IEICE TRANS. INF. & SYST., VOL.E86–D, NO.6 JUNE 2003

PAPER

Outlier Removal for Motion Tracking by Subspace Separation

Yasuyuki SUGAYA† *and* **Kenichi KANATANI**†, *Regular Members*

SUMMARY Many feature tracking algorithms have been proposed for motion segmentation, but the resulting trajectories are not necessarily correct. In this paper, we propose a technique for removing outliers based on the knowledge that correct trajectories are constrained to be in a subspace of their domain. We first fit an appropriate subspace to the detected trajectories using RANSAC and then remove outliers by considering the error behavior of actual video tracking. Using real video sequences, we demonstrate that our method can be applied if multiple motions exist in the scene. We also confirm that the separation accuracy is indeed improved by our method.

key words: *feature tracking, outlier removal, subspace separation, robust estimation, RANSAC*

1. Introduction

Segmenting individual objects from backgrounds is one of the most important techniques of video processing. For images taken by a stationary camera, many segmentation algorithms based on interframe subtraction have been proposed. For images taken by a moving camera, however, the segmentation is very difficult because the objects and the backgrounds are both moving in the images.

While most existing methods for multi-body segmentation combine such information as optical flow, color, and texture along with miscellaneous heuristics, Costeira and Kanade [1] presented a segmentation algorithm based only on the image motion of feature points. Since then, various modifications and extensions of their method have been proposed.

Gear [3] used the reduced row echelon form and graph matching. Ichimura [5] applied the discrimination criterion of Otsu [18]. He also used the QR decomposition for feature selection [6]. Inoue and Urahama [9] introduced fuzzy clustering. Kanatani [12]–[14] introduced model selection and robust estimation based on a new interpretation of the Costeira-Kanade algorithm. Maki and Wiles [17] and Maki and Hattori [16] used Kanatani's method for analyzing the effect of illumination on moving objects. Wu, et al. [23] introduced orthogonal subspace decomposition. Sugaya and Kanatani [20] proposed model selection for automatic camera model selection.

Manuscript received June 24, 2002.
Manuscript revised November 14, 2002.

†The authors are with the Depertment of Information Technology, Okayama University, Okayama-shi, 700-8530 Japan.

For all these methods, two issues need to be resolved. One is the estimation of the number of independent motions. Many authors set an appropriate threshold for this, but it has been reported that estimating the number of motions is often more difficult than the segmentation itself [3]. To cope with this problem, the use of model selection criteria has been proposed [13], [15].

The other issue is the feature tracking. Most authors use the Kanade-Lucas-Tomasi algorithm [21], but the resulting trajectories are not always correct. In order to improve the tracking results, Ichimura and Ikoma [8] and Ichimura [7] introduced nonlinear filtering. Huynh and Heyden [4], motivated by 3-D reconstruction applications, showed that outlier trajectories in an image sequence of a static scene taken by a moving camera can be removed by fitting a 4-dimensional subspace to them by LMedS.

In this paper, we extend the idea of Huynh and Heyden [4] to multiple motions and introduce a more realistic criterion. Adopting Kanatani's geometric interpretation [12]–[14], we fit an appropriate subspace to the detected trajectories using RANSAC and remove outliers by considering the error behavior of actual video tracking.

Section 2 summarizes the subspace constraint introduced by Kanatani [12], [13]. Sections 3 and 4 describe our procedure. In Sec. 5, we show real video examples and demonstrate that our method is superior to the method of Huynh and Heyden [4] and can be applied if multiple motions exist in the scene. We also confirm that the separation accuracy is indeed improved by our method. Section 6 gives our conclusion.

2. Subspace Constraint

We track N rigidly moving feature points over M frames and let $(x_{\kappa\alpha}, y_{\kappa\alpha})$ be the image coordinates of the αth point in the κth frame. We stack all the image coordinates vertically and represent the entire trajectory by the following $2M$-dimensional *trajectory vector*:

$$\boldsymbol{p}_\alpha = (x_{1\alpha}\ y_{1\alpha}\ x_{2\alpha}\ y_{2\alpha} \cdots\ x_{M\alpha}\ y_{M\alpha})^\top. \tag{1}$$

