In Proc. CVPR, pp. II-282 -- II-289, June 2000.

Automatic Recovery of Relative Camera Rotations for Urban Scenes

Matthew E. Antone Seth Teller[†]
Computer Graphics Group
MIT Laboratory for Computer Science
545 Technology Square, Cambridge MA, 02139
{tone,seth}@mit.edu

Abstract

In this paper we describe a formulation of extrinsic camera calibration that decouples rotation from translation by exploiting properties inherent in urban scenes. We then present an algorithm which uses edge features to robustly and accurately estimate relative rotations among multiple cameras given intrinsic calibration and approximate initial pose. The algorithm is linear both in the number of images and the number of features.

We estimate the number and directions of vanishing points (VPs) with respect to each camera using a hybrid approach that combines the robustness of the Hough transform with the accuracy of expectation maximization. Matching and labeling methods identify unique VPs and correspond them across all cameras. Finally, a technique akin to bundle adjustment produces globally optimal estimates of relative camera rotations by bringing all VPs into optimal alignment. Uncertainty is modeled and used at every stage to improve accuracy.

We assess the algorithm's performance on both synthetic and real data, and compare our results to those of semi-automated photogrammetric methods for a large set of real hemispherical images, using several consistency and error metrics.

1 Introduction

The focus of this work is determination of the extrinsic orientations of a large number of cameras over an extended area. This section gives a high-level description of our method and some relevant work on this topic.

1.1 Motivation

The goal of the MIT City Project [15] is fully automated 3-D reconstruction of urban landscapes from terrestrial (ground-level) imagery annotated with approximate intrinsic and extrinsic camera parameters. Data is acquired in sets of *nodes*; a node is a hemispherically-tiled set of images captured from a single position in space, at different rotations about the camera's optical center. Nodes are typically separated by significant (10-meter) baselines and are acquired at different times of day and under different conditions of illumination and weather.

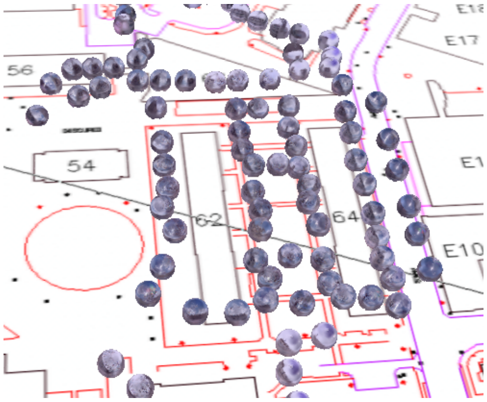

Figure 1: Acquired Data
Part of a data set consisting of 210 hemispherical nodes containing over 4,000 images, each 1.5 million pixels.

An initial mosaic step [3] registers all planar images within a given node to form a hemispherical image, and also estimates intrinsic calibration parameters. Extrinsic (inter-node) camera registration is currently semi-automated, relying on manual point correspondences. Since this task becomes cumbersome as the number of acquired nodes grows, we are developing robust, scalable techniques that determine camera pose without human intervention.

1.2 Method Overview

Here we consider only the rotational component of extrinsic pose. We decouple it from the translational component by inferring 3-D edge directions (vanishing points), which are invariant under camera translation, from 2-D edge observations. Our method operates under several assumptions:

- *Viewed scenes contain sets of parallel lines.* Urban environments typically consist of regular structures such as building facades with repeating windows.
- *Intrinsic camera parameters are known.* These parameters are estimated by a separate algorithm [3].

[†] Support of the Advanced Research Projects Agency (under contract DACA76-97-K-0002), the Office of Naval Research (under MURI award SA 1524-2582386), NTT, and Intel Corp. are gratefully acknowledged.

- *Extrinsic pose is approximately known*. This information is obtained by the acquisition platform and is used both to determine camera adjacency for wide baselines and to resolve rotational ambiguities.
- *Images are omnidirectional.* Our methods can be applied to single rectangular images as well, but hemispherical images increase the accuracy of vanishing point (VP) estimation and facilitate determination of correspondences among VPs.

The method comprises several stages. First, edge features in the 2-D images are obtained to sub-pixel accuracy using an edge detection and point chaining technique. A hybrid approach consisting of robust Hough transform (HT) and accurate expectation maximization (EM) components uses these edges, along with intrinsic camera calibration parameters, to determine the number and 3-D orientations of VPs in the scene. VPs are then matched across cameras, and the correspondences are used to estimate the optimal rotations (represented as quaternions [5, 7]) that register the cameras.

Our method has several advantages:

- *Scalability.* The algorithm is linear both in the number of images and the number of edge observations.
- *Global optimality.* Error propagation and bias are minimized by considering all available data simultaneously.
- *Robustness*. The method handles arbitrarily wide baselines and significant error in initial rotation estimates as long as cameras observe overlapping geometry. Also, since image edges rather than intensity or color are used, the method is virtually insensitive to varying weather conditions and illumination.

1.3 Past Work

The problem of 3-D camera registration has been extensively studied. Only a small relevant subset of the large body of existing work is mentioned here.

