Outlier Detection in Video Sequences under Affine Projection*

D. Q. Huynh
School of Information Technology
Murdoch University
Perth WA 6150, AUSTRALIA
Email: d.huynh@murdoch.edu.au

A. Heyden
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, SWEDEN
Email: heyden@maths.lth.se

Abstract

A novel robust method for outlier detection in structure and motion recovery for affine cameras is presented. It is an extension of the well-known Tomasi-Kanade factorization technique designed to handle outliers. It can also be seen as an importation of the LMedS technique or RANSAC into the factorization framework. Based on the computation of distances between subspaces, it relates closely with the subspace-based factorization methods for the perspective case presented by Sparr and others and the subspace-based factorization for affine cameras with missing data by Jacobs. Key features of the method presented here are its ability to compare different subspaces and the complete automation of the detection and elimination of outliers. Its performance and effectiveness are demonstrated by experiments involving simulated and real video sequences.

1. Introduction

One of the main problems in computer vision is the so-called structure and motion problem, where both the structure of the scene and the motion of the camera are estimated from image measurements only. The problem can appear in several different settings depending on the camera model used and the type of image features available. In this paper we will investigate the problem for the affine camera model and point features.

The affine camera model, introduced in [9], has been used frequently to simplify the structure from motion problem in computer vision [6]. It uses fewer parameters and is a good approximation of the pin-hole camera when the distance between the camera and the object is large in comparison to the depth-variations in the object. Mathematically, it can be described as an affine transformation from three-dimensional affine space to two-dimensional affine space.

It is well-known that, given corresponding point coordinates, both structure and motion can be obtained by factorizing a matrix built up from image measurements. Unlike perspective projection which requires the estimation of the *relative depths* (or *projective depths*) of all point coordinates before factorization can be carried out (see [13, 4, 12]), under affine projection these relative depths can all be set to unity [14] and so the structure and motion problem is much easier to deal with. The factorization method requires the singular value decomposition of a rather large matrix.

The problem of outliers in structure and motion recovery from images is well known in the literature. The RANSAC (Random Sample Consensus) paradigm proposed by Fischler and Bolles [2] detects outlying data by first randomly selecting samples of the minimum number of data items required to estimate a given entity and then looking for consensus of the estimates among the samples. This paradigm and the LMedS (Least Median Squares) approach have both been applied to the computation of the fundamental matrix [15, 19, 16, 18]. One of the advantages of the factorization approach [14], compared to those using matching tensors, is that all the available image data are used simultaneously and uniformly. Unfortunately, a problem with the current factorization approach is that the image measurement matrix is assumed to be outlier-free. Although Tomasi and Kanade [14] and Jacobs [5] have both extended the factorization approach to deal with missing data, there have been no reports, to date, dealing with outliers for this approach.

A major contribution of this paper is to extend the factorization approach to one that can handle outliers while treating all the data simultaneously. This is done by considering the subspaces (see the work reported by Sparr [12] and Triggs [17]) spanned by different subsets of columns in the image measurement matrix and introducing a similarity measure on these subspaces. This similarity measure makes it possible to compare different subspaces, which is an essential part of our method. This is not a trivial problem, since a fixed subspace can be represented in many different ways and the subspace distance function must be so defined that it is independent of the choices of basis that spans the subspaces.

The paper is organized as follows. In Section 2, we give a brief review of the affine factorization method. Then

*This research was, in part, supported by the Murdoch Special Research Grant MU.AMH.D.413 and the SSF/SRC-JIG project 95-64-222.

0-7695-1272-0/01 $10.00 © 2001 IEEE

similarity functions for comparing linear subspaces are discussed in Section 3. The proposed method for robust factorization is described in Section 4 and some experiments given in Section 5. Some related issues are discussed in Section 6. Finally, conclusions are presented in Section 7.

2. Review on affine factorization

2.1. Notation

In the text that follows, we use uppercase letters to represent matrices, uppercase bold letters to represent scene points and the special joint shape matrix, lowercase bold letters to represent vectors, image points, and the special joint image measurement matrix. We denote subspaces by calligraphic letters. All vectors are assumed to be column vectors and $^\top$ denotes matrix and vector transpose.