Regarding the XYZ camera coordinate system as the world coordinate system, we fix a 3-D object coordinate system to the moving object. Let $\boldsymbol{t}_\kappa$ and $\{\boldsymbol{i}_\kappa, \boldsymbol{j}_\kappa, \boldsymbol{k}_\kappa\}$ be, respectively, its origin and 3-D orthonormal basis in the κth frame. If we let $(a_\alpha, b_\alpha, c_\alpha)$ be the 3-D object coordinates of the αth point, its 3-D position in the κth frame is

$$\boldsymbol{r}_{\kappa\alpha} = \boldsymbol{t}_\kappa + a_\alpha \boldsymbol{i}_\kappa + b_\alpha \boldsymbol{j}_\kappa + c_\alpha \boldsymbol{k}_\kappa \tag{2}$$

with respect to the world coordinate system.

If an affine camera model (e.g., orthographic, weak perspective, or paraperspective projection) is assumed, the 2-D position of $\boldsymbol{r}_\alpha$ in the image is given by

$$\begin{pmatrix} x_{\kappa\alpha} \\ y_{\kappa\alpha} \end{pmatrix} = \boldsymbol{A}_\kappa \boldsymbol{r}_{\kappa\alpha} + \boldsymbol{b}_\kappa, \tag{3}$$

where $\boldsymbol{A}_\kappa$ and $\boldsymbol{b}_\kappa$ are, respectively, a 2×3 matrix and a 2-dimensional vector determined by the position and the orientation of the camera and its internal parameters in the κth frame. From Eq. (2), we can write Eq. (3) as

$$\begin{pmatrix} x_{\kappa\alpha} \\ y_{\kappa\alpha} \end{pmatrix} = \tilde{\boldsymbol{m}}_{0\kappa} + a_\alpha \tilde{\boldsymbol{m}}_{1\kappa} + b_\alpha \tilde{\boldsymbol{m}}_{2\kappa} + c_\alpha \tilde{\boldsymbol{m}}_{3\kappa}, \tag{4}$$

where $\tilde{\boldsymbol{m}}_{0\kappa}, \tilde{\boldsymbol{m}}_{1\kappa}, \tilde{\boldsymbol{m}}_{2\kappa}$, and $\tilde{\boldsymbol{m}}_{3\kappa}$ are 2-dimensional vectors determined by the position and orientation of the camera and its internal parameters in the κth frame. From Eq. (4), the trajectory vector $\boldsymbol{p}_\alpha$ in Eq. (1) can be written in the form

$$\boldsymbol{p}_\alpha = \boldsymbol{m}_0 + a_\alpha \boldsymbol{m}_1 + b_\alpha \boldsymbol{m}_2 + c_\alpha \boldsymbol{m}_3, \tag{5}$$

where $\boldsymbol{m}_0$, $\boldsymbol{m}_1$, $\boldsymbol{m}_2$ and $\boldsymbol{m}_3$, are the $2M$-dimensional vectors obtained by stacking $\tilde{\boldsymbol{m}}_{0\kappa}, \tilde{\boldsymbol{m}}_{1\kappa}, \tilde{\boldsymbol{m}}_{2\kappa}$, and $\tilde{\boldsymbol{m}}_{3\kappa}$ vertically over the M frames, respectively.

Equation (5) implies that the trajectory vectors for the same object are constrained to be in the 4-dimensional subspace spanned by $\{\boldsymbol{m}_0, \boldsymbol{m}_1, \boldsymbol{m}_2, \boldsymbol{m}_3\}$. Huynh and Heyden [4] proposed a procedure for removing outlier trajectories from an image sequence of a static scene taken by a moving camera. They fitted a 4-dimensional subspace to the trajectories by LMedS and removed outliers using a criterion introduced for mathematical convenience; not much consideration was given to real video processing characteristics.

In this paper, we introduce a statistically consistent criterion by considering the error behavior of actual video tracking. We also extend our method to multiple motions.

3. Outlier Removal Procedure

We assume that the maximum number m of independent motions in the scene is known. Assuming too large a number m is likely to deteriorate the performance of our algorithm, but we do not go into the details, which involve a lot of subtleties [15]. In the following, we are mainly concerned with the case for $m = 1$ or 2, which occurs in most practical applications (though theoretically m can be any number).

