d_t and d_{max} specifies the separability of dark and bright pixels, which is scaled by a user-defined factor k. The final accuracy-qualifier $0 \leq q_a < 1$ for the current tag is expressed by

$$q_a = min\left(1, \frac{d_t}{d_{max}} \cdot k\right) \tag{4}$$

and further used as a trustness parameter for the decoding step. Clearly, we currently decode tags with $w_a > 0.9$, otherwise they are discarded.

In our experiments we observed that a tag can be correctly identified even when the white border is partially occluded. This robustness in tag-detection has a major drawback: assume that the tag is masked on the left side such that the white border is still visible but not fully seen. The corners would be found accurately and the tag's ID could be decoded if it passes the accuracy-test. Obviously, the detected corner points are not correct and can cause errors in the projector-camera-correspondences. In order to avoid such situations we compute 3 inner corners $X_{0\dots2}$ for each tag as shown in figure 3. Notice that only 3 corners can be found since the 4^th one is superimposed by the white rotation bit r_0.

In the projector frame we create a 2D-array of tags placed on black background as shown in figure 4. The size of the tag is chosen such that the tag-size in the camera image does not fall below 12 pixels, which is the minimum size for robust detection.

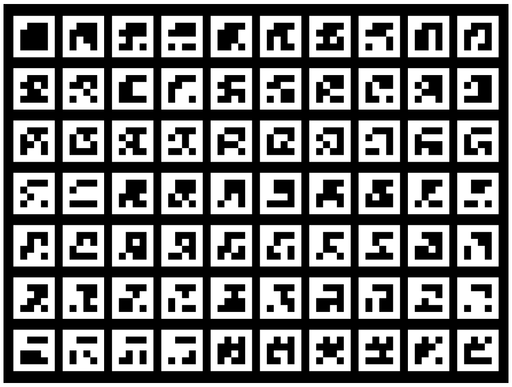

Figure 4. A series of 70 tags is placed on black background in the projector frame.

4. 3D-2D-Correspondences

In a setup with N cameras positioned around the observed space, a considered tag is seen by a subset $n \leq N$ cameras at a given time. For each of the 3 tag-corners $X_{0\dots2}$ the master computer estimates their 3D-positions w.r.t. the world coordinate system defined by the pre-calibrated cameras. In case the 3D-position can be computed accurately, it is, together with the corresponding 2D-location in the projector frame, added to a correspondence list. If the tag is correctly seen a couple of times, it is labeled as *observed* and thus it will no longer be projected. This on-line feedback is a first proof how well the tags can be reconstructed and, if the reprojection errors in the n camera images is below a given threshold, how accurate the 3D-position can be estimated.

The 3D-position computation is split into 3 parts, which will now be explained in detail. Note that the method remains the same for all 3 corner points $X_{0\dots2}$ and thus we will denote p_i as 2D-image coordinates in the i^{th} camera frame.

4.1. 2-Ray Intersection

From the given 2D-points p_i, $i = 1 \dots n$, the rays in 3D can be computed by inverse projection. One might estimate the common 3D-point by minimizing the mean backprojection error in the n images. However, potential outliers may result in divergence and thus we try to eliminate these outliers even before.

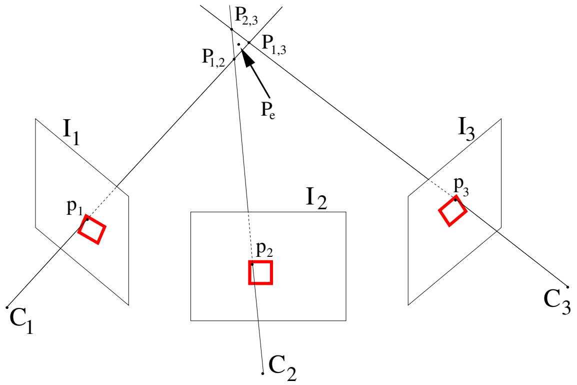

Figure 5. Intersecting the rays from 3 cameras. All 3 intersections are computed and their corresponding mean reprojection error is evaluated. After 3D-point estimation we retrieve P_e.

We first compute all possible $k = \binom{n}{2}$ intersections between pairs of rays in 3D. If the lines do not lie in the same plane, the resulting intersection point is positioned halfway of the shortest distance between both rays. When the mean reprojection error of this intersection point exceeds a conservatively chosen threshold both affected cameras are removed from the further processing.

Figure 5 demonstrates the above procedure for $n = 3$ cameras. The intersection between the first and the second camera is $P_{1,2}$, w.r.t. image points p_1 and p_2 respectively. The mean reprojection error for both cameras 1 and 2 is evaluated and if it exceeds a threshold, only camera 3 remains left.

