Faster and more Accurate Feature-based Calibration for Widely Spaced Camera Pairs

Deepak Dwarakanath[1,2], Alexander Eichhorn[1], Carsten Griwodz[1,2], Pål Halvorsen[1,2]
[1]*Simula Research Laboratory,* [2]*University of Oslo, Norway*
{*deepakd, echa, griff, paalh*}*@simula.no*

***Abstract*—The increasing demand for live multimedia systems in gaming, art and entertainment industries, has resulted in the development of multi-view capturing systems that use camera arrays. We investigate sparse (widely spaced) camera arrays to capture scenes of large volume space. A vital aspect of such systems is *camera calibration*, which provides an understanding of the scene geometry used for 3D reconstruction.**

Traditional algorithms make use of a calibration object or identifiable markers placed in the scene, but this is impractical and inconvenient for large spaces. Hence, we take the approach of features-based calibration. Existing schemes based on SIFT (Scale Invariant Feature Transform), exhibit lower accuracy than marker-based schemes due to false positives in feature matching, variations in baseline (spatial displacement between the camera pair) and changes in viewing angle.

Therefore, we propose a new method of SIFT feature based calibration, which adopts a new technique for the detection and removal of wrong SIFT matches and the selection of an optimal subset of matches. Experimental tests show that our proposed algorithm achieves higher accuracy and faster execution for larger baselines of up to ≈2 meters, for an object distance of ≈4.6 meters, and thereby enhances the usability and scalability of multi-camera capturing systems for large spaces.

***Keywords*-Camera arrays; Multiview capture; Feature-based calibration; SIFT; Interactive multimedia for large spaces;**

I. Introduction

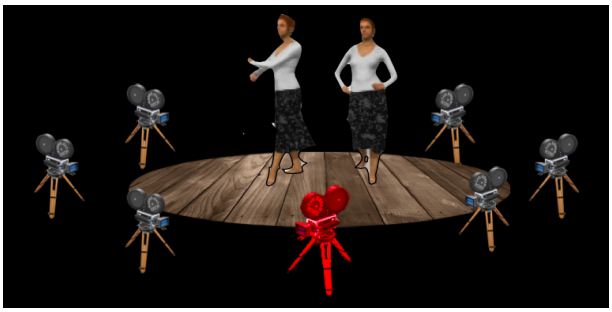
Growing computing performance and the massive parallelization in multi-core processors and specialized graphics hardware have made it possible to process complex computer graphics and computer vision algorithms in real-time. At the same time, camera sensors are becoming cheaper and improve in performance. As a consequence, new kinds of live multimedia systems based on stereoscopic and multi-view video become increasingly attractive for gaming, art and entertainment productions.

Several types of camera arrays are in practical use and development today [1], [2]. They differ in camera density and physical extent. While some image processing techniques such as light-field processing, stereoscopic and multi-view video require relatively dense camera placement, other image processing applications such as free-viewpoint rendering,visual hull reconstruction, tracking or geometrical scene reconstruction can deal with relatively sparse placement.

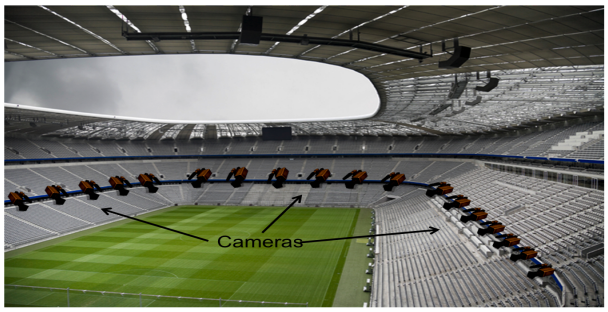
Common to all types of camera arrays is the need for geometric calibration, that is, the identification of intrinsic camera parameters (focal length, principal point and lens distortions) and extrinsic parameters (the geometrical displacement of cameras against each other). Many techniques for the calibration of low-cost camera sensors exist in the computer vision literature, with the most popular ones being methods that use a planar checkerboard pattern [3], [4] or identifiable markers [5]. The calibration accuracy that these methods achieve is sufficient for 3D image processing algorithms, but in many cases, it is inconvenient or impossible to place a measurement target like a checkerboard pattern of sufficient size in front of the cameras.