2.2. Background

Let $\tilde{\mathbf{X}} = (\tilde{X}, \tilde{Y}, \tilde{Z}, 1)^\top$ be a scene point and $\tilde{\mathbf{x}} = (\tilde{x}, \tilde{y}, 1)^\top$ be the corresponding image point. Under the affine camera model, the projection matrix $\tilde{P} \in \Re^{3\times 4}$ has the form

$$\begin{bmatrix} P & \mathbf{q} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

where $P \in \Re^{2\times 3}$ and $\mathbf{q} \in \Re^2$. The common approach to take from here is to reduce $\tilde{P}$ to a 2×3 matrix by offsetting the coordinates of all the image points by the centroid of the image point cluster. This effectively eliminates the requirement of computing $\mathbf{q}$, leaving the unknown projection matrix as a 2×3 matrix and the unknown scene point $\mathbf{X}$ in inhomogeneous coordinates. However, since the centroid of the image point cluster cannot be reliably estimated when outliers are present, it is better to retain the projection matrix as a 2×4 matrix, as given below:

$$\mathbf{x} \equiv \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} P & \mathbf{q} \end{bmatrix} \begin{bmatrix} \tilde{X} \\ \tilde{Y} \\ \tilde{Z} \\ 1 \end{bmatrix}$$

$$\Leftrightarrow \mathbf{x} = \tilde{P}\tilde{\mathbf{X}}. \quad (1)$$

By stacking up the n scene points and their image coordinates in the m images we obtain the following relation connecting the joint affine projection matrix $\hat{P} \in \Re^{2m\times 4}$, the joint affine shape matrix $\hat{\mathbf{X}} \in \Re^{4\times n}$, and the joint image measurement matrix $\hat{\mathbf{x}} \in \Re^{2m\times n}$:

$$\begin{bmatrix} \tilde{P}_1 \\ \tilde{P}_2 \\ \vdots \\ \tilde{P}_m \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{X}}^1 & \tilde{\mathbf{X}}^2 & \cdots & \tilde{\mathbf{X}}^n \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1^1 & \mathbf{x}_1^2 & \cdots & \mathbf{x}_1^n \\ \mathbf{x}_2^1 & \mathbf{x}_2^2 & \cdots & \mathbf{x}_2^n \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_m^1 & \mathbf{x}_m^2 & \cdots & \mathbf{x}_m^n \end{bmatrix}$$

$$\Leftrightarrow \hat{P}\hat{\mathbf{X}} = \hat{\mathbf{x}} \equiv \begin{bmatrix} \hat{\mathbf{x}}^1 & \cdots & \hat{\mathbf{x}}^n \end{bmatrix} \quad (2)$$

where $\tilde{P}_i \in \Re^{2\times 4}$ denotes the affine projection matrix of camera i in the joint affine projection matrix $\hat{P}$, $\tilde{\mathbf{X}}^j$ the j^{th} scene point in the shape matrix $\hat{\mathbf{X}}$, $\mathbf{x}_i^j$ the j^{th} image point in image i, and $\hat{\mathbf{x}}^j$ a $2m$-vector that encompasses all the image coordinates in the m images of the j^{th} scene point. We call $\hat{\mathbf{x}}^j$ the *observation* of the j^{th} scene point.

2.3 Outliers and subspaces

In the absence of localization errors and outliers, the true subspace $\bar{\mathcal{X}} \subset \Re^{2m}$ spanned by the columns of $\hat{\mathbf{x}}$ is 4-dimensional. Thus, any four or more observations (i.e. columns) of $\hat{\mathbf{x}}$ should give the same subspace $\bar{\mathcal{X}}$. Localization errors of image point coordinates in $\hat{\mathbf{x}}$ would inflate the rank of $\hat{\mathbf{x}}$, giving a small non-zero value to the 5^{th} singular value. Truncating $\hat{\mathbf{x}}$ to a rank-4 matrix would give a slightly distorted 4-dimensional subspace $\hat{\mathcal{X}}$. When outliers are present, such distortion can be so significant that one must first exclude the outlying observations before attempting a reasonable estimation of the 4-dimensional subspace.

To discuss further about the distortion to the true subspace $\bar{\mathcal{X}}$ due to outliers, a similarity function for comparing two subspaces must be first defined.

3. Similarity functions for subspace comparison

In the discussion that follows, uppercase calligraphic letters are used to represent subspaces, e.g. $\mathcal{A}$. An uppercase, identical letter will represent a matrix whose columns form an orthonormal basis (which can be computed using the Matlab function `orth`) of the corresponding subspace, e.g. $A \equiv \begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_m \end{bmatrix}$ is the matrix for subspace $\mathcal{A}$. Here, $m = \dim(\mathcal{A})$ is the dimension of subspace $\mathcal{A}$ and $\{\mathbf{a}_i \mid 1 \le i \le m\}$ is an orthonormal basis that spans $\mathcal{A}$.