4.2. 3D-Point Estimation

In order to guarantee convergence for the 3D-point estimation we require at least 3 cameras surviving the previous ray-intersection test. Now that we removed early outliers, the 3D-intersection is estimated by minimizing its orthographic reprojection error in all images.

Given the 2D-location p_i of the considered corner point , (R_i, t_i) are the poses of the corresponding view and P_e denotes the wanted 3D-position. For simplicity, we work with normalized image coordinates in homogenous image coordinates $\tilde{p_i}$. The unknown 3D point P_e is computed as the intersection of the viewing rays of its observations. Since these viewing rays intersect only in the error-free case, the squared distances to the rays is minimized:

$$\sum_i |R_i P_e + t_i - \lambda_i \tilde{p_i}|^2 \rightarrow min \tag{5}$$

where λ_i denotes the depth of P_e when observed from view i. This linear structure is often referred to as *object space collinearity*. Notice that in contrast to the classical pose estimation problem, i.e. [9], here we find a solution for the 3D-point P_e and thus minimize the object-space error function w.r.t. the observed image points in i cameras. The unknown λ_i can be eliminated by setting the derivation equal to zero, which results in a linear system.

4.3. Validation

Although the 3D Point Estimation in the above section usually converges well, we experienced in some cases that the method ran away from the optimal solution. Therefore the 3D-point P_e is validated for reprojection error in all addressed camera images. If it exceeds a given threshold, we may alternatively choose the average or median of all k intersections between pairs of viewing rays. Currently we decided to discard the current point and continue with the next corner of the tag. Clearly, if the reprojection error is above the threshold, the point estimation diverged and the 3D-point is not added to the 3D-2D-correspondence list. Experiments showed that final reprojection errors of less than 0.5 pixels can be reached with acceptable imaging conditions.

5. Tag-Update

When all tags have been processed, we increase for each tag a counter if its 3 corner points correspond to valid 3D-points P_e. If the tag hs been seen a sufficient number of times in the recording sequence, it is marked as *observed* and thus will no longer be displayed by the projector. If at least 80% of all tags are observed, the calibration takes place and we can switch to the next projector in the setup.

6. Calibration and Epipolar Validation

Based on the correspondence between known 2D-points in the projector frame and estimated 3D-points, we can run the calibration procedure. Thereby we use a nonlinear optimization as described in [1] and retrieve a mean reprojection error of approx. 1.5 pixels. Alternatively, the noncoplanar calibration method of Tsai [15] may be used.

It is important to mention that the error can be further reduced when re-calibrating the projector and the cameras by applying bundle-adjustment. Indeed, this would also affect the calibration matrices of the cameras and thus change the world coordinate system origin.

In order to validate the calibration result, we compare the projector with each camera by testing the epipolar geometry. As can be seen in figure 6 a line l_1 in the first image (projector), which goes through the epipole, i.e. projection of the camera center C_c into projector frame, appears as line

l'_1 in the second (camera) image going through the corresponding epipole, i.e. projection of the projector center C_p into camera frame. This relation is referred to as *epipolar line homography*.

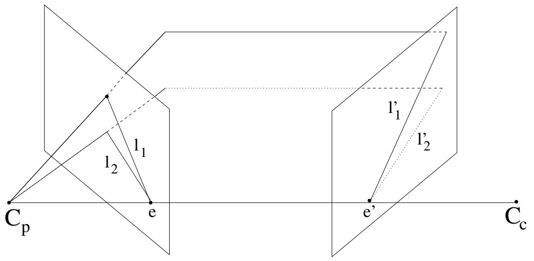

Figure 6. Epipolar Validation based on epipolar line homography: lines through the epipole in the first image appear as lines in the second image through the second epipole, regardless of the objects in the scene.

For validation we create a series of lines in the projector frame w.r.t. a given camera. The grabbed camera image is overlayed with the corresponding lines and the overlapping error is a measure for the accuracy of the calibration.

7. Results

In our experiments we use a 50cm×40cm metal plate on which the tags are projected. The working volume defined by the arrangement of the cameras and projectors is around 1m in diameter. Figure 7 shows a recorded sequence, whereby a total of 70 tags are generated. In order to cover a wide range of the working volume, the user freely moves the plate until the tags disappear. The overall time for the shown sequence is approx. 20 seconds.