Calibration based on image feature detection, for example using SIFT [6] (Scale Invariant Feature Transform), has been proposed [7], [8], [9] as an improvement over the traditional, often manual, methods that need a calibration target. Using SIFT, these systems automatically match the features between camera images, which are then used to perform the calibration. However, a particular limitation of SIFT is the decreased feature matching performance with an increase in viewing angle between two perspectives. With a growing *baseline*, the direct distance between any two cameras in an array, less similarities exist between images and consequently, fewer SIFT features are matched. However, the difference may not only manifest in lower overlap or an increased number of occlusions. It may also result in more false positive SIFT matches.

In this work, we extend the prior state-of-art and propose an extrinsic calibration method called $newSIFTcalib$, for pairs of cameras with an arbitrary baseline that works without a calibration target. Our $newSIFTcalib$ is also based on image features obtained using SIFT, but we address some of the limitations of current SIFT-based methods. Specifically, the novelty of the method lies in (a) a new technique for the detection and removal of wrong SIFT matches and (b) a method for selecting a small subset of all detected SIFT features. Our $newSIFTcalib$ particularly compensates for increased viewing angles and large baselines, making SIFT-based calibration usable for camera arrays with large baselines. While the calibration accuracy using SIFT features depends on different factors such as camera baseline and rotation, image resolution, motion blur and external lighting, we focus on the effects of camera baselines and assume that other factors remain constant. We assume further that



(a) Mixed Reality Art Performance Stage



(b) Soccer stadium

Figure 1. Large volume application examples

intrinsic camera parameters are already known or have been determined in a prior calibration step. Based on experimental results, we show that our new method $newSIFTcalib$ can achieve higher calibration accuracy than traditional methods, works with larger baselines than existing calibration schemes and requires less execution time.

In the remainder of the article, we first introduce some example application scenarios where camera baselines are typically large. Section III presents some representative related work. Our new feature-based calibration system is introduced in section IV. Experimental setup and results are described in section V before we conclude the paper in section VI.

II. Applications with large capturing volumes

In several application scenarios, it is necessary to distribute cameras at wide baselines around a large space to capture the entire volume from an optimal number of viewpoints. Examples for such scenarios are:

Mixed Reality On-Stage Performances As in figure 1(a), a camera array is typically placed around the stage. On a remote stage the captured performers are embedded as free-viewpoint video to correct for perspective differences and achieve an aesthetically appealing result.

Sports Events in large arenas such as soccer or baseball games are captured from a large number of perspectives from around the stadium (see figure 1(b)). The video feeds obtained from multiple cameras can be used in various ways such as for silhouette extraction, video mosaicing, motion tracking of players, content analysis.

High accuracy in camera calibration is a prerequisite for high-quality processing of images from cameras at various angles. Accuracy at wide baselines and long shots that are typical in the huge volumes of arenas becomes even more important.

III. Related Work

Previously, similar work on calibration has been carried out using SIFT by, for example, Yun et al. [7], Li et al. [8] and Liu et al. [9]. However, in such algorithms, all the point correspondences obtained by SIFT feature matching have been used for calibration. This is redundant and prone to noise due to mismatches of SIFT features. Eliminating such wrong matches has been studied by Jiayuan et al. [10], using a error canceling algorithm based on RANSAC (Random Sample Consensus - a widely used algorithm for outlier removal). Alternatively, we use a simpler method based on the geometry of lines joining the matched points. Our outlier removal process is faster than and performs as good as RANSAC in our test scenario.

IV. System Description

The system overview is illustrated in figure 2, where a number of stereo camera pairs capture a scene of interest. For every 2D stereo images, we use Vedaldi's library [11] to detect SIFT feature points in stereo images and match them. As a preprocessing step, outliers (false positives in the matching process) are detected and removed. Only a subset of stable points (referred as $FeatureVector$ in rest of the paper), less prone to noise, are used for calibration. We assume the cameras are pre-calibrated for intrinsics.