Full structure-from-motion formulations are widely used. The majority of these rely on point correspondences, assuming short baselines in order to track features over time, and cannot be applied to wide-baseline problems. Most also estimate structure and motion using only information from a pair [10] or triple [6] of images at a time, which can lead to drift and error accumulation as the sequence progresses.

Vanishing points have been used to solve various calibration problems. Although attempts have been made at matching VPs across images to determine relative camera pose [8], global, multiple-camera rotational registration using VPs has not previously been examined.

Various techniques have been developed to detect and estimate VPs. Interactive systems (e.g. [14, 1]) rely on manual edge classification; however, this process is impractical when the number of edges or images is large. Features themselves are often input manually rather than detected, unnecessarily introducing additional error.

Image space approaches (e.g. [9, 11]) find VPs in the image plane by computing all possible 2-D edge intersections, then clustering them into groups corresponding to distinct VPs. Such methods are computationally expensive and become ill-conditioned when 3-D edges are nearly parallel to the image plane.

Perhaps the most commonly used technique is the Hough transform, which is fast and robust but whose accuracy is limited by discretization [2]. Clustering and least-squares approaches in non-discretized dual spaces (e.g. [12]) are well-conditioned over the entire input space and do not suffer from discretization artifacts; however, the clustering process can be computationally expensive or inaccurate, even in hybrid discrete/continuous approaches [2] which use a hard threshold to reject “outliers” that may in fact be noisy data.

This work addresses and overcomes some of the main difficulties with prior approaches, under a few modest assumptions. Our method handles an arbitrary number of cameras, edge features, and vanishing points in linear time. VP detection is automatic and fast (due to the HT), and VP estimation is robust and accurate (due to EM). The method operates with minimal error accumulation over arbitrarily wide baselines, as long as adjacent cameras view overlapping geometry and scenes contain sets of parallel lines. Uncertainty is modeled and used in all stages for more reliable and precise alignment.

2 Background

Before describing our method and the various algorithms therein, we present the geometric framework in which it operates.

2.1 Edge Geometry

Under pinhole projection, an image edge can be represented in several ways (Figure 2).

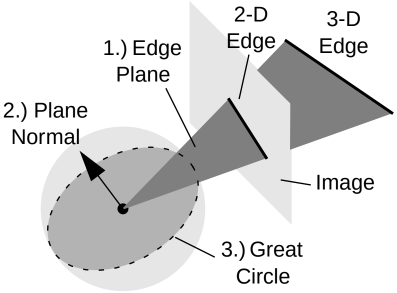

Figure 2: Edge Representations

A 3-D edge and its 2-D projection can be represented by 1.) the plane through the edge and the focal point, 2.) by the normal to this plane, or 3.) by the intersection of the plane with the Gaussian sphere.

Consider a set of parallel 3-D edges (Figure 3):

- All great circles corresponding to the edges intersect at two antipodal points on the Gaussian sphere. The direction of intersection is parallel to the 3-D edges.
- All plane normals corresponding to the edges lie on a plane whose intersection with the Gaussian sphere is a great circle. The normal to this plane is parallel to the 3-D edges.

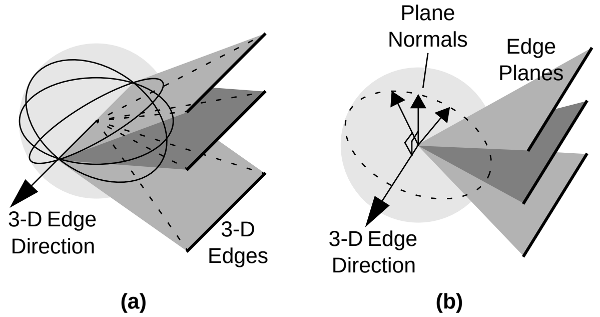

Figure 3: Parallel Edge Geometry

In (a), great circles intersect at a common point whose direction is parallel to their corresponding edges. In (b), edge plane normals all lie on a great circle.

Though formally VPs are 2-D quantities, we use the terms “vanishing point” and “3-D edge direction” interchangeably throughout this paper.

2.2 Edge Uncertainty

Image edges are parameterized by (a, b, c) satisfying

$$ax + by + c = 0 \qquad (1)$$

and estimated to sub-pixel accuracy. The estimation process also produces a 3×3 covariance matrix Λ for each edge representing uncertainty in the estimated line parameters.

A linear transformation $\boldsymbol{S}$ composed of shift, scale, and rotation can be applied to these parameters to obtain the edge plane normal $\boldsymbol{x}$. The covariance of $\boldsymbol{x}$ is then given by

$$\boldsymbol{C} = \boldsymbol{S}\Lambda\boldsymbol{S}^T . \qquad (2)$$

2.3 Position Invariance

The 3-D directions of VPs inferred from a given camera are parallel to their corresponding 3-D edges. These directions are thus scene-relative quantities expressed in the local coordinate frame of the camera, depending only on the camera’s orientation relative to the scene and not on its position. Multiple cameras, as long as they observe overlapping geometry, will infer the same VPs regardless of the cameras’ positions.