In figure 8 the partial tag detection state of 3 selected cameras is shown in 3 consecutive frames. We can observe that the tag with ID=24 (indicated on the bottom right position of the tag), in the first two frames (left column) can be recognized in all 3 cameras. In the third image it can be seen that the tag has been removed from the display list and thus it has been correctly detected in a sufficient number of frames. Notice that in the bottommost camera some tags are not detected because of the minimal edge length of 12 pixels.

After successful recording of the sequence, a total number of 2100 correspondence points remain for calibration. The coverage of the working volume is shown in figure 9, whereby the position of the camera centers w.r.t. the world coordinate origin is indicated by violet lines.

The calibration with nonlinear optimization resulted in an mean reprojection error of less than 1.5 pixels including radial distortion correction. For simplicity we evaluate the epipolar geometry in the distorted case and the result is depicted in figure 10. The top image shows the raw camera image and on the bottom image the transformed lines between projector and camera are superimposed. One can observe a mismatch of less than 2 pixels, which is fairly accurate for our purposes.

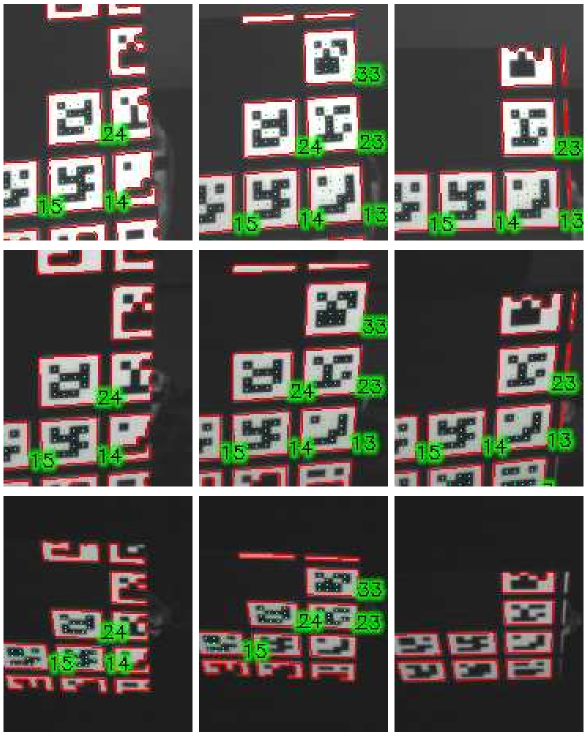

Figure 8. Internal recognition state of 3 cameras (rows) at 3 consecutive timestamps (from left to right).

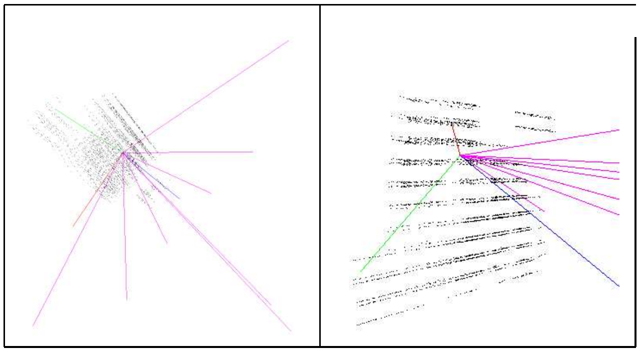

Figure 9. 3D-point cloud with 2100 tag-points detected and validated from 8 cameras. Left: top-view of the setup. Right: detailed view. The on-line recording took approx. 20 seconds.

8. Conclusion

We have proposed an intuitive way of interactively calibrating projectors in a multi-projector-camera environment. By projecting self-identifying bi-tonal tags and robustly decoding them in all camera images, a list of 2D-

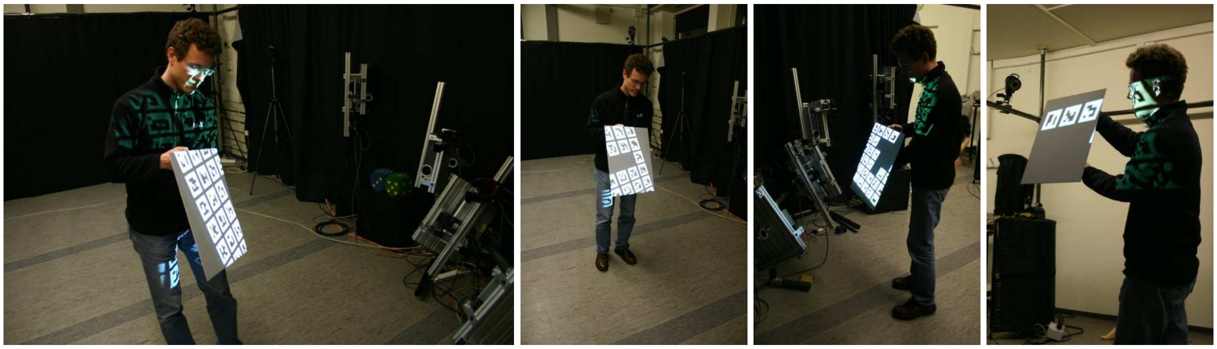

Figure 7. Recorded sequence of 70 tags projected onto a metal plate. After a few seconds the first tags disappear, indicating that the user has to move the plate to a different position until enough tags are recognized for calibration.