We note that if m motions exist, the trajectory vectors $\{\boldsymbol{p}_\alpha\}$ should belong to a $4m$-dimensional subspace. Exploiting this knowledge, we fit a $4m$-dimensional subspace to the detected trajectory vectors using RANSAC [2], [10] and remove outliers by setting an appropriate threshold for the residual (Fig. 1).

In order that a $4m$-dimensional subspace can be fitted, we assume that more than $4m$ feature points are tracked throughout the sequence. Let $n = 2M$ and $d = 4m$. Our procedure is as follows:

1. Randomly choose d vectors $\boldsymbol{q}_1$, $\boldsymbol{q}_2$, …, $\boldsymbol{q}_d$ from $\{\boldsymbol{p}_\alpha\}$, $\alpha = 1, \ldots, N$.
2. Define an $n \times n$ matrix
$$\boldsymbol{M}_d = \sum_{i=1}^{d} \boldsymbol{q}_i \boldsymbol{q}_i^\top. \tag{6}$$
3. Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$ be the d eigenvalues of matrix $\boldsymbol{M}_d$, and $\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_d\}$ the orthonormal system of corresponding eigenvectors.
4. Define an $n \times n$ projection matrix
$$\boldsymbol{P}_{n-d} = \boldsymbol{I} - \sum_{i=1}^{d} \boldsymbol{u}_i \boldsymbol{u}_i^\top. \tag{7}$$
5. Let S be the number of points $\boldsymbol{p}_\alpha$ that satisfy
$$\|\boldsymbol{P}_{n-d}\boldsymbol{p}_\alpha\|^2 < (n-d)\sigma^2, \tag{8}$$
where $\|\boldsymbol{P}_{n-d}\boldsymbol{p}_\alpha\|^2$, which we call the *residual*, is the squared distance of point $\boldsymbol{p}_\alpha$ from the fitted d-dimensional subspace. The constant σ is an estimate of the noise standard deviation.
6. Repeat the above procedure a sufficient number of times†, and determine the projection matrix $\boldsymbol{P}_{n-d}$ that maximizes S.
7. Remove those $\boldsymbol{p}_\alpha$ that satisfy
$$\|\boldsymbol{P}_{n-d}\boldsymbol{p}_\alpha\|^2 \geq \sigma^2 \chi^2_{n-d;99}, \tag{9}$$
where $\chi^2_{r;a}$ is the ath percentile of the χ^2 distribution with r degrees of freedom.

If the noise in the coordinates of the feature points is an independent Gaussian random variable of mean 0 and standard deviation σ, the residual $\|\boldsymbol{P}_{n-d}\boldsymbol{p}_\alpha\|^2$

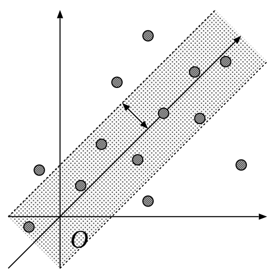


Fig. 1 Removing outliers by fitting a subspace.

†In our experiment, we stopped if S did not increase 200 times consecutively.