This suggests that rotational pose error can be corrected independently of translational pose error: if correspondences between VPs are known for a given camera pair, then the rotation that aligns the VPs is precisely the relative rotation between the cameras.

3 Vanishing Point Estimation

In this section we describe the component of our system that identifies and estimates prominent 3-D directions in the scene, given a set of image edges. The problem is composed of two tightly-coupled sub-problems: classification (grouping observed edges into parallel sets) and estimation (finding the best VP for each set).

Most VP estimation techniques utilize some form of discretized Hough transform (HT), which is simply a mapping between parameter spaces. Each observed image edge is parameterized, and VPs are found by locating peaks in a histogram of the parameters. The peaks identify the number and directions of VPs, and give a rough classification of edges. HT-based techniques are simple and robust but, depending on the parameterization, can exhibit singularities and discretization artifacts. Accurate peak detection is also a difficult problem.

Other estimation techniques operate in continuous rather than discrete spaces and, given good edge classification, can accurately estimate VPs. However, as discussed in [2], existing classification methods tend to be unstable, computationally inefficient, or imprecise.

We present a hybrid approach to VP detection and estimation which combines the robustness of the HT (for detection) with the accuracy of least squares (for estimation). An EM algorithm is formulated to probabilistically model edges and their directional uncertainty, obtaining accurate edge classification and direction estimates in the presence of numerous outliers and significant noise. A final verification step rejects spurious directions.

3.1 Formulation

Given a set of 2-D edges represented as uncertain normals on the Gaussian sphere, we wish to identify and precisely estimate the prominent 3-D edge directions in the original scene. Here we present a probabilistic mixture model formulation which assumes that the number of directions M is known (this is not true in practice; Section 3.5 describes a technique for finding M).

Let each of the N edges be represented by a point $\boldsymbol{x}_n$ on the unit sphere (Figure 2), and denote each of the M 3-D edge directions by $\boldsymbol{d}_j$. We wish to estimate the $\boldsymbol{d}_j$ so as to maximize a likelihood function:

$$max \prod_{n=1}^{N} P(\boldsymbol{x}_n) = max \sum_{n=1}^{N} \ln P(\boldsymbol{x}_n) . \qquad (3)$$

Since the plane normals of parallel edges lie on a great circle on the Gaussian sphere, the points $\boldsymbol{x}_n$ (in the ideal case) form coplanar sets, and the normals to the planes are the 3-D edge directions $\boldsymbol{d}_j$. If we had a classification that grouped the $\boldsymbol{x}_n$ into such sets, then we could estimate the $\boldsymbol{d}_j$ independently by fitting a 3-D plane through each set.

Similarly, if we had good estimates of the 3-D directions, we could classify each point as belonging to one of the directions.

The statistical method of expectation maximization [4] performs both classification and estimation tasks by alternating between finding the best classification given the current estimates (the E-step), and finding the best estimates given a classification (the M-step). EM is guaranteed to converge on the optimal solution given a fixed number of mixtures M and a reasonable initialization.

3.2 E-Step

Given an estimate of a direction $\boldsymbol{d}_j$ and its associated variance σ_j^2 we can compute the probability that a given point $\boldsymbol{x}_n$ belongs to this direction. Here we use a weighted zero-mean Gaussian model,

$$P(\boldsymbol{x}_n | \boldsymbol{d}_j) = \frac{1}{w_{nj}\sqrt{2\pi}\sigma_j} \exp\left(\frac{-\theta_{nj}^2}{2\sigma_j^2}\right), \tag{4}$$

where $\theta_{nj} = \sin^{-1}(\boldsymbol{x}_n \cdot \boldsymbol{d}_j)$. This formulation weights the point according to its angular deviation from the plane and w_{nj}, its uncertainty in the direction of $\boldsymbol{d}_j$. The weight w_{nj} is computed by finding the maximum eigenvalue of the symmetric matrix

$$\tilde{\boldsymbol{C}}_{nj} = (\boldsymbol{d}_j \boldsymbol{d}_j^T) \boldsymbol{C}_n (\boldsymbol{d}_j \boldsymbol{d}_j^T), \tag{5}$$

the projection of the edge's covariance matrix $\boldsymbol{C}_n$ onto $\boldsymbol{d}_j$.

From these conditional probabilities we use Bayesian arguments to derive the reverse conditionals,

$$P(\boldsymbol{d}_j | \boldsymbol{x}_n) = \frac{P(\boldsymbol{x}_n | \boldsymbol{d}_j) P(\boldsymbol{d}_j)}{P(\boldsymbol{x}_n)} \tag{6}$$

$$P(\boldsymbol{x}_n) = \sum_{j=1}^{M} P(\boldsymbol{x}_n | \boldsymbol{d}_j) P(\boldsymbol{d}_j), \tag{7}$$

where $P(\boldsymbol{d}_j)$ is the *a priori* probability of direction j (the fraction of observed points classified as belonging to $\boldsymbol{d}_j$). The probabilities $P(\boldsymbol{d}_j | \boldsymbol{x}_n)$ give each $\boldsymbol{x}_n$ a likelihood of belonging to each $\boldsymbol{d}_j$, providing a weighting mechanism for subsequent fitting steps.

