



HAL
open science

An Efficient Shortest Triangle Paths Algorithm for Uncertainty-based Multi Camera Calibration

Ferid Bajramovic, Joachim Denzler

► **To cite this version:**

Ferid Bajramovic, Joachim Denzler. An Efficient Shortest Triangle Paths Algorithm for Uncertainty-based Multi Camera Calibration. The 8th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras - OMNIVIS, Rahul Swaminathan and Vincenzo Caglioti and Antonis Argyros, Oct 2008, Marseille, France. inria-00325327

HAL Id: inria-00325327

<https://inria.hal.science/inria-00325327>

Submitted on 28 Sep 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Efficient Shortest Triangle Paths Algorithm for Uncertainty-based Multi Camera Calibration

Ferid Bajramovic and Joachim Denzler

Chair for Computer Vision, Friedrich-Schiller-University Jena
Ernst-Abbe-Platz 2, 07743 Jena, Germany
{bajramov, denzler}@informatik.uni-jena.de

Abstract. Extrinsically calibrating a multi camera system based on scene images only is in general a very difficult problem. One promising approach computes the extrinsic parameters from sufficiently many pairwise relative poses, which can be reasonably estimated even in case of very many incorrectly matched point correspondences. Obviously, the quality of the calibration highly depends on the correctness of the relative pose estimates. As a limited number of relative poses suffices, we propose automatically selecting only the most reliable ones based on an uncertainty measure. Our selection criterion is equivalent to computing a shortest subgraph consisting of shortest triangle paths. The contribution of this paper consists of a mathematically rigorous proof of the correctness of an efficient algorithm for that problem. In experiments on synthetic and real data, we show that our selection algorithm produces greatly improved calibration results, both, in case of varying portions of outliers as well as varying noise.

1 Introduction

Multi-camera systems become increasingly important in computer vision. For many applications, however, the system has to be calibrated, i.e. the intrinsic and relative extrinsic parameters of the cameras have to be determined. Existing methods can be very roughly classified by the type of input or scene knowledge they require: 1. Pattern based: a classical (planar) calibration pattern either has to be visible in all images [1] or the poses of multiple calibration objects have to be known [2]. 2. LED based: some easily detectable, moving single feature, like an LED in a dark room, is recorded over time [3–6]. 3. Self-calibration (in the broader sense): images are taken from an unknown scene, typically with some (unknown) 3D structure and texture [7–9]. Clearly, the third class is most appealing from a practical point of view, as it is the most flexible one.

From a pure multiple view geometry point of view, multiple camera calibration can be interpreted as a structure from motion (sub)problem [10]. In case of self-calibration of multiple physical cameras, however, point correspondences have to be extracted from images in a medium or wide baseline situation, which typically leads to a very high portion of outliers making subsequent calibration very difficult. In this paper, we focus on such a situation. We want to stress that we make no assumptions about the scene except that it has some arbitrary 3D structure and texture. As the relative pose of two cameras can be estimated even in presence of very many outliers [11, 12], we will use known relative poses between some camera pairs as input. The quality of the results, of course,

highly depends on the relative pose estimates, which varies a lot depending on the quality of the extracted point correspondences. Selecting good relative pose estimates can hence greatly improve calibration results. This problem has been briefly mentioned by Chen, David and Slusallek [5]. It has been more recently addressed by Martinec and Pajdla [7, 13]. They weight relative poses by a measure based on the number of inliers found by RANSAC [14] and also on the “importance” of a relative pose. While their measure is plausible, a theoretical justification is missing.

A further method has been proposed by Vergés-Llahí, Moldovan and Wada [8]. They use an uncertainty measure for relative poses, which consists of a residual and a constraint violation term. Note, however, that constraint violations can be avoided by using the five point algorithm [11]. Selection of relative poses is performed by finding shortest triangle paths in a graph connecting cameras according to known relative poses with edge weights set to these uncertainty values.

In earlier work [15], we have adopted their graph-based approach and proposed theoretically sound uncertainty measures on relative poses based on the work of Engels and Nistér [16]. We have provided a theoretical justification for the shortest triangle paths approach [8] and presented an efficient algorithm. The contribution of this paper consists of a mathematically rigorous proof of the correctness of that algorithm.

The paper is structured as follows: Section 2 presents the specific problem treated in this paper and introduces important notation. Section 3 continues with the uncertainty-based relative pose selection algorithm. The proof is presented in Section 4. In section 5, we present experiments on simulated and real data. Conclusions are given in Section 6.

2 Basics

2.1 Relative and Absolute Poses

We assign a number $i \geq 1$ to each camera. The relative pose between two cameras i and j is denoted by $(\mathbf{R}_{ij}, \mathbf{t}_{ij})$. It is defined such that it maps a 3D point \mathbf{p}_i from the i th camera coordinate system to the j th one as follows: $\mathbf{p}_j = \mathbf{R}_{ij}\mathbf{p}_i + \mathbf{t}_{ij}$. The absolute poses (transforming points from world coordinates to k th camera coordinates) are denoted by $(\mathbf{R}_k, \mathbf{t}_k)$. Note that relative poses have a double subscript, while absolute poses have a single one. When estimating relative poses from images only, the scale of \mathbf{t}_{ij} cannot be determined. We emphasize unknown scale by a star: \mathbf{t}_{ij}^* . Relative pose estimation will be treated in section 3.

A multi camera system together with the set of all known relative poses can be represented by the *camera dependency graph* [8] \mathcal{G}_R , which is also known as the *image connectivity graph* [17]: each camera is a vertex and camera i is connected to camera j iff the relative pose $(\mathbf{R}_{ij}, \mathbf{t}_{ij}^*)$ is known. An example is given in Figure 1. Note that edges are directed to distinguish $(\mathbf{R}_{ij}, \mathbf{t}_{ij}^*)$ from $(\mathbf{R}_{ji}, \mathbf{t}_{ji}^*)$. As relative poses can be easily inverted, however, all edges will be treated as bidirectional. We will denote the sets of vertices and edges of any graph \mathcal{G} by $\mathcal{V}(\mathcal{G})$ and $\mathcal{E}(\mathcal{G})$.

