Simultaneous Camera Pose and Correspondence Estimation in Cornerless Images

Wen-Yan Lin¹, Guo Dong², Ping Tan¹, Loong-Fah Cheong¹ and Chye-Hwang Yan²
¹National University of Singapore, ²Defense Science Organization,
¹{g0600124,eletp,eleclf}@nus.edu.sg, ²{gdong,ychyehwa}@dso.org.sg

Abstract

We propose an algorithm which can jointly estimate camera pose and point set registration. Given point sets from two views of a stationary scene, our algorithm registers the point sets while retaining internal scene structure. It simultaneously ensures that the resultant registration is consistent with that of a moving camera viewing a static scene (adheres to some epipolar constraint). Our statistical formulation can incorporate but does not necessarily require additional constraints such as brightness constancy and high dimensional point descriptors such as SIFT. We show that our algorithm is stable over a variety of scenes and offers a pose from edge system which handles currently difficult structure from motion scenes more robustly.

1. Introduction

Obtaining camera pose is an important step in the Structure from Motion (SfM) problem. This pose recovery problem has traditionally been divided into two sequentially modules: 1) feature correspondence, i.e. the process of matching feature points across two or more different images; 2) camera pose estimation, i.e. recovering the relative positions of cameras given the matched feature points.

This sequential framework has greatly influenced the SfM community and to date has delivered many notable advances. However, it is highly dependent on our ability to reliably locate and match unique feature points between images, which in turn has greatly influenced our perception of what scene is difficult for SfM. High resolution images with an abundance of distinctive feature points are considered easy SfM problems, while low resolution images with long straight edges and few corners are considered challenging problems because of the difficulty in obtaining unique feature matches along a smooth contour (aperture problem). The later case is of practical importance, because indoor environments contain many un-textured scenes where the primary structure cue is a deformation of contours induced by camera motion [31, 32]. In fact, these contours carry structure cues to solve the SfM problem. In this paper, we propose a method to simultaneously compute camera pose (i.e. the Essential matrix) and point-to-point matches among image contours. We argue that the difficulty in handling scenes with abundant contours but less corners is not due to the intrinsic difficulty of the SfM problem. In contrast, it is because of the traditional sequential formulation, which causes SfM algorithms to break down when there are insufficient number of good matches. Feature matchers such as [18, 3, 13] cannot handle indoor scenes well, because the pixels along contours are not considered as features and, hence, no attempt is made at matching them.

Contour registration algorithms such as [22, 6, 4] match image contours by taking into account the scene structure and are generally able to make broadly accurate registrations. However, they are not designed for point-to-point correspondence and suffer from the aperture problem. This is illustrated in Figure 1, where we test the algorithm of [22]. The algorithm attempts to spread the blue circles over the red crosses while maintaining a coherent scene structure. However, the inherent ambiguity in registering lines means that compressing the rightmost vertical arm while bending up the lower arm yields an alternative ‘good’ registration which differs from the true solution. Although the overall scene structure is well registered, this registration is not consistent with any epipolar geometry and, hence, is not useful for obtaining camera poses.

Figure 2 shows that if one had the true camera pose, the aperture problem can be disambiguated using epipolar geometry and point-to-point correspondence is possible. This is a chicken-egg problem, as the camera pose cannot be obtained without accurate point-to-point correspondence and accurate correspondence cannot be obtained without camera pose. We suggest that the solution is to jointly estimate both pose and point-to-point correspondence.

In this paper, we propose a Joint Pose and Correspondence (JPC) algorithm with a sophisticated correspondence step which considers issues such as scene coherence, correspondence of high dimensional feature descriptors as well
as the epipolar constraint. The iterative algorithm alternates between trying to register edges and computing an approximate epipolar constraint from the noisy registration. As each refinement step minimizes a single statistical cost function, convergence is guaranteed. Our empirical research finds that the algorithm can frequently converge to the true pose without prior initialization. Other than computing camera pose from edges, the generality of our formulation allows it to be used as a feature point (corner) based JPC algorithm. This added flexibility would be especially useful in incorporating recently proposed edge descriptors [20]. To the best of our knowledge, there has been no previous attempt to construct such an algorithm.