3.3 M-Step

Given a set of weights for each point and each direction, we estimate the variable quantities σ_j^2, $P(\boldsymbol{d}_j)$, and $\boldsymbol{d}_j$ so as to maximize the likelihood function in (3). The estimation of prior probabilities and variance is straightforward:

$$P(\boldsymbol{d}_j) \approx \frac{1}{N} \sum_{n=1}^{N} P(\boldsymbol{d}_j | \boldsymbol{x}_n) \ . \tag{8}$$

$$\sigma_j^2 = \langle \theta_j^2 \rangle \approx \frac{\sum_{n=1}^{N} \theta_{nj}^2 P(\boldsymbol{d}_j | \boldsymbol{x}_n)}{N P(\boldsymbol{d}_j)} \tag{9}$$

Since $\theta_{nj} \approx \boldsymbol{x}_n \cdot \boldsymbol{d}_j$ for small deviations from the plane, the directions $\boldsymbol{d}_j$ can be estimated using the weighted linear least-squares formulation

$$min_{\boldsymbol{d}_j} \left\| \boldsymbol{W}_j^{n \times n} \boldsymbol{A}_j^{n \times 3} \boldsymbol{d}_j \right\|^2, \tag{10}$$

where $\boldsymbol{W}_j$ is a diagonal matrix containing the weights $P(\boldsymbol{d}_j | \boldsymbol{x}_n)$, and $\boldsymbol{A}_j$ is a matrix whose rows are the points $\boldsymbol{x}_n$. The solution can be found in closed form, assuming that the rank of $\boldsymbol{A}_j$ is at least 2 (i.e. the x_n are not all coincident), or via the SVD, by finding the vector associated with the minimum singular value of $\boldsymbol{W}_j \boldsymbol{A}_j$.

3.4 Outlier Rejection

Real images contain many edges that do not belong to any significant parallel sets. We therefore modify the EM approach above to use $M + 1$ mixtures; the last is given a large initial variance, approximating a uniform distribution over the sphere to account for spurious edges. Any edges classified as belonging to this mixture, i.e. which are best attributed to a uniform noise process, are outliers and implicitly given infinitesimal weight.

3.5 Initialization

We use a modified Hough transform to find the number M of prominent 3-D edge directions and their approximate directions $\boldsymbol{d}_j$, both of which are crucial to EM formulation and convergence. It is important to note that in this application, the HT is used only for initialization of another technique; thus, many of the concerns of pure HT approaches, such as precision, error modeling, and overcoming discretization artifacts, need not be considered in detail. Our implementation is kept as simple as possible to remain reliable, fast, and accurate.

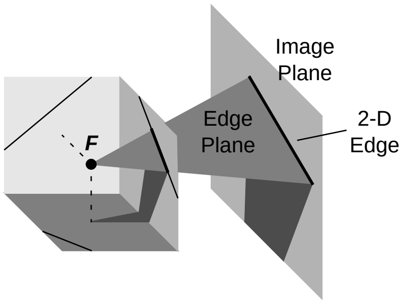

Figure 4: Hough Transform Space

Each edge forms a plane through the focal point ***F*** that intersects three faces of a cube centered at ***F***. Rasterization of the intersection increments histogram bins in each face.

Our HT parameterizes edges by intersecting their representative edge planes with the surface of a cube centered at the focal point [16], ensuring a bounded, symmetric parameter space and simplifying the implementation to 2-D line clipping and rasterization. To eliminate sign ambiguities, only three cube faces (front, top, and side) are used. These faces are discretized such that the maximum angle subtended by any bin is smaller than a specified θ_{max}, producing a complete and reasonably uniform discretization with $\theta_{max} < 2\theta_{min}$. In practice we use $\theta_{max} \approx 1°$.

It was noted in Section 2.1 that the planes of parallel edges intersect at a common point in the same direction as the 3-D edges. Peaks in the histogram thus correspond to vanishing points and are used to initialize the EM algorithm. Candidate peaks are found by searching the histogram for relative maxima, i.e. points $h_{i,j}$ in a square window of size w satisfying

$$h_{i,j} \geq h_{i+m,\, j+n} \qquad -w \leq (m, n) \leq w\,. \tag{11}$$

The window size is chosen so that its angular coverage is approximately constant with respect to varying histogram bin size. A normalized measure of peak curvature $p_{i,j}$ is also computed:

$$p_{i,j} = \frac{Wh_{i,j} - S}{(W-1)h_{i,j}} \qquad \begin{aligned} W &= (2w+1)^2 \\ S &= \sum_m \sum_n h_{i+m,\, j+n} \end{aligned}\,. \tag{12}$$

The curvature satisfies $0 \leq p_{i,j} \leq 1$, and is approximately independent of window and bin sizes. Peaks are ordered by a "strength" metric $s_{i,j} = p_{i,j} h_{i,j}$, the product of histogram count and curvature, so that both absolute and relative magnitudes are considered.