2.2 Calibration Task From Relative Poses

We assume known relative poses up to scale $(\mathbf{R}_{ij}, \mathbf{t}_{ij}^*)$ and seek to compute absolute poses $(\mathbf{R}_k, \mathbf{t}_k)$ expressed in an arbitrary (but common) world coordinate system up to

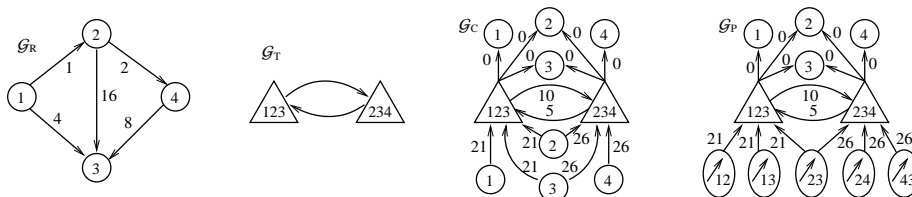


Fig. 1. From left to right: a camera dependency graph \mathcal{G}_R with four cameras and five known relative poses, the associated triangle graph \mathcal{G}_T , the camera-to-camera triangle graph \mathcal{G}_C , and the relative-pose-to-camera triangle graph \mathcal{G}_P .

only one common unknown scale factor. More precisely, calibration is possible up to one 3D similarity transformation per triangle connected component (see Definition 1) in the camera dependency graph [18]. We will assume a triangle connected camera dependency graph. Otherwise, the algorithms can be applied to each component separately.

2.3 Calibration via Triangulation

A *triangle* in \mathcal{G}_R consists of three cameras and three known relative poses. If one of the scales $\|t_{ij}\|$ is known (or fixed), the remaining two scales can be computed by triangulation [10]. By treating one triangle after the other such that all consecutive triangles have a common edge, all scales can be computed up to one common unknown scale parameter. A basic calibration algorithm based on relative poses can be formulated nicely by using the *triangle graph* \mathcal{G}_T , which is defined as follows: each triangle in \mathcal{G}_R becomes a vertex in \mathcal{G}_T , two such vertices are connected in both directions iff the two triangles have a common edge in \mathcal{G}_R . An example is given in Figure 1.

Once the scales have been estimated, the absolute poses can be extracted from the relative poses based on the following two equations: $\mathbf{R}_j = \mathbf{R}_{ij}\mathbf{R}_i$ and $\mathbf{t}_j = \mathbf{R}_{ij}\mathbf{t}_i + t_{ij}$. These steps can be integrated into the following *relative pose calibration algorithm*: First choose a starting triangle $(i, j, k) \in \mathcal{V}(\mathcal{G}_T)$ and set $\|t_{ij}\| = 1$, $\mathbf{R}_i = \mathbf{I}$, and $\mathbf{t}_i = 0$. Then traverse \mathcal{G}_T , e.g. via breadth first search (BFS). When visiting a triangle, estimate the missing scale factors via triangulation and extract the missing absolute poses.

3 Selection of Relative Poses

In the relative pose calibration algorithm, the traversal order is unspecified. Furthermore, in case of a dense camera dependency graph, only a rather small subset of triangles has to be visited in order to calibrate all cameras. Thus, the traversal order corresponds to a selection of triangles and thus relative poses. As the quality of the relative pose estimates varies between camera pairs, a more sophisticated traversal is desirable.

As in previous work [15], we calibrate along *shortest triangle paths* in the camera dependency graph adopting the approach of Vergés-Llahí, Moldovan and Wada [8]. A formal definition of shortest triangle paths follows in Section 3.1. We have suggested three different uncertainty measures on relative poses based on a paper by Engels and

Nistér [16]: the local information, the global entropy and an intermediate measure, which we called smoothed information. Roughly speaking, relative pose are computed using an extension of RANSAC [14] based on the five point algorithm [11]. During the iterations, additional information is collected to compute the uncertainty measures.

We choose one of these measures to assign an uncertainty $w(e)$ to each edge $e \in \mathcal{G}_R$. These values can also be interpreted as edge weights $w : \mathcal{E}(\mathcal{G}_R) \rightarrow [0, \infty[$. Furthermore, we have shown that our measures are well suited for a theoretically sound formulation of relative pose selection as a discrete optimization problem. This provides a theoretical justification for the shortest triangle paths approach.

3.1 Criterion for the Selection of Relative Poses

Calibrating the pose of a camera j relative to a reference camera i involves a sequence of triangles in \mathcal{G}_R such that each two consecutive triangles have a common edge. We call such a sequence a *triangle path from camera i to camera j* which we formally define as: $\mathcal{P} = (i, \mathcal{T}_1, \dots, \mathcal{T}_n, j)$, where $\mathcal{T}_k \subseteq \mathcal{G}_R$ is a triangle for all $1 \leq k \leq n$, $\mathcal{E}(\mathcal{T}_k) \cap \mathcal{E}(\mathcal{T}_{k+1}) \neq \emptyset$ for all $1 \leq k < n$, $i \in \mathcal{V}(\mathcal{T}_1)$, and $j \in \mathcal{V}(\mathcal{T}_n)$. As pointed out before, there are often several alternative triangle paths \mathcal{P} . Based on the uncertainty measures defined above, we assign a total uncertainty to each triangle path \mathcal{P} and choose one with minimum total uncertainty. We model the total uncertainty as

$$w_1(\mathcal{P}) \stackrel{\text{def}}{=} \sum_{(i,j) \in \mathcal{E}_1(\mathcal{P})} w(\mathbf{t}_{i,j}^*), \quad \text{where} \quad \mathcal{E}_1(\mathcal{P}) \stackrel{\text{def}}{=} \bigcup_{k=1}^n \mathcal{E}(\mathcal{T}_k) \quad (1)$$

denotes the set of all edges in \mathcal{P} . For our three uncertainty measures, this model is theoretically sound [15]. The total uncertainty $w_1(\mathcal{P})$ can be interpreted as the *length* of the triangle path \mathcal{P} (see also Definition 3).

Using this approach to calibrate a *whole* multi camera system involves shortest triangle paths from i to all other cameras. Note, however, that the subgraph $\mathcal{S}_i \subseteq \mathcal{G}_R$ induced by all these triangle paths (i.e. the union of all triangles in all paths) has to be triangle connected for reconstructing consistent scales for all \mathbf{t}_{ij}^* .