Apart from addressing the automated camera pose recovery problem, our work may be of interest to researchers who desire to have limited manual intervention to obtain better results/ increased stability as this offers researchers the option of outlining certain significant contours (which can be subsequently refined using snakes) then using our algorithm to obtain the camera pose and registration. This is much less user-intensive than manual matching of features.

1.1. Related works

Iterative refinement of pose and correspondence has a long and rich history in SfM. Examples include BEEM [12], RANSAC [11], SLAM algorithms [7, 21], Visual Servoing [16] and the JPC algorithms [17, 8, 23, 25, 30, 19]. Many of these are landmark works which greatly improve SfM’s stability in previously difficult scenes.

The SfM algorithms that most closely resemble ours, are the joint pose and optical flow estimations of [25, 23, 30]. These algorithms overcome the aperture problem by implicitly or explicitly computing the normal flow, then finding a camera pose which is consistent with the said flow. Our approach differs from these in the sense that we utilize a full-fledged registration framework rather than relying on the image gradient to help constrain the problem. By not explicitly assuming the constancy of the image gradient, we can elegantly handle situations where the normal is not well defined or changing. Further, we are not dependent on small motion to help pre-localize the edges, thus enabling us to use longer baselines. This is an advantage as the ratio of flow magnitude to noise tends to be higher for small motion. This is especially likely to be a problem on edges, where the background noise tends to be higher. Finally, our more general formulation can elegantly incorporate scene invariant cues/ key points [18].

There also exist feature point based JPC algorithms such as [17, 8, 24, 19]. They handle situations where there are many ambiguous features by finding a camera pose which is consistent with the point sets. However, they are dependent upon well localized and substantially distinctive feature points and are not suited to the images of featureless contours. Given two views of a long contour, they will be degenerate since epipolar lines in nearly any direction will intersect some section of the contour. Further, [17] deals primarily with the affine fundamental matrix, [24] with the orthographic camera.

For multiple views, it is also possible to make use of structure from lines algorithms to overcome the apertures problem [14, 2, 10, 26]. An interested reader might like to peruse other works dealing with various aspects of curve / line reconstruction [1, 27, 15, 31, 28, 32].

2. Formulation

In this paper, the problem addressed is the recovery of the calibrated cameras’ relative pose (i.e. orientation and position) given two different views of a static scene.
The formulation emphasizes generality and tries to help the reader easily adapt the formulation to allow different inputs.

2.1. Definitions

Edges are described by point sets obtained by sampling the edge map of an image. Each point takes the form of a \( D \) dimensional feature vector,

\[
\begin{bmatrix}
x \\
y \\
r \\
g \\
b \\
\ldots
\end{bmatrix}_T^{1 \times D},
\]

with \( x \) and \( y \) being image coordinates, while the remaining optional dimensions can incorporate other local descriptors such as color, curvature, etc. We are given two point sets. A base point set \( B_0 \) and a target point set \( T_0 \) can be incorporated. We seek to align \( B \) until it is aligned to the target point set \( T \) while still preserving the internal scene structure of \( B_0 \). The evolution of \( B \) consists of changing only the image coordinates (first two entries) of the \( b_i \) vectors. The remaining entries are held constant to reflect the brightness/feature constancy assumption. When attempting to align the evolving base set \( B \) to the target set \( T \), we try to ensure that the image coordinates of \( b_i \), \( b_0 \), are consistent with that of a moving camera viewing a static scene (i.e. abide by some epipolar constraint).

As many equations only involve the first two dimensions of \( b_0 \) and \( b_i \), to simplify our notation, we define them as the sub-vectors \( \beta_0 \) and \( \beta_i \) respectively. We further denote the first two columns of \( B_0 \) and \( B \) by \( B_0 \) and \( B \), which are \( M \times 2 \) matrices formed by \( \beta_0 \) and \( \beta_i \). As \( B_0 \) uniquely defines \( B_0 \), the matrices can often be used interchangeably in probabilities and function declarations. The constancy of much of the \( b_i \) vector also means that the algorithms run time is largely independent of the size of \( D \), hence one can apply high dimensional descriptors on the contour points with little additional cost.