The $s_{i,j}$ are assumed to be drawn from a random distribution, and every peak for which $s_{i,j} > \mu + \alpha\gamma$ (where μ and γ are the sample mean and standard deviation, respectively) is treated as statistically significant and passed to the EM. In practice, a conservative threshold of $\alpha = 1.5$ includes all true peaks and rejects most false peaks.

3.6 Validation

False positives in HT peak detection may produce spurious EM mixtures and false VPs. Thus a validation step is performed to verify that the directions estimated by EM are statistically significant. To be considered significant, each direction $\boldsymbol{d}_j$ must meet several criteria:

- $\boldsymbol{d}_j$ matches an initial, as yet unmatched HT peak
- Edge count metric $P(\boldsymbol{d}_j) > \mu_e - \gamma_e$
- Variance metric $-\log(\sigma_j^2) > \mu_v - \gamma_v$

where (μ_e, γ_e) and (μ_v, γ_v) are the respective mean and standard deviation of the edge count and variance metrics. The logarithm of the variance is used rather than the variance itself to compensate for the extremely large variation in values that occurs in practice. Applying these criteria tends to discard VPs that lack sufficient statistical evidence (constituent edges and/or coherence).

If there is no change between the current set of edge directions and the output of the validation step, the process terminates. Otherwise, EM is performed on the validated edge directions and the process repeats.

4 Matching

Corresponding VPs must be identified across all cameras before registration is possible; the cameras can then be rotated to bring these corresponding directions into optimal alignment. We first determine an adjacency structure among all cameras, then use this structure to create a single set of unique, global (scene-relative) VPs as well as a consistent labeling identifying which cameras view each VP.

4.1 Adjacency

Each camera's k nearest neighbors (we use $k \leq 8$) are determined using approximate positions obtained from the acquisition platform. First and second order statistics on the inter-camera distances between all neighboring pairs are then calculated, and any pair separated by a distance greater than one standard deviation above the mean distance is removed. The result is an adjacency graph whose nodes correspond to individual cameras and whose arcs connect cameras likely to have viewed overlapping geometry.

4.2 Matching Adjacent Pairs

In order to correspond VPs across all cameras, VPs must first be matched between pairs of nodes across each arc of the adjacency graph. Since at least two correspondences are needed to uniquely align a pair of cameras, angles between all possible VP pairs in each camera are computed. For a given pair of cameras A and B, define θ_{mn} as the angle between VPs m and n in A, and φ_{pq} as the angle between VPs p and q in B. Each θ is compared with each φ, and a match is considered found if

$$\begin{aligned} &(\theta_{mn} - \varphi_{pq})^2 \leq t_{mnpq} \\ \text{or} \qquad &(\pi - \theta_{mn} - \varphi_{pq})^2 \leq t_{mnpq} \end{aligned} \tag{13}$$

where $t_{mnpq} = \sigma_m^2 + \sigma_n^2 + \sigma_p^2 + \sigma_q^2$. Both cases must be considered due to sign ambiguity (Figure 5).

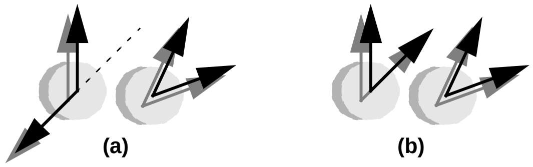

Figure 5: Sign Ambiguity

Two pairs of VPs which appear to have different relative angles (a) may actually correspond if one VP is negated (b).

A score is computed for each VP pair match. The offset camera's VPs are rotated to the reference camera (Section 5.1), and the angle of rotation is noted. Correspondences between the remaining VPs are then established using a two-case criterion similar to (13), where the squared angle between a candidate pair of VPs is compared with the sum of their variances. The score with the most correspondences is chosen to be the "best" score; if scores tie, e.g. due to rotational ambiguity (Figure 6), the score with minimum rotation angle is chosen.

Ambiguity can be further reduced when adjacent cameras view vertical edges. Vertical VPs are easily identified (either by prominence or proximity to an approximate vertical) and can be assumed to match.

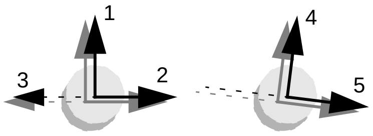

Figure 6: Rotational Ambiguity

VPs 4 and 5 most likely match 1 and 2 if approximate pose is known, although they could also match 3 and 1.

4.3 Graph Traversal

We form distinct groupings of VPs using a series of linear-time constrained depth-first searches (CDFS) on the adjacency graph. We find all matches to a given VP V by launching a CDFS from a node N that views V. The CDFS recursively traverses arcs in the graph, and for each arc performs the pair-wise matching step described in Section 4.2. If an unassigned VP in the newly reached node is found to match the set containing V, it is assigned to that set. If no such match is found, the CDFS sub-traversal terminates.

The above CDFS produces all correspondences for a single VP V across a single connected component rooted at N. A series of these searches is performed until all VPs are assigned. This algorithm results in a set of distinct VP groups, each of which represents a unique scene-relative edge direction and contains references to the individual cameras that view it.