Definition 1 A subgraph \mathcal{G}'_R of \mathcal{G}_R is called *triangle connected* iff for all cameras $i, j \in \mathcal{V}(\mathcal{G}'_R)$ there is a triangle path from i to j in \mathcal{G}'_R . A triangle connected component of \mathcal{G}_R is a maximal triangle connected subgraph of \mathcal{G}_R .

While the common reference camera only guarantees that \mathcal{S}_i is connected, using a common reference edge $r \in \mathcal{E}(\mathcal{G}_R)$ instead actually guarantees that \mathcal{S}_r is *triangle* connected. Hence, we use shortest triangle paths from a common relative pose to all cameras. We choose the reference relative pose $r \in \mathcal{E}(\mathcal{G}_R)$ with minimum total uncertainty $w(\mathcal{S}_r) = \sum_{e \in \mathcal{E}(\mathcal{S}_r)} w(e)$. In other words, we use the *shortest* triangle connected *shortest* triangle paths subgraph. This extends the shortest triangle paths approach of Vergés-Llahí, Moldovan and Wada [8].

3.2 Computing Shortest Triangle Paths

In this section, we present an efficient algorithm for our selection criterion, which is a shortest triangle paths problem. Note that Vergés-Llahí, Moldovan and Wada [8] men-

tion the need for an efficient shortest triangle paths algorithm, but do not actually present one. This gap is filled by our algorithm [15]. In this paper, we prove its correctness. Note also that the shortest (ordinary) paths algorithm in the related paper by Martinec and Pajdla [13] solves a *different* problem.

Definition 2 *The camera-to-camera triangle graph \mathcal{G}_C is an extension of the triangle graph \mathcal{G}_T . A vertex in \mathcal{G}_T is called triangle vertex. Additionally, there is an entry vertex and an exit vertex for each camera. There is an edge from an entry vertex to a triangle vertex iff the respective camera is in the respective triangle. There is an edge from a triangle vertex to an exit vertex iff the respective camera is in the respective triangle. Edge weights w_C are defined as follows:*

$$w_C(k, l) = \begin{cases} \sum_{e \in \mathcal{E}(\mathcal{T}_i)} w(e) & \text{if } k \text{ is an entry vertex} \\ 0 & \text{if } l \text{ is an exit vertex} \\ \sum_{e \in \mathcal{E}(\mathcal{T}_i) \setminus \mathcal{E}(\mathcal{T}_k)} w(e) & \text{if } k, l \text{ are triangle vertices} \end{cases} \quad (2)$$

Figure 1 shows an example of a camera-to-camera triangle graph. Triangle paths \mathcal{P}_R from a camera i to a camera j in \mathcal{G}_R directly correspond to ordinary paths \mathcal{P}_C in \mathcal{G}_C from an entry vertex to an exit vertex. We call such a path \mathcal{P}_C an *entry-exit path*. Also, shortest entry-exit paths in \mathcal{G}_C directly correspond to triangle paths from i to j with minimum total uncertainty in \mathcal{G}_R . Details and a proof will follow in Section 4. Hence, we can use a standard shortest paths algorithm (e.g. Dijkstra) on \mathcal{G}_C to compute shortest triangle paths in \mathcal{G}_R .

As argued in the previous section, in order to calibrate all cameras, we want to compute shortest triangle paths from a reference relative pose r to all cameras. For that task, we construct a modification of \mathcal{G}_C , which we call *relative-pose-to-camera triangle graph* \mathcal{G}_P : instead of an entry vertex for each camera, there is now an entry vertex for each relative pose and there is an edge from an entry vertex to a triangle vertex iff the respective camera pair is in the respective triangle in \mathcal{G}_R . The edge weights are defined identically as in equation (2). Figure 1 shows an example.

Triangle paths from a relative pose r to a camera i in \mathcal{G}_R directly correspond to entry-exit paths in \mathcal{G}_P . Also, *shortest* entry-exit paths in \mathcal{G}_C directly correspond to *shortest* triangle paths from r to j in \mathcal{G}_R (see Section 4). The shortest paths tree from the entry vertex of a reference relative pose r to all exit vertices in \mathcal{G}_P (*not to all* vertices in the graph!) induces a *triangle* connected subgraph \mathcal{S}_r in \mathcal{G}_R , which consists of shortest triangle paths from r to all cameras. We choose r such that the total uncertainty $w(\mathcal{S}_r) = \sum_{e \in \mathcal{E}(\mathcal{S}_r)} w(e)$ is minimal by applying Dijkstra to all entry vertices r .

4 Proof

In this section, we will prove that shortest triangle paths correspond to shortest paths in \mathcal{G}_C and \mathcal{G}_P respectively. This requires a more graph theoretical point a view. Furthermore, we will be using multisets, i.e. sets which allow for duplicate elements. We will use standard set notation with an M subscript. Example: $\{a, b, b, b, b\}_M \setminus_M \{b, b\}_M =_M \{a, b, b\}_M$. We use the notation $\mathcal{U}(\mathcal{A})$ to denote the *set* containing all elements in a multiset \mathcal{A} , i.e. with duplicate elements removed.

Definition 3 (alternative length of a triangle path) Let $\mathcal{P} = (i, \mathcal{T}_1, \dots, \mathcal{T}_n, j)$ be a triangle path from a camera i to a camera j . The multiset $\mathcal{E}_2(\mathcal{P})$ consists of all edges in all triangles without adding common edges in consecutive triangles twice: $\mathcal{E}_2(\mathcal{P}) \stackrel{\text{def}}{=} \mathcal{E}(\mathcal{T}_1) \sqcup \bigsqcup_{i=2}^n \mathcal{E}(\mathcal{T}_i) \setminus \mathcal{E}(\mathcal{T}_{i-1})$, where \sqcup denotes the multiset union operation. This leads to the alternative definition of the length of a triangle path: $w_2(\mathcal{P}) = \sum_{e \in \mathcal{E}_2(\mathcal{P})} w(e)$.