When dealing with two subspaces, one is often interested in obtaining a similarity function that measures the *closeness* between them. There are some criteria that such a similarity function ϕ should satisfy:

- $\phi(\mathcal{A}, \mathcal{B}) = 0 \Leftrightarrow \mathcal{A} = \mathcal{B}$.
- $0 \le \phi(\mathcal{A}, \mathcal{B}) = \phi(\mathcal{B}, \mathcal{A}) < \infty$ for all $\mathcal{A}$ and $\mathcal{B}$.
- $\phi(\mathcal{A}, \mathcal{B}) \le \phi(\mathcal{A}, \mathcal{C}) + \phi(\mathcal{C}, \mathcal{B})$ for all $\mathcal{A}, \mathcal{B}, \mathcal{C}$.

The first criterion imposes the condition that identical subspaces should have a zero similarity measure, the second criterion that the similarity function must be symmetric and finite, and the third criterion that the triangle inequality holds for any three subspaces. In the mathematics literature such concepts are called *metrics*.

We have investigated into two different similarity functions for subspaces. We shall show that these functions are related and then describe an alternative similarity function which has significant computational advantages over the other two.

3.1. Similarity function ϕ_1

The first similarity function is derived from a set of *principal angles* (see [1, 3]) between the two subspaces of concern. The definition of the largest principal angle is given below.

Definition. Let $\mathcal{A}$ and $\mathcal{B}$ be two m-dimensional subspaces in $\Re^n$ satisfying the condition $\dim(\mathcal{A}) = \dim(\mathcal{B}) = m \geq 1$. The largest *principal angle* $0 \leq \theta_m \leq \pi/2$ between $\mathcal{A}$ and $\mathcal{B}$ is defined by $\cos(\theta_m) = s_m$, i.e.

$$\phi_1(\mathcal{A}, \mathcal{B}) = \cos^{-1}(s_m)$$

where s_m is the smallest singular value of $A^\top B$. □

The estimation of θ_m thus requires carrying out the SVD of $A^\top B$. We note that the orthonormal bases in matrices A and B can be arbitrary since these bases would be aligned by the SVD to give the desired θ_m. It is straightforward to verify that ϕ_1 satisfies all of the three listed criteria.

3.2. Similarity function ϕ_2

The second similarity function that we have investigated is known as the *subspace distance* as described below (see also Golub and Van Loan [3]).

Definition. Let $\mathcal{A}$ and $\mathcal{B}$ be two m-dimensional subspaces in $\Re^n$ where $m \geq 1$. Let A and B be, respectively, the two matrices whose columns are the orthonormal bases of $\mathcal{A}$ and $\mathcal{B}$. Then the *subspace distance* ϕ_2 between $\mathcal{A}$ and $\mathcal{B}$ is defined by

$$\phi_2(\mathcal{A}, \mathcal{B}) = \|A^\top B^\perp\|_2 = \|B^\top A^\perp\|_2 \qquad (3)$$

where $A^\perp$ denotes a matrix whose columns form an orthonormal basis for $\mathcal{A}^\perp$ (the orthogonal complement of the subspace $\mathcal{A}$), and $\|.\|_2$ denotes the 2-norm of the matrix concerned. □

It can be verified that ϕ_2 gives an upper bound of 1, corresponding to when $\mathcal{A} \cap \mathcal{B}^\perp \neq \emptyset$ (and $\mathcal{B} \cap \mathcal{A}^\perp \neq \emptyset$). The function ϕ_2 also satisfies the criteria given above provided that the subspaces in question have the same dimension.

The similarity functions ϕ_1 and ϕ_2 are related by (a proof can be found in [3] (pages 76–77))

$$\phi_2(\mathcal{A}, \mathcal{B}) = \sqrt{1 - \cos^2(\phi_1(\mathcal{A}, \mathcal{B}))}. \qquad (4)$$

3.3. Similarity function ϕ_3

While $\phi_1(\mathcal{A}, \mathcal{B})$ involves an SVD on $A^\top B$ and $\phi_2(\mathcal{A}, \mathcal{B})$ involves an SVD on $A^\top B^\perp$, function ϕ_2 involves an additional SVD on B (or A) to get $B^\perp$ (or $A^\perp$). To avoid computation of the orthogonal complement of a subspace, as required in ϕ_2, and the inverse of cosine, as required in ϕ_1, we may define

$$\phi_3(\mathcal{A}, \mathcal{B}) = \sqrt{1 - s_m^2} \qquad (5)$$

where s_m, as before, is the smallest singular value of $A^\top B$. This ensures that ϕ_3 and ϕ_2 return the same distance measure of two given subspaces.