We also define a two dimensional continuous “velocity function” (in this paper, the terms velocity and flow are used loosely and do not imply any small motion approximation), \( v(.) \) such that

\[
\beta_i = v(\beta_0) + \beta_0, \tag{1}
\]

2.2. Problem formulation

We seek an aligned base set \( B \) and camera pose \( E \) (whose 5 degrees of freedom are parameterized in the form of rotation and translation), which has maximum probability given the original base and target point sets \( B_0 \), \( T_0 \). Using Bayse’s rule and assuming all \( T_0 \) and \( E \) are equally likely, this probability takes the form

\[
P(B, E | B_0, T_0) \propto P(T_0, E | B_0)
\]

\[
= P(E | B_0) P(T_0 | B_0, E).
\tag{2}
\]

We first study the term \( P(E | B_0) \). Given camera pose \( E \) and assuming independent isotropic gaussian noise of standard deviation \( \sigma_b \), the evolving base point set \( B \) has an associated probability given by

\[
P(B | E, B_0) = P(\mathcal{B} | E, B_0) = e^{-\lambda \Psi(\mathcal{B})} \prod_{i=1}^{M} g(d_i, \sigma_b).
\tag{3}
\]

where \( g(k, \sigma) = e^{-\frac{||k||^2}{2\sigma^2}} \) is a gaussian function.

The first exponent in equation (3) contains the regularization term defined as

\[
\Psi(\mathcal{B}) = \min_{v(s)} \left( \int_{\mathbb{R}^2} \left| v'(s) \right|^2 ds \right),
\]

where \( v'(s) \) is the fourier transform of a velocity field which satisfies equation (1) and \( g'(s) \) is the fourier transform of a gaussian smoothing function (with spatial standard deviation \( \gamma \)). Minimizing such a ratio suppresses the high frequency components of the velocity field, and ensures that adjacent contour points have similar motion tendencies, thus helping preserve the scenes internal structure. This is the motion coherence form proposed in [33], to allow the imposition of coherent motion on discrete point sets. It is also the form adopted in the contour registration work of [22].

The second term in equation (3) contains the epipolar constraint defined by camera pose, \( E \). As mentioned earlier, we desire that the image coordinate pairs \( \beta_0 \), \( \beta_i \), to be consistent with \( E \). Hence, \( d_i \) is the perpendicular distance of the point \( \beta_i \) from the epipolar line defined by point \( \beta_0 \) and pose \( E \). This is the probabilistic version of the reprojection error [14].

In matrix notation,

\[
d_i = \left\| R^T \left( \beta_i - (\beta_0 + r_i) \right) \right\|
\tag{4}
\]

where \( r_i \) is a two dimensional vector representing the rotational “velocity” induced on point \( \beta_0 \) by the camera pose \( E \), thus ensuring that \( \beta_0 + r_i \) is a point on the epipolar line. \( l_i \) is a two dimensional unit vector perpendicular to the epipolar line defined by \( E \) and \( \beta_0 \).

We now consider \( P(T_0 | B, E, B_0) \). This probability is the confidence measure of \( T_0 \) given \( B \). We let each \( b_i \) represent a \( D \) dimensional centroid of an equi-variant gaussian functions with standard deviation \( \sigma_j \) (we assume that the data has been pre-normalized). These form the gaussian mixture probability of \( T_0 \). Hence,

\[
P(T_0 | B, E, B_0) = \prod_{j=1}^{N} \sum_{i=1}^{M} g(t_{0j} - b_i, \sigma_j).
\tag{5}
\]
Under the Expectation Maximization (EM) framework, $\mathcal{B}$ is not necessarily close to $T_0$, thus making the probability very small. However, using the EM algorithm, we use these initial, low probabilities to move $\mathcal{B}$ to a better alignment with $T_0$.