Any group containing a single reference (i.e. any VP seen by only one camera) is removed from consideration, as are cameras that do not view at least two of the resulting VPs. Despite the fact that all cameras do not view the same subsets of scene geometry, and that some images give rise to spurious VPs, the resulting VP groupings are globally consistent.

5 Rotational Registration

Once VPs have been estimated and a consistent set of unique edge directions has been found over all cameras, we determine an optimal set of camera orientations (relative to an arbitrary rotational origin). We first discuss the two-camera solution, then generalize to $N \geq 2$ cameras.

5.1 Two-Camera Alignment

Determining the optimal rotational registration between two cameras given two or more ray correspondences has been solved in closed form [7]. Define $\boldsymbol{d}_{jk}$ ($k = 1, 2$) to be the j^{th} edge direction relative to camera k. We find the unit quaternion $\boldsymbol{q}$ that optimally aligns $\boldsymbol{d}_{jk}$ for all j by solving the least-squares system

$$min_{\boldsymbol{q}}\left\|\boldsymbol{A}^{4M\times 4}\boldsymbol{q}\right\|^2 = min_{\boldsymbol{q}}\left\|\left[\boldsymbol{A}_1 \ \dots \ \boldsymbol{A}_M\right]^T\boldsymbol{q}\right\|^2 \qquad (14)$$

where

$$\boldsymbol{A}_j = \begin{bmatrix} 0 & -s_{jx} & -s_{jy} & -s_{jz} \\ s_{jx} & 0 & a_{jz} & a_{jy} \\ s_{jy} & -a_{jz} & 0 & a_{jx} \\ s_{jz} & -a_{jy} & -a_{jx} & 0 \end{bmatrix} \qquad \begin{aligned} \boldsymbol{s}_j &= \boldsymbol{d}_{j2} - \boldsymbol{d}_{j1} \\ \boldsymbol{a}_j &= \boldsymbol{d}_{j2} + \boldsymbol{d}_{j1} \end{aligned} . \qquad (15)$$

The optimal least-squares solution to this system is the unit eigenvector associated with the minimum eigenvalue of the matrix $\boldsymbol{A}^T\boldsymbol{A}$ [5].

5.2 Weighting Correspondences

The above formulation gives equal weight to all edge directions. In practice, however, some edge directions (such as the vertical direction in urban scenes) are more prevalent than others and are estimated with higher certainty. After the VP detection stage, each edge direction $\boldsymbol{d}_{jk}$ has an associated uncertainty σ^2_{jk}, which can be used as a weighting factor in the above minimization. We thus replace the matrix $\boldsymbol{A}_j$ in (15) with

$$\boldsymbol{B}_j = \frac{1}{(\sigma^2_{j1} + \sigma^2_{j2})}\boldsymbol{A}_j , \qquad (16)$$

which improves rotation estimates by weighting high-certainty directions more heavily.

5.3 Multiple-Camera Alignment

To register $N \geq 2$ cameras, we could perform the above two-camera registration for all adjacent pairs; this would cause error to propagate and accumulate, however, since pairings are inter-dependent. One manifestation of this error is inconsistency of self-loops, e.g. $\boldsymbol{q}_{1\to 2} \times \boldsymbol{q}_{2\to 3} \neq \boldsymbol{q}_{1\to 3}$.

Here we introduce an iterative extension of the two-camera solution that accounts for all correspondences and all unique edge directions across all cameras. The algorithm produces globally optimal estimates of all camera orientations in $O(N)$ time.

We use a two-step approach much like that of full 6-DOF bundle adjustment techniques. First, relative rotations are assumed to be fixed, and all corresponding edge directions are averaged to find a best representative direction. Next, the best directions are assumed fixed, and each camera is rotated to align with them using the two-camera technique above. The procedure repeats until there is no significant change in any camera's orientation.

The resulting camera orientations are expressed relative to the current "ground-truth" directions, but can all be adjusted by a single rotation to align with any desired reference frame. This technique minimizes error accumulation and bias by optimizing all cameras simultaneously rather than in pairs.

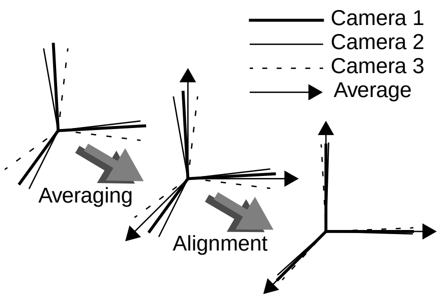

Figure 7: Iterative Rotational Refinement

Three misaligned vanishing points from three cameras are shown over one algorithm iteration.

5.4 Merging Redundancies

Occlusion and missed detection of individual VPs can give rise to multiple global VPs corresponding to the same scene geometry. After each step of the multiple-camera refinement, such redundant VPs are detected and merged when sufficiently near each other, i.e. when

$$\theta_{ij}^2 < \sigma_i^2 + \sigma_j^2 , \qquad (17)$$

where θ_{ij} is the angle between averaged VPs i and j, and σ^2 is the angular variance of an averaged VP.