In addition to the saving in computation, another advantage of using ϕ_3 instead of ϕ_2 arises when we need to deal with subspaces of different dimensions. We note that ϕ_2 is not symmetric but ϕ_3 is (and so is ϕ_1) when the two subspaces in consideration have different dimensions, i.e. $\phi_2(\mathcal{A}, \mathcal{B}) \neq \phi_2(\mathcal{B}, \mathcal{A})$ but $\phi_3(\mathcal{A}, \mathcal{B}) = \phi_3(\mathcal{B}, \mathcal{A})$ if $\dim(\mathcal{A}) \neq \dim(\mathcal{B})$.

From here on, we will use the similarity function ϕ_3 to determine the *closeness* of two given subspaces.

4. Subspace computation and outlier detection

The rank-4 property of our joint image measurement matrix $\hat{\mathbf{x}}$ given in (2) requires a minimum of 4 observations to estimate the 4-dimensional subspace. Thus, if a quadruplet of observations is tracked through a video sequence then the rank-4 subspace spanned by them can be used for factorization, provided that the 4 observations are the projections of 4 scene points in general position, i.e. no more than 3 of them are coplanar in space.

Given that there are n observations in a video sequence, our goal is to look for those observations that are inliers and use only them to factorize to the required $\hat{P}$ and $\hat{\mathbf{X}}$ matrices. The approach we have taken is analogous to the LMedS method [10] implemented by Zhang et al for the fundamental matrix computation. Since localization errors and outliers often inflate the rank of $\hat{\mathbf{x}}$, it would be of interest for our future study to examine the magnitude of the 5^{th} singular value of the sub-matrix of $\hat{\mathbf{x}}$ if more observations are selected. Thus, in this report, we have chosen to sample 5 observations from $\hat{\mathbf{x}}$ to form a $\Re^{2m\times 5}$ matrix and study the rank-4 truncation of the matrix.

Assume that 5 observations are drawn randomly from the n observations and a $\Re^{2m\times 5}$ matrix $\check{\mathbf{x}}$ is constructed and truncated to rank 4. If a matrix $\check{\mathbf{x}}^j \in \Re^{2m\times 6}$, constructed by including the j^{th} observation of $\hat{\mathbf{x}}$ in $\check{\mathbf{x}}$ and then truncated to rank 4, is compared with $\check{\mathbf{x}}$, the distance between the subspaces allows us to assess the consistency of the two subspaces. We can add the remaining observations, one (indexed by j) at a time, to $\check{\mathbf{x}}$ and compute the subspace distance, d^j, between the subspaces in $\check{\mathbf{x}}$ and $\check{\mathbf{x}}^j$. Obviously, if the original sample that constitutes $\check{\mathbf{x}}$ is free of contamination then the list of subspace distances $\{d^j \mid 1 \leq j \leq n\}$ would have many small values and so a small median. Out of a sufficient number of samples chosen from the n obser-

vations, the winning sample should have the smallest median and should be free of outliers.

The number of samples w that require testing is often quite small. If the percentage of outliers is known to be ϵ and the probability of obtaining at least one outlier-free sample is v then

$$1 - (1 - (1 - \epsilon)^p)^w = v$$

where p, which is number of observations in each sample, is 5 in our method.

The procedure of our method is given below.

1. Randomly draw a quintuplet of observations from the n observations.
2. For each quintuplet, indexed by k, construct the rank-4 matrix $\check{\mathbf{x}}_k \in \Re^{2m\times 5}$, i.e. $\check{\mathbf{x}}_k$ has its fifth, smallest singular value set to 0.
3. For each observation $\hat{\mathbf{x}}^j$ (where $j = 1, \cdots, n$) in the n tracked observations,
   (a) construct the rank-4 matrix $\check{\mathbf{x}}_k^j \in \Re^{2m\times 6}$ defined as $\check{\mathbf{x}}_k^j = \left[\, \check{\mathbf{x}}_k \;\; \hat{\mathbf{x}}^j \,\right]$. Again, the rank-4 constraint is imposed to $\check{\mathbf{x}}_k^j$.
   (b) compute the subspace distance d_k^j between $\check{\mathbf{x}}_k$ and $\check{\mathbf{x}}_k^j$ using the similarity function ϕ_3 described in Section 3.