In equation (1), we have already defined the length $w_1(\mathcal{P})$. An example for the difference between $w_1(\mathcal{P})$ and $w_2(\mathcal{P})$ is shown in Figure 2: The edge e is in $\mathcal{E}_2(\mathcal{P})$ at least once more than in $\mathcal{E}_1(\mathcal{P})$.

Definition 4 (shortest triangle path) Let i and j be two cameras and $k \in \{1, 2\}$. A triangle path \mathcal{P} from i to j is called a w_k shortest triangle path from i to j iff all triangle paths \mathcal{P}' from i to j are at least as long: $w_k(\mathcal{P}) \leq w_k(\mathcal{P}')$.

Lemma 5 Let \mathcal{P}_R be a triangle path from camera i to camera j and let \mathcal{P}_C be the respective entry-exit path in \mathcal{G}_C . Then $w_2(\mathcal{P}_R) = w_C(\mathcal{P}_C)$, where $w_C(\mathcal{P}_C)$ denotes the classical path length in \mathcal{G}_C .

Proof: Both, $w_2(\mathcal{P}_R)$ and $w_C(\mathcal{P}_C)$, are ultimately sums of edge weights in \mathcal{G}_R . By construction, both terms sum over the same multiset of edges. \square

Corollary 6 Shortest entry-exit paths in \mathcal{G}_C bijectively correspond to w_2 shortest triangle paths from a camera i to a camera j in \mathcal{G}_R .

In case of the definition w_1 for the length of a triangle path, it seems to be impossible to construct equally suitable weights in \mathcal{G}_C . Nevertheless, the weights w_C can be used to compute *shortest* triangle paths with respect to w_1 triangle path lengths.

Theorem 7 Let \mathcal{P} be a triangle path from a camera i to a camera j . Let $w(e) > 0$ for all $e \in \mathcal{E}(\mathcal{G}_R)$. \mathcal{P} is a w_1 shortest triangle path iff \mathcal{P} is a w_2 shortest triangle path.

The proof will be given at the end of this section. Due to Corollary 6 and Theorem 7, w_1 shortest triangle paths from a camera i to a camera j can be computed as shortest paths in \mathcal{G}_C . The latter is a standard problem, which can be solved using e.g. the Dijkstra algorithm. The restriction $w(e) > 0$ for all $e \in \mathcal{E}(\mathcal{G}_R)$ poses no problem as in practice the uncertainty will never be zero.

In case of shortest triangle paths from a relative pose r to a camera j , we have the same situation: w_1 shortest triangle paths from a relative pose r to a camera j can be computed as shortest entry-exit paths in \mathcal{G}_P . The same results and proofs apply after substituting “from a camera i ” by “from a relative pose r ”, “ i ” by “ r ”, “camera-to-camera” by “relative-pose-to-camera”, and “ \mathcal{G}_C ” by “ \mathcal{G}_P ”.

Lemma 8 Let \mathcal{P} be a triangle path. Then we have

$$\mathcal{E}_1(\mathcal{P}) \subseteq_M \mathcal{E}_2(\mathcal{P}), \quad \mathcal{E}_1(\mathcal{P}) = \mathcal{U}(\mathcal{E}_2(\mathcal{P})), \quad \text{and} \quad w_1(\mathcal{P}) \leq w_2(\mathcal{P}). \quad (3)$$

Proof: The first two equations follow trivially from Definition 3. As all edge weights in \mathcal{G}_R are non-negative, the third equation follows from the first one. \square

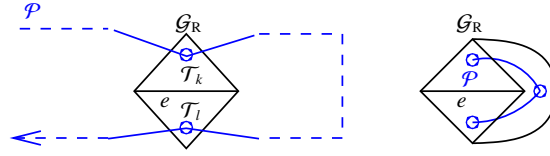


Fig. 2. Second case in the proof of Lemma 9. Left: general case, right: minimal case.

Lemma 9 Let \mathcal{P} be a triangle path from a camera i to a camera j . Let $w(e) > 0$ for all edges $e \in \mathcal{E}(\mathcal{G}_R)$. If $w_2(\mathcal{P}) > w_1(\mathcal{P})$, then there is a triangle path \mathcal{P}' from i to j with $\mathcal{E}_2(\mathcal{P}') \subsetneq_{\mathbb{M}} \mathcal{E}_2(\mathcal{P})$ and $w_2(\mathcal{P}') < w_2(\mathcal{P})$.

Proof: Let \mathcal{P} be a triangle path from a camera i to a camera j with $w_2(\mathcal{P}) > w_1(\mathcal{P})$. As $w_2(\mathcal{P}) > w_1(\mathcal{P})$ and $\mathcal{E}_1(\mathcal{P}) \subseteq_{\mathbb{M}} \mathcal{E}_2(\mathcal{P})$, we have $\mathcal{E}_1(\mathcal{P}) \subsetneq_{\mathbb{M}} \mathcal{E}_2(\mathcal{P})$, i.e. there are duplicate elements in $\mathcal{E}_2(\mathcal{P})$. We will construct a path \mathcal{P}' from \mathcal{P} which removes at least one duplicate element from $\mathcal{E}_2(\mathcal{P})$. Let $\mathcal{P} = (i, \mathcal{T}_1, \dots, \mathcal{T}_n, j)$ with $n \geq 1$. Once we have $\mathcal{E}_2(\mathcal{P}') \subsetneq_{\mathbb{M}} \mathcal{E}_2(\mathcal{P})$, the inequality $w_2(\mathcal{P}') < w_2(\mathcal{P})$ follows directly. There are two cases:

1. A triangle \mathcal{T}^* appears in \mathcal{P} more than once:
There are k, l with $1 \leq k < l \leq n$ such that $\mathcal{T}_k = \mathcal{T}_l = \mathcal{T}^*$. If $l = k + 1$, the l th triangle does not add any edges to $\mathcal{E}_2(\mathcal{P})$ (Definition 3) and can simply be removed. Hence, without loss of generality, we can assume $l \geq k + 2$. Set $\mathcal{P}' = (i, \mathcal{T}_1, \dots, \mathcal{T}_k, \mathcal{T}_{l+1}, \dots, \mathcal{T}_n, j)$. \mathcal{P}' is a triangle path from i to j containing fewer triangles than \mathcal{P} (there is one occurrence of \mathcal{T}^* less in \mathcal{P}' than in \mathcal{P}). As $l \geq k + 2$, this implies $\mathcal{E}_2(\mathcal{P}') \subsetneq_{\mathbb{M}} \mathcal{E}_2(\mathcal{P})$.
2. No triangle appears twice in \mathcal{P} :
In this case, there are two non-consecutive triangles in \mathcal{P} , which have a common edge e . I.e. there are k, l with $1 \leq k < l - 1 \leq n$ such that $e \in \mathcal{E}(\mathcal{T}_k) \cap \mathcal{E}(\mathcal{T}_l)$. Figure 2 visualizes this case. Set $\mathcal{P}' = (i, \mathcal{T}_1, \dots, \mathcal{T}_k, \mathcal{T}_l, \dots, \mathcal{T}_n, j)$. The edge e is in $\mathcal{E}_2(\mathcal{P}')$ at least once less than in $\mathcal{E}_2(\mathcal{P})$. Thus $\mathcal{E}(\mathcal{P}') \subsetneq_{\mathbb{M}} \mathcal{E}(\mathcal{P})$. \square

Lemma 10 Let \mathcal{P} be a triangle path from a camera i to a camera j . Let $w(e) > 0$ for all edges $e \in \mathcal{E}(\mathcal{G}_R)$. If $w_2(\mathcal{P}) > w_1(\mathcal{P})$, then there is a triangle path \mathcal{P}' from i to j with $\mathcal{E}_1(\mathcal{P}') \subsetneq \mathcal{E}_1(\mathcal{P})$ and $w_1(\mathcal{P}') < w_1(\mathcal{P})$.

Proof: Let $\mathcal{P} = (i, \mathcal{T}_1, \dots, \mathcal{T}_n, j)$, $n \geq 1$ be a triangle path from a camera i to a camera j with $w_2(\mathcal{P}) > w_1(\mathcal{P})$. Construct a path \mathcal{P}_1 as in the proof of Lemma 9. \mathcal{P}_1 removes one occurrence of an edge e from $\mathcal{E}_2(\mathcal{P})$. There are two possibilities:

1. There is an edge $e \in \mathcal{E}_2(\mathcal{P}) \setminus_{\mathbb{M}} \mathcal{E}_2(\mathcal{P}_1)$ with $e \notin \mathcal{E}_2(\mathcal{P}_1)$: As $\mathcal{E}_1(\mathcal{P}_1) \subseteq_{\mathbb{M}} \mathcal{E}_2(\mathcal{P}_1)$, we have $e \notin \mathcal{E}_1(\mathcal{P}_1)$ and thus $\mathcal{E}_1(\mathcal{P}_1) \subsetneq \mathcal{E}_1(\mathcal{P})$ and $w_1(\mathcal{P}_1) < w_1(\mathcal{P})$.
2. Otherwise, there are again two possible cases:
 - (a) $\mathcal{E}_1(\mathcal{P}_1) \subsetneq_{\mathbb{M}} \mathcal{E}_2(\mathcal{P}_1)$: Then $w_1(\mathcal{P}_1) < w_2(\mathcal{P}_1)$ and we can recursively apply the construction in this proof to \mathcal{P}_1 instead of \mathcal{P} . This leads to a sequence of triangle paths $(\mathcal{P}, \mathcal{P}_1, \dots)$. Each step removes at least one occurrence of an edge from $\mathcal{E}_2(\mathcal{P} \dots)$. As $\mathcal{E}_2(\mathcal{P})$ is finite, the sequence cannot be infinite and thus has to reach one of the other cases.

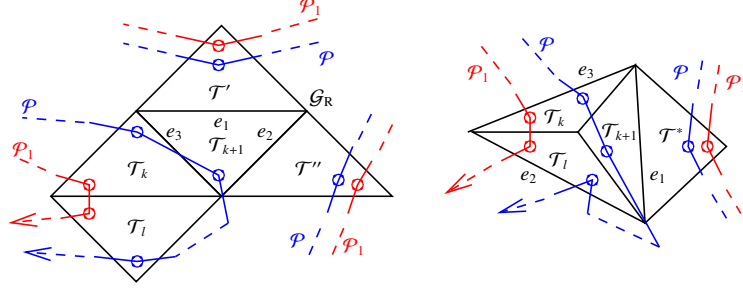


Fig. 3. Left: case 2(b)i in the proof of Lemma 10, right: case 2(b)ii.

(b) $\mathcal{E}_1(\mathcal{P}_1) =_{\text{M}} \mathcal{E}_2(\mathcal{P}_1)$: In this case, we have a closer look at \mathcal{P}_1 . There are k, l with $l \geq k + 2$ such that $\mathcal{P}_1 = (i, \mathcal{T}_1, \dots, \mathcal{T}_k, \mathcal{T}_l, \dots, \mathcal{T}_n, j)$. I.e. at least one triangle between k and l has been removed from \mathcal{P} .