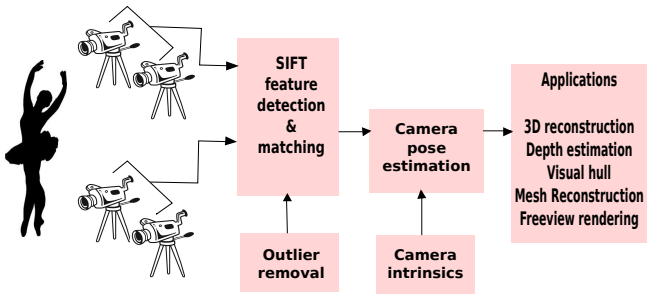


Figure 2. System Overview

A. Outlier detection

This filtering step is based on the angular deviation of the lines connecting corresponding points from, the mean direction of all the lines that connect pairs of corresponding points in two images. Consider two images from stereo

cameras placed horizontally apart from each other. Lines are drawn from every feature point in image 1 to their respective correspondences in image 2, as in figure 4.

We compute the mean (μ_θ^x) and standard deviation (σ_θ^x) of the angle between all lines and the x-axis. Now the outlier detector compares the angle between each line and the x-axis to μ_θ^x and σ_θ^x. A line l_{ij} (and thereby the point correspondence) is identified as an outlier if the angle θ_l^x differs by more than σ_θ^x, as in equation (1). The same is done for the Y-axis. In this way, we make sure that this algorithm can be used on images taken from both horizontally and vertically aligned cameras.

$$outlier = \begin{cases} l_{ij} & \text{if} \quad |\theta_l^{x/y}| > \mu_\theta^{x/y} + \sigma_\theta^{x/y} \\ 0 & \text{if} \quad |\theta_l^{x/y}| < \mu_\theta^{x/y} + \sigma_\theta^{x/y} \end{cases} \tag{1}$$

B. FeatureVector - size and selection

The feature points detected by SIFT are assigned a scale which can be interpreted as a representation of the stability of the feature detection. We exploit this property and sort the inlier point correspondences and define a $FeatureVector$, a vector consisting of point correspondences used for estimating camera pose. Tests in section V-B1 show that the dimension of $FeatureVector$ is chosen to be 25, which is the minimum number of feature points required to achieve a quality similar to the RANSAC algorithm. Next, from the pool of inlier point correspondences, five candidates of subsets from highest order of stability are chosen. Out of these five candidates, the best subset is chosen as the $FeatureVector$, based on least re-projection error, computed for the estimated camera pose.

C. Camera Pose Estimation

The $FeatureVector$ of point correspondences is used to estimate the essential matrix E using normalized 8-point algorithm [12]. Ina stereo camera setup, if the world coordinates are considered to be at the center of the reference camera, the rotation matrix of reference camera is an identity matrix and translation is a zero matrix. Relative rotation R and translation t of the second camera of the camera pair represents the camera pose, and are related to essential matrix as $E = [t]_X R$, where $[t]_X$ is a skew-symmetric matrix,

$$[t]_X = \begin{bmatrix} 0 & t_x & -t_z \\ -t_x & 0 & t_y \\ t_z & -t_y & 0 \end{bmatrix}$$

The Essential matrix can be decomposed using SVD (Singular Value Decomposition) as in [13], which is detailed as follows:

Let K_1 and K_2 be the intrinsics of the camera pair respectively. Upon SVD of E, we obtain:

$$E = USV^T \tag{2}$$

where U and V are unitary matrices and S is a rectangular diagonal matrix. Accordingly, R has two solutions R_a, R_b, and t has two solution t_a, t_b, which are given by

$$R_a = UWV^T, R_b = UW^TV^T, t_a = +u3, t_b = -u3, \tag{3}$$

where u3 is the 3rd column of matrix U and W is as follows:

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

This gives a choice of four solutions to obtain the camera pose. A projection matrix of the reference camera is given as $P_1 = K_1[I|0]$. If $P_2 = K_2[R|t]$ is the projection matrix of the camera, then solution is one of the following:

$$P_2 = K_2[R_a|t_a] \ , \ K_2[R_a|t_b] \ , \ K_2[R_b|t_a] \ , \ K_2[R_b|t_b]$$

The above four solutions have a geometrical meaning and one of the solution is always meaningful. For every possible solution of P_2, 3D points corresponding to the intersection of back projected ray from 2D point correspondences are estimated through triangulation. Using cheirality constraint [14], the 3D points obtained are checked for positive sign of depth and hence the solution for camera pose is determined.