Substituting equations (3), (5) into (2) and taking the negative log of the resultant probability, our problem becomes one of finding the $\mathcal{E}$ and $\mathcal{B}$ which minimize $A(\mathcal{B}, \mathcal{E})$,

$$A(\mathcal{B}, \mathcal{E}) = -\sum_{j=1}^{N} \sum_{i=1}^{M} g(t_{o_{j}}, b_{i}, \sigma_{i}) + \sum_{i=1}^{M} \frac{d_{i}^{2}}{2\sigma_{b}^{2}} + \lambda \Psi(\mathcal{B}).$$

(6)

The first term in $A(\mathcal{B}, \mathcal{E})$ measures how well the evolving point set $\mathcal{B}$ is registered to the target point set $T_0$, the second term, measures whether the evolving point set $\mathcal{B}$ adheres to the epipolar constraint. Finally, the third term ensures that the point set $\mathcal{B}$ evolves in a manner that tries to preserve the internal scene structure of $B_0$.

3. Joint estimation of correspondence and pose

We seek the $\mathcal{B}$ and $\mathcal{E}$ which optimize equation (6) (recall that $\mathcal{B}$ is the first two columns of $B$). Observe that this is a constrained minimization as the $l_i, r_i$ terms in $d_i$ have a non-linear relationship with the camera pose $E$ and image point $\beta_0$. This precludes other more straightforward minimization techniques. Using expectation maximization, we minimize $A(\mathcal{B}, \mathcal{E})$ by alternately updating $\mathcal{B}$ and $\mathcal{E}$. The procedure is as follows.

3.1. Updating registration, $\mathcal{B}$

In this subsection, we hold camera pose, $E^{old}$, constant, while updating $\mathcal{B}$. This results in a $\mathcal{B}^{new}$, whose associated evolving base point set $B^{new}$ is better aligned to the target point set $T_0$, while preserving its internal scene structure and respecting the epipolar lines defined by the camera pose $E^{old}$. The new registration, $\mathcal{B}^{new}$ can be computed from the $M \times 2$ linear equation in equation (8).

Here we provide the derivations. We define

$$\phi_{ij} (b_{i}, t_{o_{j}}) = g(t_{o_{j}} - b_{i}, \sigma_{i})$$

$$\overline{\phi}_{ij}(B, t_{o_{j}}) = \frac{\phi_{ij}(b_{i}, t_{o_{j}})}{\sum_{i} \phi_{ij}(b_{i}, t_{o_{j}})}.$$

(7)

Using Jensen’s inequality we obtain

$$A(\mathcal{B}^{new}, \mathcal{E}^{old}) - A(\mathcal{B}^{old}, \mathcal{E}^{old}) \leq -\sum_{j=1}^{N} \sum_{i=1}^{M} \overline{\phi}_{ij}(B^{old}, t_{o_{j}}) log \frac{\phi_{ij}(b_{i}^{new}, t_{o_{j}})}{\phi_{ij}(b_{i}^{old}, t_{o_{j}})}$$

$$+ \sum_{i=1}^{M} \left( \frac{d_{i}^{new}}{\sigma_{b}} \right)^{2} - \left( \frac{d_{i}^{old}}{\sigma_{b}} \right)^{2} + \lambda \left( \Psi(\mathcal{B}^{new}) - \Psi(\mathcal{B}^{old}) \right)$$

$$= \Delta A(\mathcal{B}^{new}, \mathcal{B}^{old}, \mathcal{E}^{old}).$$

From the above equation, we know that finding the $B^{new}$ which minimizes $\Delta A(\mathcal{B}^{new}, \mathcal{B}^{old}, \mathcal{E}^{old})$ will ensure

$$A(\mathcal{B}^{new}, \mathcal{E}^{old}) \leq A(\mathcal{B}^{old}, \mathcal{E}^{old}).$$

Dropping all the terms in $\Delta A(\mathcal{B}^{new}, \mathcal{B}^{old}, \mathcal{E}^{old})$ which are independent of $\mathcal{B}^{new}$, we obtain a simplified cost function

$$Q = \frac{1}{2} \sum_{j=1}^{N} \sum_{i=1}^{M} \phi_{ij}(B^{old}, t_{o_{j}}) \frac{\|t_{o_{j}} - b_{i}^{new}\|^{2}}{\sigma_{t}^{2}} + \sum_{i=1}^{M} \left( \frac{d_{i}^{new}}{\sigma_{b}} \right)^{2} + \lambda \Psi(\mathcal{B}^{new}).$$