- i. $\mathcal{E}(\mathcal{T}_{k+1}) \cap \mathcal{E}(\mathcal{T}_l) = \emptyset$: This case is depicted in Figure 3, left. Let $\{e_1, e_2, e_3\} = \mathcal{E}(\mathcal{T}_{k+1})$ with $\{e_1, e_2\} = \mathcal{E}(\mathcal{T}_{k+1}) \setminus \mathcal{E}(\mathcal{T}_k)$. As $\mathcal{E}_1(\mathcal{P}_1) =_{\text{M}} \mathcal{E}_2(\mathcal{P}_1)$, we know that $\mathcal{E}_2(\mathcal{P}_1)$ does *not* contain multiple elements. Thus \mathcal{T}_{k+1} cannot be in \mathcal{P}_1 (otherwise e_3 would be in $\mathcal{E}(\mathcal{P}_1)$ twice). Hence there are triangles \mathcal{T}' and \mathcal{T}'' in \mathcal{P}_1 with $\mathcal{T}' \neq \mathcal{T}''$ such that $e_1 \in \mathcal{E}(\mathcal{T}')$ and $e_2 \in \mathcal{E}(\mathcal{T}'')$. We can write $\mathcal{P}_1 = (i, \dots, \hat{\mathcal{T}}_1, \dots, \hat{\mathcal{T}}_2, \dots, \hat{\mathcal{T}}_3, \dots, j)$ with $\{\hat{\mathcal{T}}_1, \hat{\mathcal{T}}_2, \hat{\mathcal{T}}_3\} = \{\mathcal{T}_k, \mathcal{T}', \mathcal{T}''\}$. Now set $\mathcal{P}_2 = (i, \dots, \hat{\mathcal{T}}_1, \mathcal{T}_{k+1}, \hat{\mathcal{T}}_3, \dots, j)$. Then we have: $\mathcal{E}_1(\mathcal{P}_2) \subseteq \mathcal{E}_1(\mathcal{P}_1)$, i.e. there are no new edges in $\mathcal{E}_1(\mathcal{P}_2)$ compared to $\mathcal{E}_1(\mathcal{P}_1)$, as $\mathcal{E}(\mathcal{T}_{k+1}) = \{e_1, e_2, e_3\} \subseteq \mathcal{E}_1(\mathcal{P}_1)$. Furthermore, $(\mathcal{E}(\hat{\mathcal{T}}_2) \setminus \mathcal{E}(\mathcal{T}_{k+1})) \cap \mathcal{E}_1(\mathcal{P}_2) = \emptyset$, i.e. there are two edges less in $\mathcal{E}_1(\mathcal{P}_2)$ than in $\mathcal{E}_1(\mathcal{P}_1)$. Thus we have $\mathcal{E}_1(\mathcal{P}_2) \subsetneq \mathcal{E}_1(\mathcal{P}_1)$ and $w_1(\mathcal{P}_2) < w_1(\mathcal{P}_1) \leq w_1(\mathcal{P})$.
- ii. $\mathcal{E}(\mathcal{T}_{k+1}) \cap \mathcal{E}(\mathcal{T}_l) \neq \emptyset$: This case is depicted in Figure 3, right. Let $\{e_1\} = \mathcal{E}(\mathcal{T}_{k+1}) \setminus \mathcal{E}(\mathcal{T}_k) \setminus \mathcal{E}(\mathcal{T}_l)$. Then $e_1 \in \mathcal{E}_2(\mathcal{P}) \setminus_{\text{M}} \mathcal{E}_2(\mathcal{P}_1)$, i.e. e_1 is in $\mathcal{E}_2(\mathcal{P})$ more often than in $\mathcal{E}(\mathcal{P}_1)$, because \mathcal{T}_{k+1} has been removed from \mathcal{P} to construct \mathcal{P}_1 . Combined with case 2, this implies $e_1 \in \mathcal{E}_2(\mathcal{P}_1)$. As $\mathcal{E}_1(\mathcal{P}_1) = \mathcal{U}(\mathcal{E}_2(\mathcal{P}_1))$ (Lemma 8), this leads to $e_1 \in \mathcal{E}_1(\mathcal{P}_1)$. \mathcal{T}_{k+1} is not in \mathcal{P}_1 . Otherwise each of the two edges in $\mathcal{E}(\mathcal{T}_{k+1}) \setminus \{e_1\}$ would be in $\mathcal{E}_2(\mathcal{P}_1)$ twice, but $\mathcal{E}_2(\mathcal{P}_1)$ does not contain duplicates, as $\mathcal{E}_1(\mathcal{P}_1) =_{\text{M}} \mathcal{E}_2(\mathcal{P}_1)$ (case 2b). There is a triangle $\mathcal{T}^* \neq \mathcal{T}_{k+1}$ with $e_1 \in \mathcal{E}(\mathcal{T}^*)$ in \mathcal{P}_1 , as $e_1 \in \mathcal{E}_1(\mathcal{P}_1)$.

A. \mathcal{T}^* is in \mathcal{P}_1 after \mathcal{T}_k and \mathcal{T}_l , i.e. $\mathcal{P}_1 = (i, \dots, \mathcal{T}_k, \mathcal{T}_l, \dots, \mathcal{T}^*, \dots, j)$: In this case, set $\mathcal{P}_2 = (i, \dots, \mathcal{T}_k, \mathcal{T}_{k+1}, \mathcal{T}^*, \dots, j)$. Then $\mathcal{E}_1(\mathcal{P}_2) \subseteq \mathcal{E}_1(\mathcal{P}_1)$, i.e. there are no additional edges in \mathcal{P}_2 compared to \mathcal{P}_1 . Let $\{e_2\} = \mathcal{E}(\mathcal{T}_l) \setminus \mathcal{E}(\mathcal{T}_k) \setminus \mathcal{E}(\mathcal{T}_{k+1})$. We know that e_2 is in $\mathcal{E}_2(\mathcal{P}_1)$ exactly once (as $\mathcal{E}_1(\mathcal{P}_1) =_{\text{M}} \mathcal{E}_2(\mathcal{P}_1)$) due to case 2b). Thus $e_2 \notin \mathcal{E}_2(\mathcal{P}_2)$ and thus $e_2 \notin \mathcal{E}_1(\mathcal{P}_2)$, as $\mathcal{E}_1(\mathcal{P}_2) \subseteq_{\text{M}} \mathcal{E}_2(\mathcal{P}_2)$ (Lemma 8). We get $\mathcal{E}_1(\mathcal{P}_2) \subsetneq \mathcal{E}_1(\mathcal{P}_1)$ and $w_1(\mathcal{P}_2) < w_1(\mathcal{P}_1) \leq w_1(\mathcal{P})$.

B. \mathcal{T}^* is in \mathcal{P}_1 before \mathcal{T}_k and \mathcal{T}_l , i.e. $\mathcal{P}_1 = (i, \dots, \mathcal{T}^*, \dots, \mathcal{T}_k, \mathcal{T}_l, \dots, j)$: In this case, set $\mathcal{P}_2 = (i, \dots, \mathcal{T}^*, \mathcal{T}_{k+1}, \mathcal{T}_l, \dots, j)$. With an analog argumentation to the previous case (involving an edge e_3 instead of e_2 , Figure 3), we get $\mathcal{E}_1(\mathcal{P}_2) \subsetneq \mathcal{E}_1(\mathcal{P}_1)$ and $w_1(\mathcal{P}_2) < w_1(\mathcal{P}_1) \leq w_1(\mathcal{P})$. \square

The previous lemmas roughly state that if the lengths w_2 and w_1 of a triangle path are not equal, there will be a shorter triangle path with respect to w_2 , as well as a shorter triangle path with respect to w_1 .