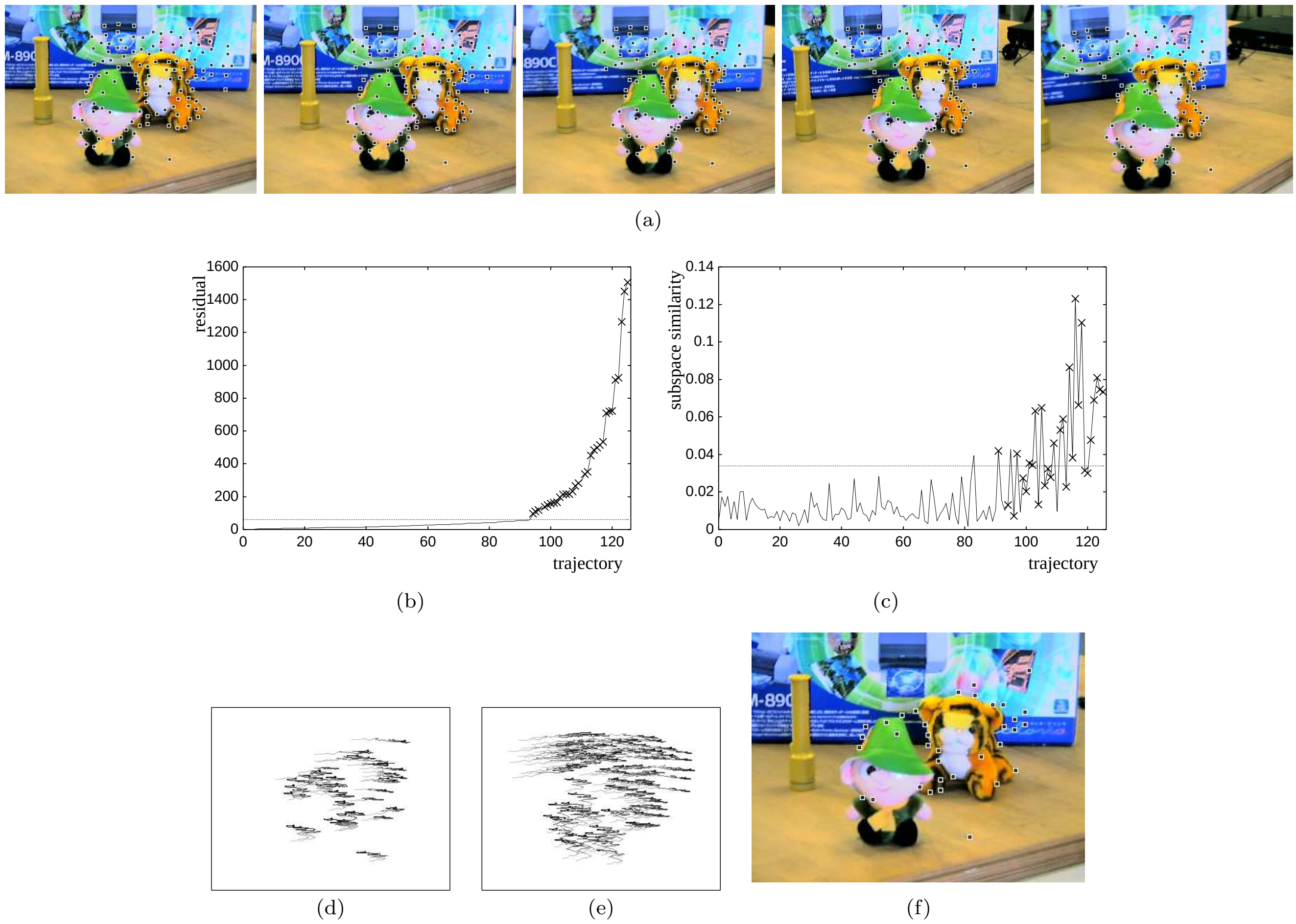


Fig. 2 (a) Five decimated frames from a 100 frame sequence of a static scene with 126 feature points successfully tracked. (b) The residuals of the trajectories ($\times$ indicates apparently incorrect trajectories). (c) The subspace similarity of Huynh and Heyden [4] ($\times$ corresponds to the trajectories with $\times$ in (b)). (d) The trajectories of detected outliers. (e) The trajectories of detected inliers. (f) The detected outlier locations.

divided by σ^2 should be subject to a χ^2 distribution with $n-d$ degrees of freedom. Hence, its expectation is $(n-d)\sigma^2$, provided $\boldsymbol{p}_\alpha$ is an inlier. The above procedure effectively fits a d-dimensional subspace that maximizes the number of the trajectories whose residuals are smaller than $(n-d)\sigma^2$. After determining the subspace, we remove those trajectories which cannot be regarded as inliers with significance level 1%.

Huynh and Heyden [4] used a similarity measure between the subspace defined by inliers and the subspace that may contain outliers. Their measure was introduced for mathematical convenience without much consideration about statistic characteristics of real video tracking. Then, they used LMedS and applied the criterion of Rousseeuw and Leroy [19], which was derived by assuming that the inlier noise is Gaussian. No justification was given to that assumption.

Our χ^2-based method is statistically consistent, reflecting the real image noise characteristics. However, the crucial element is a realistic choice of the threshold σ, for which we need to observe the noise behavior in real video tracking. We describe this in the next section.

4. Experiments

Figure 2(a) shows five frames decimated from a 100 frame sequence (320 $\times$ 240 pixels) of a static scene taken by a moving camera. We tracked 126 points as indicated by the symbol $\square$ in the images.