Using a proof similar to that in [22], we show that the regularization term $\Psi(\mathcal{B})$ is related to $\mathcal{B}, \mathcal{B}^{old}$ by

$$\Psi(\mathcal{B}) = tr(\Gamma G^{-1} \Gamma^{T}),$$

where $G(i, j) = g(\beta_0 - \beta_0, \gamma), \Gamma = (\mathcal{B} - \mathcal{B}^{old})^{T}$. Substitute the above definition of $\Psi(\mathcal{B})$ into $Q$. Take partial differentiation of $Q$ with respect to $\mathcal{B}^{new}$ and post multiplying $G$ throughout. We have

$$\frac{\delta Q}{\delta \mathcal{B}^{new}} = \begin{bmatrix} c_1 & c_2 & \ldots & c_{M-1} & c_M \end{bmatrix} + 2\lambda \Gamma^{new} G^{-1}$$

$$= C + 2\lambda \Gamma^{new} G^{-1} = 0_{2 \times M}$$

$$= C_{G} + 2\lambda \Gamma^{new} = 0_{2 \times M}$$

(8)

Here, $c_i$ is computed as

$$c_i = \sum_{j=1}^{N} \phi_{ij}(B^{old}, t_{o_{j}}) \left( \frac{\beta_{i}^{new} - t_{o_{j}}}{\sigma_{t}^{2}} \right) + q_{i}^{old}(\beta_{i}^{new} - \beta_{0}^{old} - r_{i}^{old}),$$

where $q_{i \times 2} = (l_i, r_i), \ t_{o_{j}}$ stands for the first 2 elements of $t_{o_{j}}$ and the definitions of $l_i, r_i$ are given in equation (4). Equation (8) produces $M \times 2$ linear equations which can be solved to obtain $\mathcal{B}^{new}$.

3.2. Updating camera pose, $\mathcal{E}$

We now update the camera pose on the basis of the new correspondence set $\mathcal{B}^{new}, \mathcal{B}^{old}$. In this sub-section, we show this resolves itself into a traditional SfM bundle adjustment problem.

Replacing $B$ in equation (6) with $B^{new}$ and holding it constant, we seek to minimize the cost function $A(\mathcal{B}^{new}, \mathcal{E}^{new})$ with respect to only $E^{new}$. Only the middle term in $A(\mathcal{B}, \mathcal{E})$ depends on $E$. Using the definition of $d_i$ in equation (4), we need to minimize the simplified cost function

$$\sum_{i=1}^{M} \left( \frac{d_{i}^{new}}{\sigma_{b}} \right)^{2} = \left( \frac{d_{i}^{new}}{\sigma_{b}} \right)^{2} + \lambda \left( \Psi(\mathcal{B}^{new}) - \Psi(\mathcal{B}^{old}) \right)$$

(9)

with $\beta_i^{new}$ being the image coordinates of the evolving point set $\mathcal{B}^{new}$.

Observe that the problem of finding the $E^{new}$ which produces $l_{i}^{new}, r_{i}^{new}$ which minimize the above cost function is simply a bundle adjustment problem minimizing the reprojected error given in [14] and can be solved using the
gradient descent algorithm described in [29], with pose initialized to \( E^{old} \).

\( \mathcal{B}^{old}, E^{old} \) are replaced with \( \mathcal{B}^{new}, E^{new} \) and the algorithm returns to the first step in section 3.1. The process is iterated until convergence, as the evolving base set \( \mathcal{B} \) registers itself to the target set \( T_0 \), guided by the iteratively updated epipolar lines.