Corollary 11 *Let \mathcal{P} be a triangle path from a camera i to a camera j . Let $w(e) > 0$ for all $e \in \mathcal{E}(\mathcal{G}_R)$. If \mathcal{P} is a w_2 or w_1 shortest triangle path, then $w_2(\mathcal{P}) = w_1(\mathcal{P})$.*

Proof: Let \mathcal{P}_C be a w_2 [w_1] shortest triangle path from i to j . Thus, there cannot be a shorter triangle path. Due to Lemma 9 [Lemma 10], this implies $w_2(\mathcal{P}) \leq w_1(\mathcal{P})$. Using Lemma 8, we know that $w_2(\mathcal{P}) \geq w_1(\mathcal{P})$. Thus, $w_2(\mathcal{P}) = w_1(\mathcal{P})$. \square

Proof of Theorem 7: First part: Let \mathcal{P} be a w_1 shortest triangle path from i to j . Let \mathcal{P}' be any triangle path from i to j . Using Lemma 8, Definition 4, and Corollary 11 we have: $w_2(\mathcal{P}') \geq w_1(\mathcal{P}') \geq w_1(\mathcal{P}) = w_2(\mathcal{P})$. Thus \mathcal{P} is a w_2 shortest path.

Second part: Let \mathcal{P} be a w_2 shortest triangle path from i to j . Let \mathcal{P}' be a w_1 shortest triangle path from i to j . This implies $w_1(\mathcal{P}') \leq w_1(\mathcal{P})$. Using Corollary 11 twice, we get $w_2(\mathcal{P}') = w_1(\mathcal{P}') \leq w_1(\mathcal{P}) = w_2(\mathcal{P})$ and thus $w_2(\mathcal{P}') \leq w_2(\mathcal{P})$. As \mathcal{P} is a w_2 shortest triangle path, we also have $w_2(\mathcal{P}) \leq w_2(\mathcal{P}')$. Combining these inequalities, we get $w_2(\mathcal{P}) = w_2(\mathcal{P}')$ and thus $w_1(\mathcal{P}') = w_1(\mathcal{P})$. As \mathcal{P}' is a w_1 shortest path, \mathcal{P} is also a w_1 shortest path. \square

5 Experiments

As a multi camera system can only be calibrated up to a 3D similarity transformation, we compute and apply an optimal transformation between the calibration result and the ground truth using a linear algorithm followed by nonlinear refinement. As an error measure for the calibration of a multi camera system in comparison to the ground truth, we use the mean position error $e = \frac{1}{n} \sum_i \|\mathbf{R}_i \mathbf{t}_i - \mathbf{R}_{G,i} \mathbf{t}_{G,i}\|_2$, where the subscript G indicates ground truth. The ground truth calibration is normalized such that the distance of the first two cameras is one. This defines the scale of the error measure e .

5.1 Simulation

The simulation consists of ten (instead of six as in previous work [15]) virtual pin-hole cameras $(\mathbf{K}, \mathbf{R}_{G,i}, \mathbf{t}_{G,i})$ with image size 640×480 and calibration matrix $\mathbf{K} = ((1500, 0, 320), (0, 1500, 240), (0, 0, 1))^T$. The scene consists of 100 random 3D points uniformly distributed in a cuboid. The cameras are placed in a circle above the points looking roughly towards the center of the cuboid such that all cameras can see all points. The height above the scene of every second camera is slightly increased to avoid planarity. The 3D points are projected into the cameras with subpixel precision. Noise is simulated by adding random values uniformly distributed in $[-\phi/2, \phi/2]$ to all coordinates. We choose $\phi = 1$ for all experiments. Outliers are simulated by replacing a certain fraction of the point correspondences of each camera pair by points uniformly distributed within the image areas. We perform experiments with the following amounts of outliers: 0%, 30%, 70%, and 80%. All experiments are repeated 50 times and the resulting mean position errors e are plotted in sorted order. This corresponds to computing all error quantiles. A good algorithm should, of course, have a low and flat error curve.

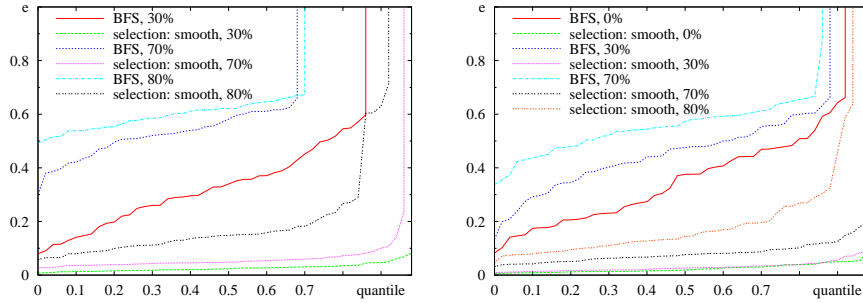


Fig. 4. Comparison of the BFS traversal with our uncertainty based selection of relative poses using the smoothed information measure and the Blake-Zisserman distribution. The percent values denote the portion of outliers. The plot shows sorted mean position error values e (quantiles). Left: first experiment, right: second experiment.



Fig. 5. Setup of the real experiment. Left: the multi camera system observing the pattern for the Zhang calibration, right: the scene (image from the sixth camera).

We perform the following two experiments. In the first one, we increase the portion of outliers of the camera pairs $(i, i + 1)$ for $1 \leq i \leq 9$ such that only half of the inliers remain (e.g. instead of 30%, these camera pairs have 65% outliers). In the second experiment, the noise level of these camera pairs is increased from 1 to 5 without changing the portion of outliers. We compare the BFS traversal to our selection criterion using the Blake-Zisserman distribution and the smoothed information measure [15]. Figure 4 clearly shows that our selection criterion greatly improves the calibration results in both situations. This shows that in both cases the camera pairs with contaminated data can be identified by the smoothed information measure and that the selection mechanism is able to avoid them. Overall, the simulation experiments show that our relative pose selection criterion greatly improves calibration results in case of some camera pairs with additional outliers or additional noise.