   If the k^{th} quintuplet sample contains no outliers then the 4-dimensional subspace should be consistent with the majority of the tracked observations. That is, the majority of values in the list of subspace distances $\left\{ d_k^j \mid 1 \le j \le n \right\}$ should be small. If outliers are present in the sample then the majority of the tracked observations would be large.
4. For the k^{th} sample, retrieve the median from the list $\left\{ d_k^j \mid 1 \le j \le n \right\}$ and store it in the entity d_k. That is, $d_k = \text{median} \left\{ d_k^j \mid 1 \le j \le n \right\}$.
5. Go back to step 1 for the next sample until w samples have been randomly drawn.
6. Follow the LMedS method and retrieve the smallest median value $\hat{d}$ in the list and the sample index $\hat{k}$ that corresponds to $\hat{d}$. That is, set $\hat{d} = \min_k d_k$ and $\hat{k} = \arg\min_k d_k$.
7. The next step is to isolate the outliers and eliminate them. The robust standard deviation estimate is given by [10]

$$\hat{\sigma} = 1.4826 \left(1 + \frac{5}{n-p}\right) \sqrt{\hat{d}} \qquad (6)$$

   and a threshold value t is set to $2\hat{\sigma}$. Those observations (indexed by j) having their $d_{\hat{k}}^j$ values (stored in the list $\left\{ d_{\hat{k}}^j \mid 1 \le j \le n \right\}$) larger than the threshold t are classified as outliers.
8. One may now proceed with the traditional approach of setting the image origin of each image at the centroid of the inlying point cluster. Construct the joint image measurement matrix using these observations, truncate it to rank 3, and factorize it to retrieve the projective form of $\hat{P}$ and $\hat{\mathbf{X}}$.

5. Experiments

We have tested our method on both synthetic image data and real video sequences. In all the conducted experiments, the percentage of outliers ϵ was assumed to be 40% and the probability v was set to 99%. This gave $w = 57$ (number of samples), regardless of the number of observations and number of image frames taken to compose matrix $\hat{\mathbf{x}}$.

For all the synthetic tests, 18 to 24 scene points were synthesized. Their image points were obtained by affinely projecting the scene points onto 5 images to simulate the capturing of a distant scene from 5 different viewpoints. Assuming that, in the feature tracking process, we used a window of approximately 15×15 pixels to identify corresponding image points, the coordinates of a number of observations in *some* images were perturbed by as large as ± 7 pixels to simulate outliers. In addition, all the inliers were also perturbed up to ± 0.2 pixel to simulate small localization errors. In the synthetic test reported here, 24 scene points with localization errors were created and 9 outliers, corresponding to a 37.5% contamination, were synthesized: these were points $2, 6, 11, 13, 15, 16, 17, 19, 20$. The algorithm given in the previous section was applied. To illustrate the effectiveness of using subspace distances to identify outliers, the values of two d_k^j lists (see Step 7) are given in Table 1. In the experiment, the 5^{th} sample (i.e. $\hat{k} = 5$), which contained the observations $\{1, 3, 8, 18, 23\}$, was selected by the algorithm to be the winner. In comparison with a sample that had been contaminated by just 1 outlier, e.g. the 3^{rd} sample which contained the quintuplet $\{4, 11, 22, 23, 24\}$ shown in Table 1, the winning sample clearly has its majority of subspace distances much smaller. The smallest median value $\hat{d}$ of this experiment was computed to be 0.0050, giving $\hat{\sigma} = 0.0093$ and the threshold value $t = 0.0187$. All the outliers were successfully identified.

The reprojection errors and reconstruction errors (after using control points from the scene to upgrade the projective structure to Euclidean) were also compared in the synthetic tests. In the synthetic test reported here, the root mean squared reprojection errors with all scene points included and with only the inlying scene points included were respectively 3.741 and 0.237 pixels. The corresponding root mean squared 3D reconstruction errors were 2.1130 and 0.1033

Table 1. The computed ϕ_3 values for the 5th and the 3rd samples.

j	1	2	3	4	5	6
d_5^j	0.0014	0.0605	0.0011	0.0014	0.0014	0.0434
j	7	8	9	10	11	12
d_5^j	0.0047	0.0030	0.0028	0.0011	0.0255	0.0029
j	13	14	15	16	17	18
d_5^j	0.0933	0.0058	0.0935	0.0381	0.0548	0.0003
j	19	20	21	22	23	24
d_5^j	0.1368	0.0221	0.0023	0.0052	0.0005	0.0077
j	1	2	3	4	5	6
d_3^j	0.1878	0.1435	0.0672	0.0000	0.0052	0.1144
j	7	8	9	10	11	12
d_3^j	0.2391	0.2062	0.1200	0.0749	0.0001	0.0243
j	13	14	15	16	17	18
d_3^j	0.0761	0.1422	0.1809	0.2732	0.2088	0.1005
j	19	20	21	22	23	24
d_3^j	0.4936	0.1220	0.0946	0.0005	0.0043	0.0028

units. The improvement to the reprojection errors and reconstruction errors by our method is significant.