Figure 2(b) plots the residuals of the 126 trajectories; they are enumerated on the horizontal axis in the order of the residual. We visually inspected all the trajectories frame by frame to see if they are really correct. The trajectories we found incorrect are marked with $\times$ in the plot. From Fig. 2(b), we see that the value σ = 0.5 can correctly separate incorrect trajectories from correct ones. The horizontal line indicates the threshold determined by Eq. (9).

For comparison, we applied the method of Huynh

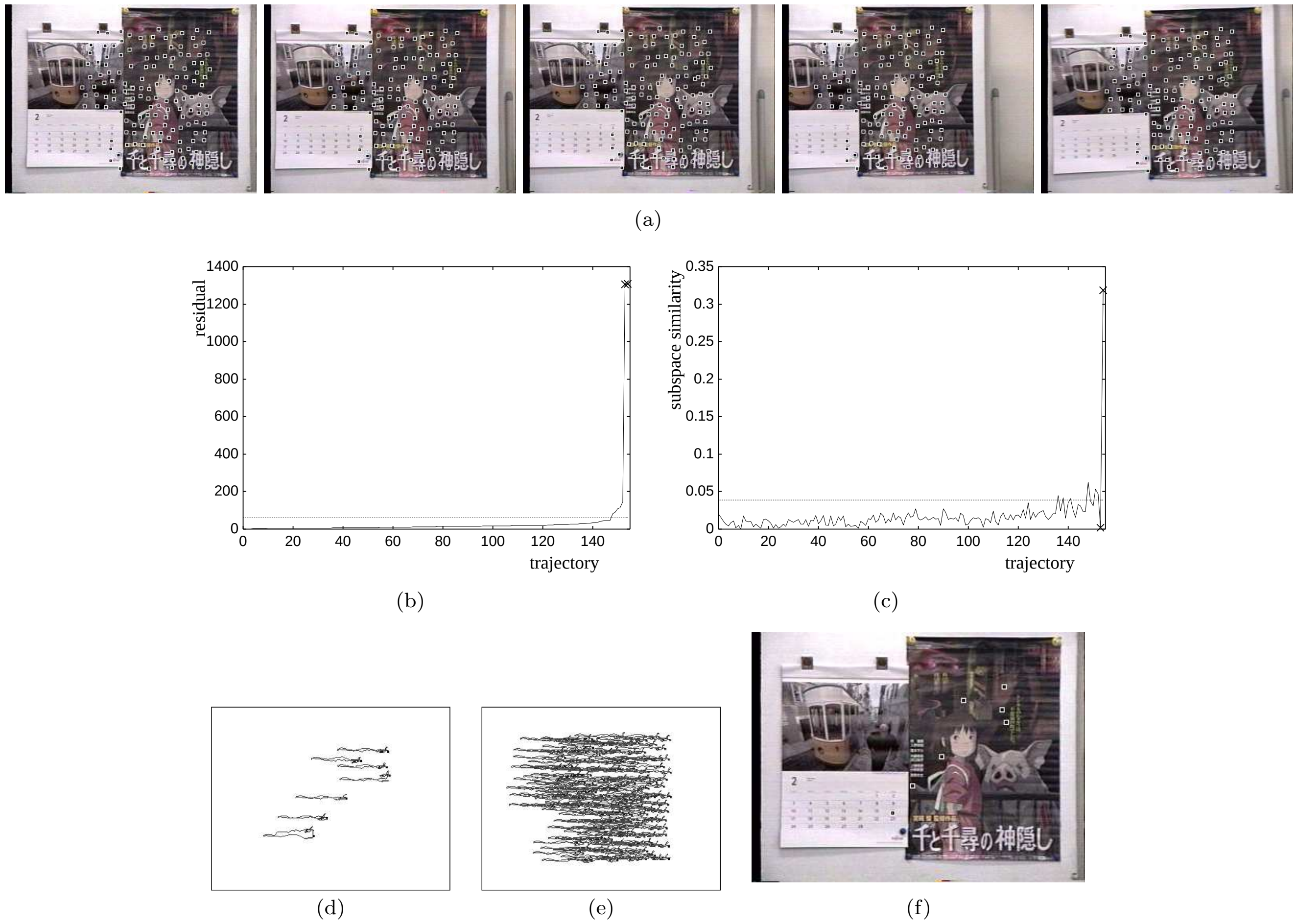


Fig. 3 (a) Five decimated frames of a 100 frame sequence of a static scene with 155 feature points successfully tracked. (b) The residuals of the trajectories (× indicates apparently incorrect trajectories). (c) The subspace similarity of Huynh and Heyden [4] (× corresponds to the trajectories with × in (b)). (d) The trajectories of detected outliers. (e) The trajectories of detected inliers. (f) The detected outlier locations.