5.2 Real Data

The experiment on real data consists of two AVT Marlin monochrome cameras and six AVT Pike color cameras observing a scene, as depicted is Figure 5. We estimate the intrinsic camera parameters using Zhang’s [1] calibration pattern based method. To

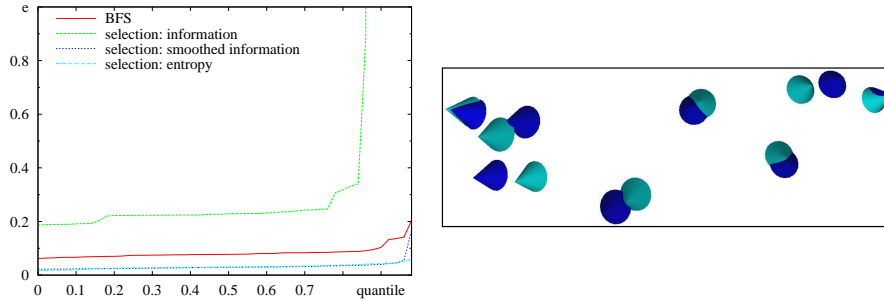


Fig. 6. Experiment on real data. Left: mean position error quantile results for BFS as well as our selection criterion. Right: a visualization of a calibration result (cyan / light gray) with a mean position error $e \approx 0.03$ compared to a Zhang calibration (blue / dark gray). Even though the two calibrations are not identical, the structure of the multi camera system is well reconstructed.

be able to evaluate our calibration results, we use Zhang’s method also to compute a “ground truth” for the extrinsic calibration. Note that this “ground truth” is *not* free of errors, but still provides a reasonable comparison. As input for our calibration, we extract 100 point correspondences in each image pair using SIFT [19].

Figure 6 contains the results for three different uncertainty measures using the Blake-Zisserman distribution [15]. Our selection criterion using the smoothed information measure or the entropy measure provides greatly improved calibration results compared to the BFS traversal. This demonstrates that our method is actually beneficial in practical situations, where the quality of the correspondences between camera pairs cannot be strictly divided into “good” and “bad”.

6 Conclusions

In the context of calibrating a multi camera system via pairwise relative poses, we formulated the selection of relative poses as a graph based discrete optimization problem using an uncertainty measure as edge weights. We presented an efficient algorithm for that problem and a mathematically rigorous proof of its correctness. In the simulation experiments, we showed that our selection criterion is able to greatly improve calibration results by avoiding bad relative poses caused by an increased outlier portion or increased noise. In experiments on real data, we demonstrated that calibration results are actually improved in a practical situation.

References

1. Zhang, Z.: A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **22** (2000) 1330–1334
2. Kitahara, I., Saito, H., Akimichi, S., Ono, T., Ohta, Y., Kanade, T.: Large-scale Virtualized Reality. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Technical Sketches.* (2001)

3. Baker, P., Aloimonos, Y.: Complete Calibration of a Multi-camera Network. In: Proceedings of the IEEE Workshop on Omnidirectional Vision (OMNIVIS). (2000) 134–144
4. Svoboda, T., Martinec, D., Pajdla, T.: A Convenient Multi-Camera Self-Calibration for Virtual Environments. *PRESENCE: Teleoperators and Virtual Environments* **14** (2005) 407–422
5. Chen, X., Davis, J., Slusallek, P.: Wide Area Camera Calibration Using Virtual Calibration Objects. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Volume 2. (2000) 2520–2527
6. Barreto, J.P., Daniilidis, K.: Wide Area Multiple Camera Calibration and Estimation of Radial Distortion. In: Proceedings of the IEEE Workshop on Omnidirectional Vision (OMNIVIS). (2004)
7. Martinec, D., Pajdla, T.: Robust Rotation and Translation Estimation in Multiview Reconstruction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2007) 1–8
8. Vergés-Llahí, J., Moldovan, D., Wada, T.: A new reliability measure for essential matrices suitable in multiple view calibration. In: *VISAPP 2008: Proceedings of the Third International Conference on Computer Vision Theory and Applications*. Volume 1. (2008) 114–121
9. Hörster, E., Lienhart, R.: Calibrating and optimizing poses of visual sensors in distributed platforms. *ACM Multimedia System Journal, Special Issue on Multimedia Surveillance System* **12** (2006) 195–210
10. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. 2nd edn. Cambridge University Press (2003)
11. Stewénius, H., Engels, C., Nistér, D.: Recent Developments on Direct Relative Orientation. *ISPRS Journal of Photogrammetry and Remote Sensing* **60** (2006) 284–294
12. Brückner, M., Bajramovic, F., Denzler, J.: Experimental Evaluation of Relative Pose Estimation Algorithms. In: *VISAPP 2008: Proceedings of the Third International Conference on Computer Vision Theory and Applications*. Volume 2. (2008) 431–438
13. Martinec, D., Pajdla, T.: 3D Reconstruction by Gluing Pair-Wise Euclidean Reconstructions, or "How to Achieve a Good Reconstruction from Bad Images". In: *3DPVT '06: Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*. (2006) 25–32
14. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* **24** (1981) 381–395
15. Bajramovic, F., Denzler, J.: Global Uncertainty-based Selection of Relative Poses for Multi Camera Calibration. In: *Proceedings of the British Machine Vision Conference (BMVC)*. (2008) to appear
16. Engels, C., Nistér, D.: Global uncertainty in epipolar geometry via fully and partially data-driven sampling. In: *ISPRS Workshop BenCOS: Towards Benchmarking Automated Calibration, Orientation and Surface Reconstruction from Images*. (2005) 17–22
17. Snavely, N., Seitz, S., Szeliski, R.: Modeling the World from Internet Photo Collections. *International Journal of Computer Vision (IJCV)* (**to appear**) (2008)
18. Mantzel, W.E., Choi, H., Baraniuk, R.G.: Distributed Camera Network Localization. In: *Proceedings of the 38th Asilomar Conference on Signals, Systems and Computers*. Volume 2. (2004) 1381–1386
19. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision (IJCV)* **60** (2004) 91–110