V. Experimentation

A. Dataset

We used widely accepted multi view image dataset by Microsoft Research Laboratory [15] to test our algorithm against others. The dataset was produced using a setup as illustrated in figure 3. All 8 cameras (separated by $\approx$ 0.3 meters distance) captured an event (taken place at $\approx$ 4.6 meters) with a resolution of 1024x728, and rate of 15fps. The calibration parameters for these cameras were computed using traditional approach (checkerboard). These known calibration parameters are used for comparing parameters estimated using other algorithm.

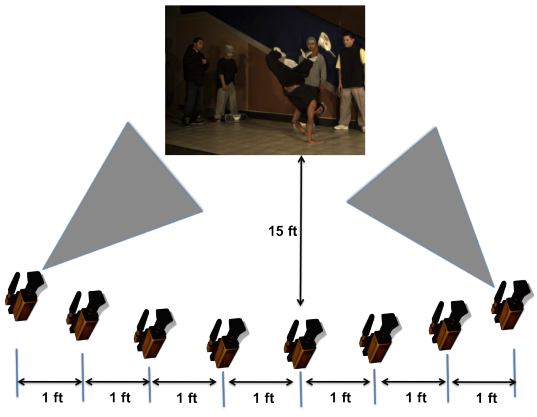


Figure 3. Illustration of setup used by Microsoft [15] to produce the multi-view dataset

B. Test results

First, we conducted an experiment to evaluate the performance of the outlier removal module, and then to evaluate our *newSIFTcalib* algorithm, which comprises of two main techniques - outlier removal and *FeatureVector* selection.

1) Testing the outlier removal performance: To evaluate the performance of outlier removal, we use a first order approximation to geometric error, referred as $EpipolarErr$ in this paper, as stated in [14] and computed as in equation 4, where F is the fundamental matrix: $F = K_2^{-T} E K_1^{-1}$.

$$E_p = \sum_{i=1}^{N} \left(\frac{x\prime_i F x_i}{(Fx_i)_1^2 + (Fx_i)_2^2 + (F^T x\prime_i)_1^2 + (F^T x\prime_i)_2^2} \right) \quad (4)$$

After outlier (solid lines in figure 4) removal, *Epipolar Err* is computed for following methods (a) 8-pt algorithm without outlier detection (b) 8-pt algorithm with RANSAC (c) 8-pt algorithm with our proposed outlier removal.

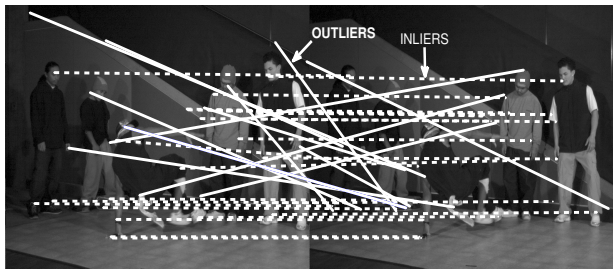


Figure 4. Process of outlier detection: outliers (solid), inliers (dotted)

The test results in fig 5 shows the RANSAC method performs better than 8-point algorithm without outlier removal, as expected. It is very evident that our proposed outlier removal performs as good as RANSAC, and the computation time is drastically reduced because RANSAC requires a large number of points for estimation. From the figure, we can deduce minimum number of points in the *FeatureVector* required for the good performance of outlier removal. Therefore we choose the size of the *FeatureVector* to be 25 points, where our outlier detection performs as good as RANSAC, while reducing the computation time. However, our outlier detector performance is tested only with relative rotation around vertical axis.