6 Results

We used synthetic cameras and geometry to assess our method in the presence of various types of data corruption. We also tested the method on two real data sets consisting of a large number of hemispherical images, their associated 2-D edges, and initial pose estimates.

6.1 Synthetic Data

Synthetic 3-D edges in four directions (one vertical, two horizontal, one random) were generated and projected onto synthetic cameras. Error was introduced in several forms:

- Zero-mean Gaussian angular noise in edge projections
- Uniformly-distributed outlier edges
- Random edge removal (fixed at 30%)
- Zero-mean Gaussian rotational camera perturbation

Figure 8-a shows performance of VP detection and rotation estimation with varying amounts of edge projection error and outlier noise. Error values for controlled quantities represent the standard deviation of the noise distribution; the number of outlier edges is expressed as a percentage of the number of true scene edges. Figure 8-b shows that end-to-end rotational error using our technique (as opposed to a purely pair-wise approach) is roughly constant.

6.2 Real Data

Two sets of hemispherically-tiled, pose-annotated images from the City Project database were used as the basis for testing on real data. All tests were run on a 250MHz SGI Octane and required no more than 6MB of memory.

	TechSquare	EastCampus
total nodes	45	90
images/node	46	20
avg features/node	4,517	2,225
avg VPs/node	2.95	2.72
avg time/node	31.41s	9.75s
unalignable nodes	1	2
avg matches/pair	2.55	2.32
avg VP angle error	0.067°	0.047°

Table 1: Data Statistics

Matches/pair indicates the number of VP matches per arc of the adjacency graph. Angle error refers to the deviation of inter-VP angles from known values. Times exclude file I/O.

Although ground-truth pose is unknown, the camera orientations obtained from the TechSquare data set were compared with the results of a semi-automated registration method [3] in which point correspondences across different images were manually specified and the nodes bundle-adjusted. Relative rotations differed by no more than $0.25°$.

Convergence of VP bundles in multiple-camera registration is shown in Figure 8-c. Variances at convergence for several representative global VPs are plotted in Figure 8-d as a function of the number of cameras observing them.

We also studied the effects of camera field-of-view (FOV) on VP estimation performance (Figure 8-e, f). The VP variance was generally found to decrease as the number of images increased, although in some cases the additional feature observations introduced by larger FOV images negated this effect.

7 Conclusions

We have described a method for global rotational registration of an arbitrary number of cameras over wide baselines. In doing so, we have addressed several shortcomings of existing approaches, including computational complexity, robustness, and limitations on baseline and illumination. VP estimation proved to be virtually insensitive to outliers due to a mixture model that implicitly gives them low weight. Unalignable cameras were automatically identified and discarded. We found that global registration typically converged in just a few iterations, even with initial camera rotation errors exceeding $30°$.

Our method has several limitations. First, angle comparison in the matching process is $O(N^4)$ in the number of VPs. However, this number is typically small even for extended urban scenes, and if a vertical direction is identifiable in all images the order becomes quadratic. Second, the numerical accuracy of least-squares formulations depends on the quality and quantity of available data. 3-D directions estimated from a small number of uncertain image edges