Several experiments involving real data have been conducted to test our method. Due to the lack of space, only two experiments are reported. For each of the real video sequences, the KLT feature tracker [7, 11] was applied to the entire sequence of images to track feature points from one frame to the next. However, when constructing the joint image measurement matrix $\hat{\mathbf{x}}$, only a small subset of images was chosen. The reason of doing so is to illustrate the use of subspace distances to detect and eliminate outliers without getting into the computational complexity issues of factorizing a large $\hat{\mathbf{x}}$ matrix.

The first video sequence, which was downloaded from the CMU web site, contains 50 images (512×480 pixels) of buildings. A total of 475 observations in the 50 images were tracked. Five images (every 8th images) were selected from the image sequence, giving a matrix $\hat{\mathbf{x}} \in \Re^{10\times475}$. Figure 1 shows the five images used in our experiment, superimposed on each image are the tracked image feature points (marked as blue dots). Application of the algorithm described in the previous section led to the computation of $\hat{d}$ and threshold t to be 0.0058 and 0.0173. A total of 427 inliers were identified.

In this experiment, regular, repetitive patterns in the scene posed some challenge to the algorithm. Nevertheless, all the outliers we could manually detect were eliminated successfully. Figure 2 shows the detected inliers (marked as blue +'s), the outliers (red ×'s) and the winner quintuplet (cyan ◇'s) that gave the minimum $\hat{d}$ overlayed on the first and last images. Figure 3 shows the enlarged version near the top-right portion of the images where a few outlying features were present. Observations 127, 91, 128, 44, 51, 3 are all outliers that were correctly eliminated. Inlying observation 138, which was incorrectly classified as an outlier, corresponds to a feature point on a window that disappears in the image sequence due to occlusion. If the image sequence were to continue with the camera motion remained heading in the North direction, observations 138, 84, 127, etc. would disappear in the tracking.

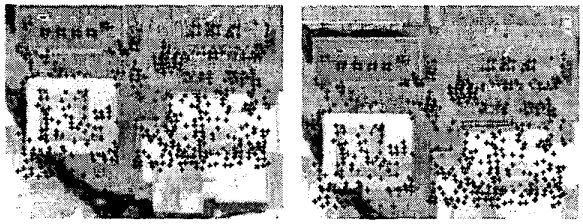

Figure 2. The tracked and inlying feature points in the first and last images.

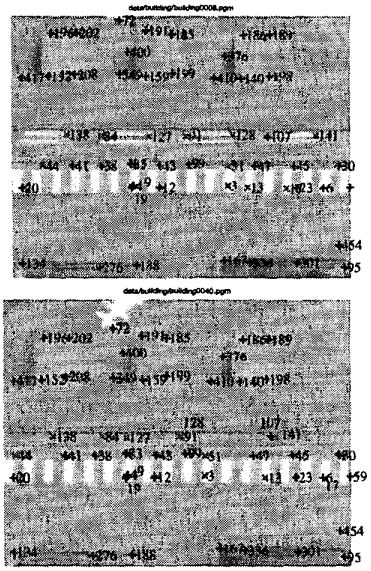

Figure 3. An enlarged portion of the images in Figure 2.

The second experiment involves a video sequence of an indoor scene taken by our Sony DCR-PC100 digital video camera. The images are of dimensions 768×576. A total of 141 frames were captured and 187 feature points were successfully tracked. 8 images (at 20 image frames apart) were selected from the video sequence, giving a 16×187 matrix for $\hat{\mathbf{x}}$. Figure 4 shows the first and last of the chosen images, with the tracked image feature points superimposed. The value of $\hat{d}$ and the robust standard deviation $\hat{\sigma}$ were computed to be 0.0178 and 0.0271, giving a threshold value t of 0.0542. This classified 179 of the observations to be inliers.