2) Testing the proposed algorithm: The performance of our proposed algorithm is compared with other existing ones. The algorithms under study are:

- *Checkerboard* algorithm represents calibration using corners detected on the checkerboard.
- *FullSift_RANSAC* algorithm represents calibration based on SIFT, using all the feature points detected and outliers removed by RANSAC.
- *FullSift* algorithm represents calibration based on SIFT, using all the feature points detected and outliers removed by our proposed method.

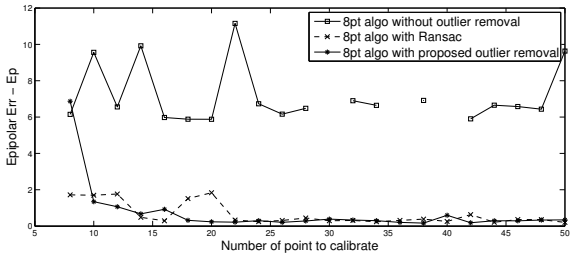


Figure 5. E_p computed for three different methods

- *newSIFTcalib / Proposed* algorithm - represents our algorithm for calibration based on SIFT, using our proposed outlier removal method and selection of stable subset (*FeatureVector*) of feature points.

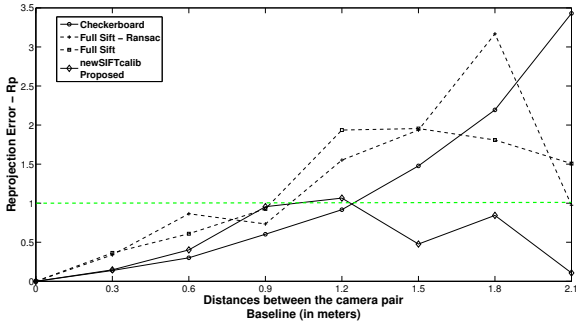


Figure 6. R_p computed for different algorithms

To evaluate the accuracy of calibration, we choose *Reprojection Error* (R_p), measured in pixels, that computes the offset between the estimated image-points using calibration parameters, with that of the measured image-points. Usually, in 3D vision applications, $R_p \leq 1$ is chosen as an acceptable re-projection error.

Given the point correspondences $\{x_1, x_2\}$ and the estimates for projection matrices P1, P2 for two cameras respectively, if we re-project estimated 3D points onto the 2D image plane - referred to as new point correspondences $\{\widetilde{x}_1, \widetilde{x}_2\}$ ($\widetilde{x}_1 = P1\hat{X}$, $\widetilde{x}_2 = P2\hat{X}$) then, re-projection error averaged over N test samples, can be computed as,

$$R_p = \frac{1}{N} \sum_{i=1}^{N} [d(x'_{1i}, \widetilde{x}'_{1i}) + d(x'_{2i}, \widetilde{x}'_{2i})] \quad (5)$$

$$d(x', \widetilde{x}') = ||(x' - \widetilde{x}')||_2 \quad (6)$$

The test result, as shown in figure 6, plots R_p against various baseline distances (in meters) between neighboring cameras. *FullSift_RANSAC* and *FullSift* perform very similarly. This verifies, as in our previous test, that our outlier

removal algorithm used in *FullSift* is as good as RANSAC method for outlier removal while being faster.

At small baselines ($\approx$ 0 - 1.2 meters), the *newSIFTcalib* algorithm performs as good as other algorithms under test, with minimal but acceptable error level of $R_p \leq 1$.

At large baselines ($\approx$ 1.2 - 2.1 meters), our *newSIFTcalib* outperforms *FullSift*, *FullSift_RANSAC* and *Checkerboard* methods. The performance of the other algorithms degrade because of the noise prone feature points, introduced due to large view-angles and baselines. On the other hand, our *newSIFTcalib* algorithm uses the *FeatureVector*, which are more stable and less prone to noise. The *newSIFTcalib* algorithm performs with high consistency at sub-pixel level and is robust to noise.

Alternatively, we compare the estimated camera pose parameters in terms of rotation angles (θ, ϕ, ψ) in 3-dimension, in comparison to the given rotation angles between cameras. Table below shows the parameters known (*Checkerboard*) and parameters estimated (*newSIFTcalib*) for different baseline distances. We can see that the estimated parameters are very close to the given values.