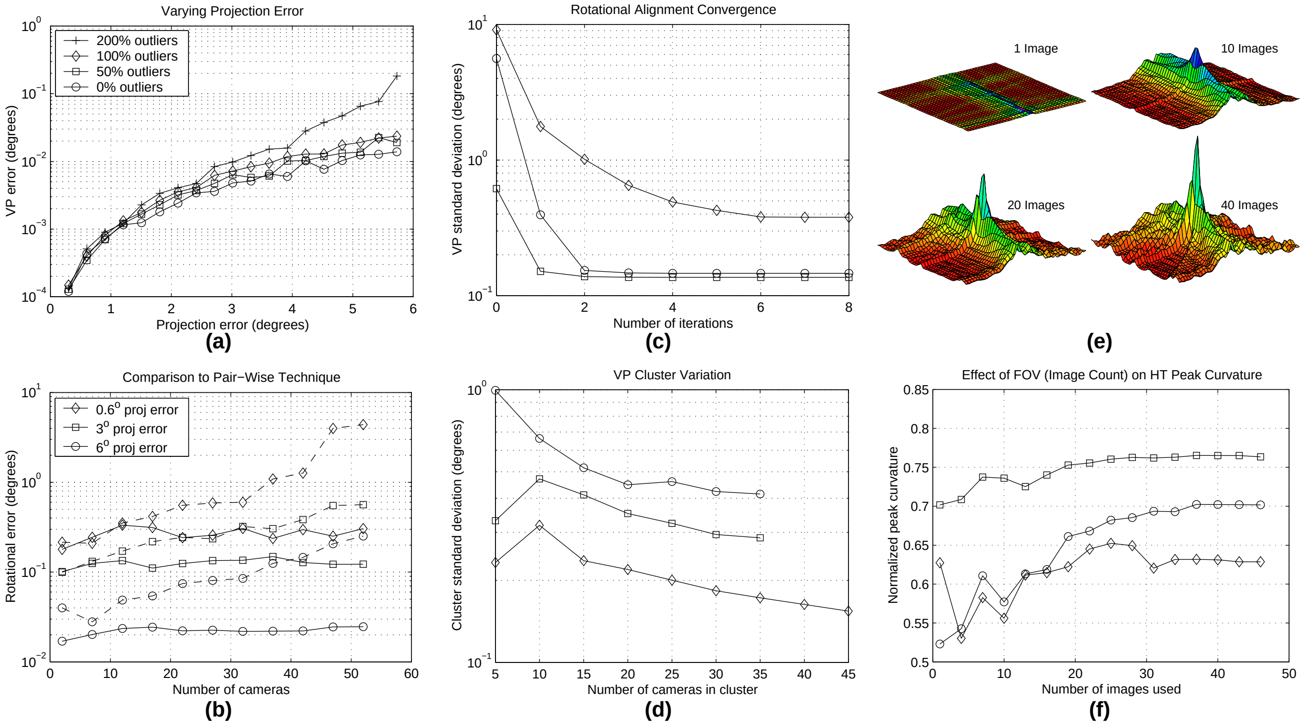

Figure 8: Performance on Synthetic (a, b) and Real (c-f) Data

(a) Effects of edge projection error on VP estimates. (b) Comparison of our multi-camera method (solid lines) to a pair-wise method (dashed lines), both with perturbed cameras ($\sigma = \pi/6$). (c) Registration algorithm convergence. (d) Variances of several global VPs at convergence, as a function of the number of cameras viewing the VPs. (e) HT sharpness as the FOV (number of images from a single node) increases. (f) Variation of HT peak curvature in several nodes as the FOV increases.

are thus somewhat unreliable, but the size and redundancy of our data set typically compensate for this effect. Finally, our method can be applied only to scenes spanning a few kilometers; "vertical" scene edges separated by more than this distance deviate in orientation by more than a milliradian due to the curvature of the Earth.

The technique described here produces not only camera orientations and VPs, but also the 3-D directions of associated image edges. This information is being used to develop algorithms for automatic translational registration.

References

[1] Becker, S. and Bove, V. M. "Semi-Automatic 3-D Model Extraction from Uncalibrated 2-D Camera Views". In *SPIE Image Synthesis Proceedings*, 1995, pp. 447-461.

[2] Collins, R. T. and Weiss, R. S. "Vanishing Point Calculation as Statistical Inference on the Unit Sphere". In *ICCV Proceedings,* 1990, pp. 400-403.

[3] Coorg, S., Master, N., and Teller, S. "Acquisition of a Large Pose-Mosaic Dataset". In *CVPR Proceedings,* 1998, pp. 872-878.

[4] Dempster, A. P., Laird, M. N., and Rubin, D. B. "Maximum Likelihood from Incomplete Data via the EM Algorithm". *Journal of the Royal Statistical Society*, Series B, No. 39, 1977, pp. 1-38.

[5] Faugeras, O. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, Massachusetts, 1996.

[6] Fitzgibbon, A. W. and Zisserman, A. "Automatic Camera Recovery for Closed or Open Image Sequences". In *ECCV Proceedings,* 1998, pp. 311-326.

[7] Horn, B. K. P. "Closed-Form Solution of Absolute Orientation Using Unit Quaternions". *Journal of the Optical Society of America*, Vol. 4, No. 4, April 1987, pp. 629-642.

[8] Leung, J. C. H. and Mclean, G. F. "Vanishing Point Matching". In *ICIP Proceedings,* 1996, pp. 305-308.

[9] Liebowitz, D. and Zisserman, A. "Metric Rectification for Perspective Images of Planes". In *CVPR Proceedings,* 1998, pp. 482-488.

[10] Luong, Q. T. and Faugeras, O. "Camera Calibration, Scene Motion, and Structure Recovery from Point Correspondences and Fundamental Matrices". *IJCV,* Vol. 22, No. 3, 1997, pp. 261-289.

[11] McLean, G. F. and Kotturi, D. "Vanishing Point Detection by Line Clustering". *PAMI*, Vol. 17, No. 11, November 1995, pp. 1090-1095.

[12] Schuster, R., Ansari, N., and Bani-Hashemi, A. "Steering a Robot with Vanishing Points". *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 4, August 1993, pp. 491-498.

[13] Shufelt, J. "Performance Evaluation and Analysis of Vanishing Point Detection Techniques". *PAMI*, Vol. 21, No. 3, March 1999, pp. 282-288.

[14] Shum, H. Y., Han, M., and Szeliski, R. "Interactive Construction of 3D Models from Panoramic Image Mosaics". In *CVPR Proceedings,* 1998, pp. 427-433.

[15] Teller, S. "Automated Urban Model Acquisition: Project Rationale and Status". In *IUW Proceedings*, 1998, pp. 455-462.

[16] Tuytelaars, T., Proesmans, M., and Van Gool, L. "The Cascaded Hough Transform". In *ICIP Proceedings*, 1998, pp. 736-739.