### 3.3. Initialization and iteration

Since we do not assume a prior estimate of camera pose or correspondence, we initialize \( \mathcal{B} \) to \( \mathcal{B}_0 \) and set \( l \) to zero which from equation (6) will cause the algorithm to ignore the epipolar constraint. \( \mathcal{B}^{new} \) is calculated from \( E^{old} \), \( \mathcal{B}^{old} \), \( E^{new} \) is calculated from \( \mathcal{B}^{new} \) and \( \mathcal{B}_0 \), after which \( \mathcal{B}^{old}, E^{old} \) are replaced with \( \mathcal{B}^{new}, E^{new} \). The process is iterated until convergence. A summary of the algorithm is given in figure 3.

For stability, we set \( \sigma_1, \sigma_b \) to artificially large values, then steadily anneal them smaller. For stability to outliers one may consider using the robustified version of \( \phi_{ij}(b_i, t_{ij}) \) in equation (7), given by \( \overline{\phi}_{ij}(B, t_{ij}) = \sum_i \phi_{ij}(b_i, t_{ij}) \mathcal{N}(k(2\sigma_i)^{-1}) \), where \( k \) controls the region of support of the uniform pdf.

**Input:** Point sets, \( \mathcal{B}_0, T_0 \)

- Initialize \( \sigma_1, \sigma_b, \gamma, \lambda \);
- Initialize \( \mathcal{B}^{old} \) as \( \mathcal{B}_0 \), \( l \) to zero vector;

while \( \sigma_1, \sigma_b \) above threshold do

while No convergence do

- Use eqn (7) to evaluate \( \phi_{ij}(b_i^{old}, t_{ij}) \) from \( \mathcal{B}^{old}, E^{old} \);
- Use eqn (8) to determine \( \mathcal{B}^{new} \) from \( \phi_{ij}(b_i^{old}, t_{ij}) \);
- Use bundle adjustment to obtain \( E^{new} \) from \( \mathcal{B}^{new} \) and \( \mathcal{B}_0 \);
- Replace \( \mathcal{B}^{old}, E^{old} \) with \( \mathcal{B}^{new}, E^{new} \);
end

Anneal \( \sigma_1 = \alpha \sigma_1, \sigma_b = \alpha \sigma_b \), where \( \alpha < 1 \).

end

Figure 3. Algorithm

### 4. Results

We present the results of our algorithm. \( \Delta T \) denotes translation error which measures the angular deviation of the computed translational direction from the true translational direction (translation magnitude is not measured due to scaling ambiguity). \( \Delta R \) denotes average rotational error about each axis. All errors are in degrees. Unless otherwise stated, displacements given in pixel units assume the pixels come from a \( 480 \times 640 \) image.

#### 4.1. Synthetic data

In figure 4, we present our attempts at registering a synthetic point set that consist of a rectangular frame with the horizontal and vertical arms set at different depths. From top to bottom, the figures show the base point set, \( \mathcal{B}_0 \), the target point set \( T_0 \), the evolved point set \( \mathcal{B} \) overlayed on the target set \( T_0 \) and the “optical flow” which maps \( \mathcal{B}_0 \) to \( \mathcal{B} \). We evaluated our algorithm and the coherent point drift algorithm of [22] on two kinds of motion. Figures 4(a),4(b) involve a pure translation in the \( Y \) direction. Figures 4(c),4(d) involve a similar translation coupled with a rotation of 0.1 radians about the \( X \) axis. If pose is calculated using the registration of [22], the error in the computed pose is \( \Delta T = 40.1^\circ, \Delta R = 6.9^\circ \) and \( \Delta T = 34.1^\circ, \Delta R = 12.8^\circ \) for figures 4(a),4(c) respectively. For our algorithm, the errors are negligible.

Other contour matching algorithms were tested [6, 4], however, our empirical evaluations support the claim in [22] that his algorithm is generally more stable than his counterparts. As such, we compare our results against that of [22].

The point sets have no feature descriptors and are totally ambiguous to a feature matcher that tries to assign matches on the basis of descriptor uniqueness. As illustrated previously, the registration produced by contour matching algorithms is unsuited for camera pose recovery. Hence, the simulation results show that our joint pose and edge estimation algorithm works well under conditions which are difficult for the sequential SfM framework.