Camera pair	Rotation		
Baseline	θ	ϕ	ψ
0.3 (known)	3.1624	-3.1100	-3.1353
0.3 (estimate)	3.1253	-3.0839	-3.1362
1.2 (known)	3.1547	-3.1015	-3.1271
1.2 (estimate)	3.1278	-2.8736	-3.1355

Now, we evaluate the execution time. The camera pose estimation using different algorithms for cameras separated by 1.2 meters is executed and the elapsed time is measured in seconds. The performance of our *newSIFTcalib* can be reasonably measured relative to other algorithms. Figure 7 shows that our *newSIFTcalib* algorithm achieves 58.82% and 74.07% of percentage decrease in the execution time compared to the *FullSift* and the *FullSift_RANSAC*. One important thing to note is, at 1.2 meters baseline distance, the quality of *newSIFTcalib* is comparable to other algorithms (as in figure 6), while the execution time of *newSIFTcalib* has drastically reduced.

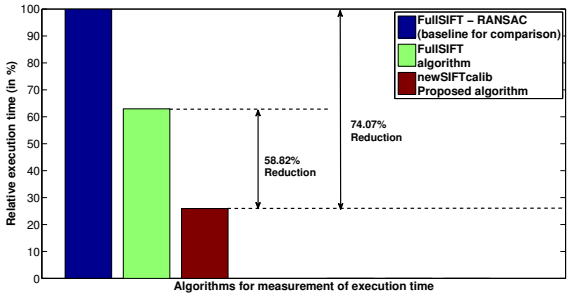


Figure 7. Execution time of various algorithms

Overall, the accuracy of our *newSIFTcalib* algorithm has been consistent at sub-pixel level over multiple baselines, while outperforming the existing algorithms, especially at large baselines. The execution time of our *newSIFTcalib* algorithm has shown a drastic reduction in comparison to other stated algorithms.

C. Operational limits

As a rule of thumb, known to SIFT users, feature detection for cameras, whose view-angle differences are more than 30°, introduces matching errors and thereby degrades the accuracy of calibration system on the whole.

which we can evaluate the performance of the algorithms under study on the operational limits.

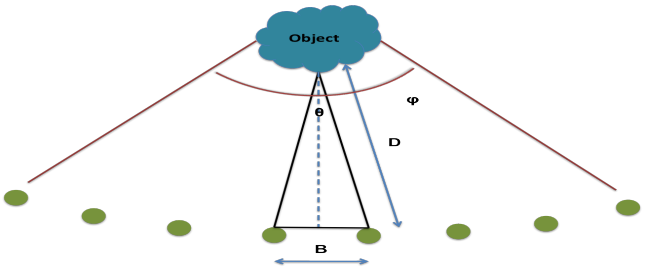


Figure 8. Deduction of relationship between object distance (D) and the baseline distance between the cameras (B)

1) Theoretical limit: Consider figure 8, where D represents the object distance from the camera, B and θ represents the baseline distance and view angle between neighboring cameras. Using triangle equations θ can be expressed as:

$$\theta = 2 * sin^{-1}(\frac{B}{2D}) \tag{7}$$

Using the condition, $\theta \leq 30°$, we have

$$2sin^{-1}(\frac{B}{2D}) \leq 30° \Rightarrow \frac{B}{2D} \leq sin(15°) \Rightarrow B \leq 0.52D$$

The relation $B \leq 0.52D$ is the theoretically defined limit for the baseline using the constraint $\theta \leq 30°$. In our dataset, the object distance is given as 4.6 meters (15 feet), and therefore the theoretically set limit for baseline would then be $\approx$2.4 meters. Let us now check the practical limit for the algorithms on the given dataset.

2) Practical limit: From results as in figure 6, the existing algorithms perform with an acceptable error ($R_p \leq 1$) only up to a baseline separation of $\approx$1 meter. Although the theoretical limit for the baseline is up to 2.4 meters, the existing algorithms practically perform well only up to $\approx$1 meter. Hence we can say that the existing algorithms are well suited for small baselines.