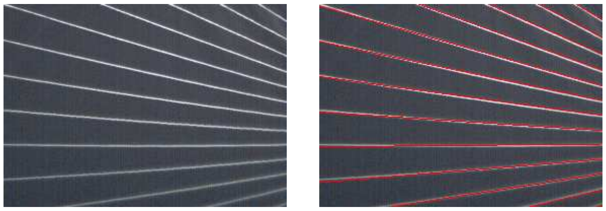

Figure 10. Epipolar validation of the projector calibration for one specific camera. Left: raw camera image. Right: camera image with superimposed epipolar lines corresponding to the projector lines.

correspondences is created. Early outliers are removed by testing the viewing ray intersections for small reprojection errors and estimating a 3D-point for each of the tag's corner points. Tags, which have been correctly detected by a minimal number of cameras in a sufficient number of frames, are further removed from the display list. After the recording state, the 3D-2D correspondences serve as input for a non-linear optimization, computing the intrinsic and extrinsic parameters of the projector. The final validation gives proof of the calibration accuracy. Our method can be applied to any kind of setups where at least 3 pre-calibrated cameras observe the same working volume. Overall recording including calibration is performed in less than a minute. The interactive nature, the fact that no special calibration target is required, and the applicability to any kind of projector-camera environments makes the presented method unique in its own way.

References

[1] B. Triggs et al. Bundle adjustment: A modern synthesis. *Vision Algorithms: Theory and Practice*, 1883, 2000. 4

[2] M. Billinghurst and H. Kato. Collaborative mixed reality. In *Proc. of the First International Symposium on Mixed Reality*, 1999. 2

[3] C. Cruz-Netra, D. Sandlin, and T. DeFanti. Surround-screen projection-based virtual reality: The design and implementation of the cave. In *Proceedings of SIGGRAPH*, 1993. 1

[4] M. Fiala. Artag, a fiducial marker system using digital techniques. In *Proc. of the CVPR-Workshops*, 2005. 1, 2

[5] M. Fiala. Automatic projector calibration using self-identifying patterns. In *Proc. of the CVPR-Workshops*, 2005. 1

[6] A. Griesser, T. P. Koninckx, and L. V. Gool. Adaptive real-time 3d acquisition and contour tracking within a multiple structured light system. In *12th Pacific Conf. on Computer Graphics and Applications*, 2004. 1

[7] J. C. Lee et al. Automatic projector calibration with embedded light sensors. In *UIST '04: Proc. of the 17th annual ACM symp. on User interface software and technology*, 2004. 1

[8] J. M. Rehg et al. Projected light displays using visual feedback. In *Proc. Seventh Intl. Conf. on Control, Automation, Robotics and Vision*, 2002. 1

[9] C. P. Lu, G. D. Hager, and E. Mjolsness. Fast and globally convergent pose estimation from video images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(6), 2000. 4

[10] M. Ashdown et al. A flexible projector-camera system for multi-planar displays. In *Proc. of the CVPR*, 2004. 1

[11] M. Gross et al. blue-c: a spatially immersive display and 3d video portal for telepresence. *ACM Trans. Graph.*, 22(3), 2003. 1

[12] T. Okatani and K. Deguchi. Autocalibration of a projector-camera system. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(12), 2005. 1

[13] R. Raskar et al. Multi-projector displays using camera-based registration. In *VIS '99: Proc. of the Conf. on Visualization*, 1999. 1

[14] T. Svoboda, D. Martinec, and T. Pajdla. A convenient multi-camera self-calibration for virtual environments. *PRESENCE: Teleoperators and Virtual Environments*, 14(4), 2005. 2

[15] R. Tsai. An efficient and accurate camera calibration technique for 3d machine vision. In *Proc. of the CVPR*, 1986. 4

[16] S. Zhang and P. S. Huang. High-resolution, real-time 3-d shape acquisition. In *IEEE Workshop on Real-time 3D Sensors and Their Use (joint with CVPR'04)*, 2004. 1