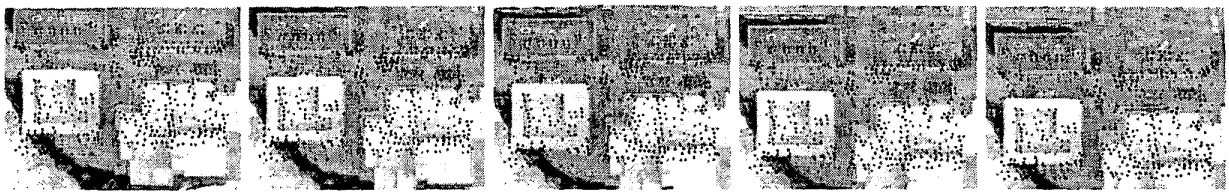

Figure 1. Five images of the building video sequence and the tracked image feature points.

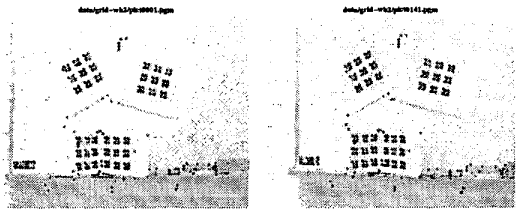

Figure 4. The first and last images of a video sequence of an indoor scene. The blue dots are the tracked image feature points.

Apart from localization errors of a few observations, this second image sequence contains no outliers. However, our method appears to work well in detecting erroneous coordinates due to poor localization (see Figure 5). Again, inliers are labelled as blue +'s, outliers as red ×'s, the winner quintuplet as cyan ◇'s.

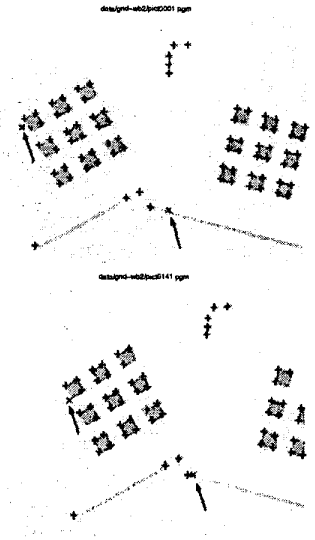

Figure 5. An enlarged portion of the images showing the detection of localization errors: two tracked image points were classified as outliers (pointed by the arrows).

6. Discussions

As with the RANSAC and LMedS methods implemented in [15, 19], extra computation is required in order to detect outliers. For each sample, one SVD is required to truncate the matrix $\check{\mathbf{x}}_k$ to rank 4. Another SVD is then required to truncate each of the n $\check{\mathbf{x}}_k^j$ matrices. To compute the distance of the subspaces spanned by columns of $\check{\mathbf{x}}_k$ and $\check{\mathbf{x}}_k^j$, one SVD is required (see Section 3). Thus, each sample requires $2n+1$ SVD computations. Since these SVD computations involve no more than 6 columns, a more economical method, such as the power method or the QR factorization incorporated in an iterative refinement process [3] could be employed [8]. If speed was a concern then the LMedS approach used in our method could also be replaced with RANSAC to achieve an early jump-out (see [19] for a discussion about RANSAC and LMedS).

Although video sequences have been used to test our method, corresponding image points found in discrete images could also be used to construct the $\hat{\mathbf{x}}$ matrix and this would be transparent to our method. To ensure that $\hat{\mathbf{x}}$ has more than 4 rows, 3 or more discrete images would be required. Our method can also be applied to each of the individual sections of a video sequence rather than to the entire video sequence.

In our random selection of samples, the first image was used as reference and was divided into disjoint buckets. A lower bound on image point separation was imposed to ensure that the observations in each sample were well distributed in the reference image. This approach is similar to that used in [19]. While it is desirable to have the matrix $\check{\mathbf{x}}_k$ to be of rank 4, a further drop in rank of the matrix is an indication of degeneracy of scene points in the sample. For instance, if the sample contains observations that are projections of 4 or more coplanar scene points then the fourth singular value of $\check{\mathbf{x}}_k$ would be negligible and the degeneracy could be easily identified and avoided. The detection of such degeneracy has been incorporated in our method.

A variation to our proposed method is to let $\check{\mathbf{x}}$ be a 4-column matrix and, for any observation $\hat{\mathbf{x}}^j$ (a one-dimensional $\Re^{2m}$ matrix), we compute the subspace distance between $\check{\mathbf{x}}$ and $\hat{\mathbf{x}}^j$. The quadruplet sample that has the minimum median subspace distance is finally selected as the winner. This alternative scheme requires choosing only 34 samples instead of 57 for $v = 99\%$. It can be used

in conjunction with ϕ_3 and is more computationally efficient.