#### 4.2. Real data

We evaluate our algorithm on pairs of real images. The results are presented in tables 1. The stereo pairs tested are given in figure 5. The first three pairs involve pure lateral translation, while the fourth and fifth involve rotation and forward translation respectively. Details of the experiments are as follows.

1. (BAL, BAH): Bundle Adjustment Low resolution, Bundle Adjustment High resolution. SIFT matches were obtained using the algorithm in [18]. RANSAC was used to help remove outliers before bundle adjustment was employed. The bundle adjustment was initialized with the true rotation and translation parameters. This algorithm was evaluated at two resolutions, low resolution is \( 60 \times 80 \) (BAL), while high resolution is \( 480 \times 640 \) (BAH). Due to the random nature of RANSAC, we made 5 iterations and took the best result.

2. (CMA): Contour Matching Algorithm. [22] is used to register two sets of edges, after which bundle adjustment is used to recover the camera pose.
3. (PFE): Pose from Edge the approach proposed in this paper. Parameters for all images were set as follows, $\sigma_b, \sigma_t$ were set at 0.1, the annealing ratio $\alpha$ was set to 0.97. $\lambda, \gamma$ were set to 1. Although our algorithm can incorporate color, it was not used because the gray value along the contour is too unstable.

4. (PFE-SIFT): Similar to PFE, except that the point set is made of unmatched SIFT features rather than edge-maps. Parameters are identical with those of PFE. Algorithm was evaluated at high resolution (480 x 640).

For (CMA) and (PFE), edges were obtained using canny edge detector. For edge stability, the images were downsampled to 60 x 80. A limited outlier rejection algorithm is applied by rejecting contours where less than 60% of their pixels did not have any neighboring contour pixels within a 30 pixel radius of their position in the next image.

4.3. Changing baseline

We evaluate a more complex version of our system with a steadily increasing baseline. We set a camera stand to its minimum elevation and progressively doubled the baseline until its maximum elevation was reached.

As we wish to focus on evaluating the intrinsic stability of our algorithm to different base-lines, we try to reduce the impact of outliers, mis-detected edges (which become more server with larger baselines), by using a more complicated edge detector and outlier rejection system. Edges were extracted using the Harris measure [9]. We use the autostitch algorithm [5] to obtain a homography between an image pair. We utilize this homography to perform a limited outlier removal. Test images are shown in figure 6, results are tabulated in table 2. The results show that our system is stable over a fairly wide baseline.

Table 2. Rotation and translation errors for stereo pairs with varying baseline shown in figure 6

<table>
<thead>
<tr>
<th>Approx displacement (pixels)</th>
<th>10</th>
<th>19</th>
<th>38</th>
<th>76</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approx displacement (pixels)</td>
<td>2.43</td>
<td>6.17</td>
<td>2.43</td>
<td>0.81</td>
</tr>
<tr>
<td>Rotation error (degrees)</td>
<td>0.74</td>
<td>1.11</td>
<td>0.4</td>
<td>0.84</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper, we present a joint pose and correspondence estimator. It is formulated in a more general fashion than previous joint pose and correspondence estimators, with the correspondence measure being able to take into account issues such as scene coherence, brightness constancy or high dimensional feature invariance. We tested our formulation using the PFE and PFE-SIFT variants.

Our PFE algorithm is able to work on low resolution images where there are few well defined corners and, hence, makes a feature based SIM algorithm impractical. PFE-SIFT is a fairly robust joint feature matching and correspondence algorithm. The long 128 dimensional feature descriptors do not add significantly to computational time, while they aid computational stability.

In general, feature point based methods are still superior (our PFE-SIFT algorithm works better than our PFE algorithm). However, our work offers an alternative for scenes
in which distinctive point features are difficult to localize. Further, we believe that much of the difficulty faced by our PFE algorithm lies in the noisiness of current edge localization techniques. Better edge detection techniques should improve our algorithms performance significantly.

6. Acknowledgements

This work was supported by DSO grant R263-000-400-592. Ping Tan is supported by the Singapore FRC Grant R-263-000-477-112. We are also deeply appreciative of the help by Liu Siying and Hiew Litt Teen.

References