On the other hand, our *newSIFTcalib* algorithm extends the practical limit for the baseline up to 2.1 meters and is well suited for large baselines. The dataset used contains

stereo images separated by a maximum distance of 2.1 meters. Due to this limitation, our *newSIFTcalib* algorithm was not tested for wider baselines, however, it might fail to maintain an acceptable performance. This is merely due to the limitations posed by the SIFT feature detection for variance in view angle.

However, it is evident that our *newSIFTcalib* algorithm pushes the practical limit of the existing algorithms and reaches very close to the theoretical limit.

VI. Conclusion

In this paper, we proposed an algorithm for feature based calibration of camera pairs with application to large volume spaces such as mixed reality performances and soccer event scenarios. Our algorithm uses novel techniques for outlier removal and selection of a lower dimension feature vector consisting of stable, low noise features.

Several tests have shown that our feature based calibration algorithm performs with high consistency and accuracy even at large baselines, compared to existing algorithms. This is definitely an improvement because cameras can be widely spaced, without compromising on the calibration accuracy. Such calibration scheme can be extended to multi camera setup easily.

The execution time of our algorithm was reduced drastically and hence, can be adopted in realtime applications such as gaming, mixed / augmented reality, networked performances and is very useful for structure-from-motion applications.

Overall, our proposed algorithm has shown better performance, which makes it suitable for wide baselines of up to ≈ 2 meters, and thereby enhances the usability and scalability for multi-view capturing system in large spaces. This contribution is the first step in reaching higher accuracies in image-based rendering, especially for large volume spaces.

In our future work, we would like to work with an extensive dataset that will help us study the effects on image resolution, object distance and size, and lighting conditions on the accuracy of feature based calibration. Moreover, it is interesting and important to understand how the accuracy of calibration affects the quality of 3D representation, and thereby, image based rendering schemes.

Acknowledgement

This work is sponsored by the Norwegian Research Council under the Verdione project (project number 187828).

References

[1] B. Wilburn, N. Joshi, V. Vaish, M. Levoy, and M. Horowitz, "High-speed videography using a dense camera array," in *Proceedings of the IEEE computer society conference on Computer Vision and Pattern Recognition*, 2004, pp. 294–301.

[2] C. Zhang and T. Chen, "A self-reconfigurable camera array," in *Proceedings of the ACM SIGGRAPH 2004 Sketches*, 2004, p. 151.

[3] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330 – 1334, November 2000.

[4] R. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323 –344, August 1987.

[5] G. Kurillo, Z. Li, and R. Bajcsy, "Wide-area external multi-camera calibration using vision graphs and virtual calibration object," in *Proceedings of the ACM/IEEE International Conference on Distributed Smart Cameras*, September 2008, pp. 1 –9.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, November 2004.

[7] J. H. Yun and R. H. Park, *Self-Calibration with Two Views Using the Scale-Invariant Feature Transform*. Springer Berlin / Heidelberg, 2006, vol. 4291/2006, pp. 589–598.

[8] C. Li, P. Lu, and L. Ma, "A camera on-line recalibration framework using sift," *The Visual Computer: International Journal of Computer Graphics*, vol. 26, pp. 227–240, March 2010.

[9] R. Liu, H. Zhang, M. Liu, X. Xia, and T. Hu, "Stereo cameras self-calibration based on sift," *International Conference on Measuring Technology and Mechatronics Automation*, vol. 1, pp. 352–355, 2009.

[10] R. Jiayuan, W. Yigang, and D. Yun, "Study on eliminating wrong match pairs of sift," in *IEEE 10th International Conference on Signal Processing*, oct. 2010, pp. 992 –995.

[11] A. Vedaldi and B. Fulkerson, "Vlfeat – an open and portable library of computer vision algorithms," in *Proceedings of the 18th annual ACM international conference on Multimedia*, 2010.

[12] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 580–593, June 1997.

[13] R. I. Hart-ley, "Estimation of relative camera positions for uncalibrated cameras." Springer-Verlag, 1992, pp. 579–587.

[14] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[15] C. Zitnick, S. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM SIGGRAPH and ACM Transactions on Graphics, Los Angeles, CA*, pp. 600–608, August 2004.