An issue of concern is: How easy could this method be extended if the pin-hole camera model was used? It appears that there is no easy answer to this question. If the relative depths of all observations have been estimated then it may be worthwhile to single out those relative depths that show abrupt changes over the different image frames. The reason is that any sudden changes to the relative depths are likely to be due to tracking errors and thus possible outliers. How to automate this process and whether subspace distances can be applied are still subject of study by us.

Another issue of concern is: If $\check{\mathbf{x}}_k$ contained an outlying observation yet the rank-4 truncation in fact eliminated the distortion effect of the outlier, could the k^{th} sample that formed the matrix $\check{\mathbf{x}}_k$ be incorrectly identified as the winning sample? In our synthetic tests, this occurred only when we increased the localization errors and when the scene points were confined to be on only 2–4 planes. Thus, in real experiments, it is unlikely that a contaminated sample would be identified as the winner. Also, for long video sequences, the errors involved in outlying observations could be much larger than the size of the tracking window because of accumulation of tracking errors. This makes it easier to identify such outliers.

7. Conclusions

We have presented a novel method of detecting and eliminating outliers for the factorization approach under affine projection. The method employs a similarity function that measures the distances of the 4-dimensional subspaces spanned by the columns of the image measurement matrix and the LMedS criteria for automatic detection and elimination of outliers. The method has been tested on many synthetic test data and real video sequences with very promising results. The contribution of this research is to demonstrate the use of subspace distances for outlier elimination, giving a more accurate 3D reconstruction of the imaged scene.

References

[1] A. Björck and G. H. Golub. Numerical Methods for Computing Angles between Linear Subspaces. *Math. Comp.*, 27:579–594, 1973.

[2] M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the Association for Computing Machinery*, 24(6):381–395, Jun 1981.

[3] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins series in the mathematical sciences. The Johns Hopkins University Press, Baltimore and London, 3rd edition, 1996.

[4] A. Heyden. Projective Structure and Motion from Image Sequences using Subspace Methods. In *Scandinavian Conference on Image Analysis*, pages 1058–1063, 1997.

[5] D. Jacobs. Linear Fitting with Missing Data: Applications to Structure-from-Motionand its Characterizing Intensity Images. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 206–212, 1997.

[6] F. Kahl and A. Heyden. Affine Structure and Motion from Points, Lines and Conics. *International Journal of Computer Vision*, 33(3):163–180, 1999.

[7] B. D. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Int. Joint Conf. on Artificial Intelligence*, pages 674–679, 1981.

[8] T. Morita and T. Kanade. A Sequential Factorization Method for Recovering Shape and Motion from Image Streams. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(8):858–867, Aug 1997.

[9] J. L. Mundy and A. Zisserman. Appendix - Projective Geometry for Machine Vision. In J. L. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*, pages 463–534. The MIT Press, 1992.

[10] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, Inc., 1987.

[11] J. Shi and C. Tomasi. Good Features to Track. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[12] G. Sparr. Simultaneous Reconstruction of Scene Structure and Camera Locations from Uncalibrated Image Sequences. In *Proc. International Conference on Pattern Recognition*, volume 1, pages 328–333, 1996.

[13] P. Sturm and B. Triggs. A Factorization Based Algorithm for Multi-Image Projective Structure and Motion. In *Proc. European Conference on Computer Vision*, pages 709–720, Apr 1996.

[14] C. Tomasi and T. Kanade. Shape and Motion from Image Streams under Orthography: a Factorization Method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[15] P. Torr. *Motion Segmentation and Outlier Detection*. PhD thesis, Department of Engineering Science, University of Oxford, Dec 1995.

[16] P. H. S. Torr and D. W. Murray. The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix. *International Journal of Computer Vision*, 24(3):271–300, 1997.

[17] B. Triggs. The Geometry of Projective Reconstruction I: Matching Constraints and the Joint Image. Unpublished manuscript, 1995.

[18] Z. Zhang. Determining the Epipolar Geometry and its Uncertainty: A Review. *International Journal of Computer Vision*, 27(2):161–195, Mar 1998.

[19] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A Robust Technique for Matching Two Uncalibrated Images through the Recovery of the Unknown Epipolar Geometry. *Artificial Intelligence*, 75(1-2):87–120, 1995.