and Heyden [4] to the same data. Figure 2(c) plots the subspace similarity resulting from the separation procedure of Huynh and Heyden [4]. The numbering of the trajectories is the same as in Fig. 2(a). The horizontal line in the figure indicates the LMedS-based threshold used by Huynh and Heyden [4].

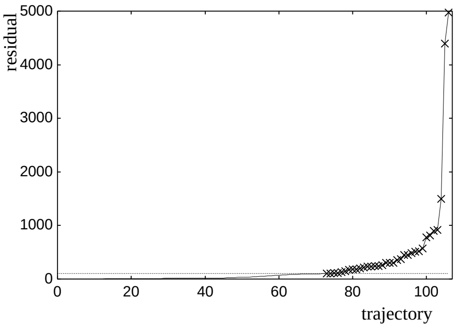
We see that many trajectories with large residuals have very small subspace similarity. As a result, some incorrect trajectories are classified as inliers. We conclude that the subspace similarity of Huynh and Heyden [4] is not appropriate for this example.

Figures 2(d) and (e) show the trajectories of the outliers and inliers, respectively, detected by our method. Figure 2(f) shows the locations of the outliers in the first frame. We find that many of them are on the occluding contours of objects. We also find that some correct trajectories are also rejected as outliers. A close examination revealed that they correspond to points fluctuating around their expected positions by a few pixels throughout the sequence. In practice, removing them is a reasonable choice, since inclusion of such unreliable trajectories would lower the reliability of the subsequent segmentation or 3-D reconstruction.

Figure 3(a) shows another sequence of a static scene. The results are arranged in the same way in Figs. 3(b)–(f). We tracked 155 feature points over 100 frames. From Fig. 3(b), we see that the value $\sigma = 0.5$ can detect conspicuous outliers, although some of unstable inliers are also removed. Figure 3(c) shows that one conspicuous outlier with a very large residual has a small subspace similarity. As a result, it is incorrectly classified to be an inlier by the method of Huynh and Heyden [4]. For this sequence, the number of outliers is relatively small, probably because the scene is a planar surface without occluding contours.

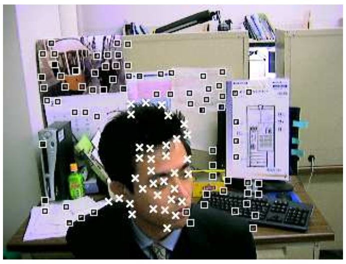
In the sequence shown in Fig. 4(a), an object (a human body) is moving independently of the background, which is also moving in the images. Figure 4(b) shows the residuals of the 107 feature points successfully tracked over 100 frames. Again, the value $\sigma =$

(a)

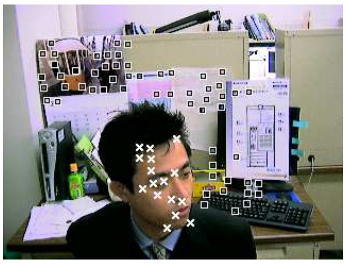


(b)

(c)



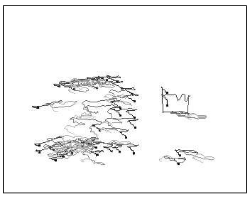
(d)



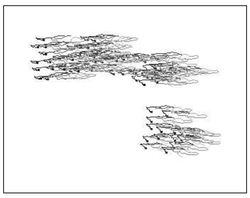
(e)

	correctness	details
with outliers	67.3%	15/40, 57/67
without outliers	100.0%	16/16, 57/57

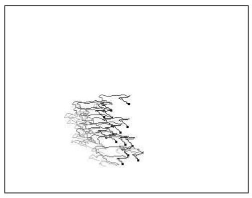
(f)



(g)



(h)



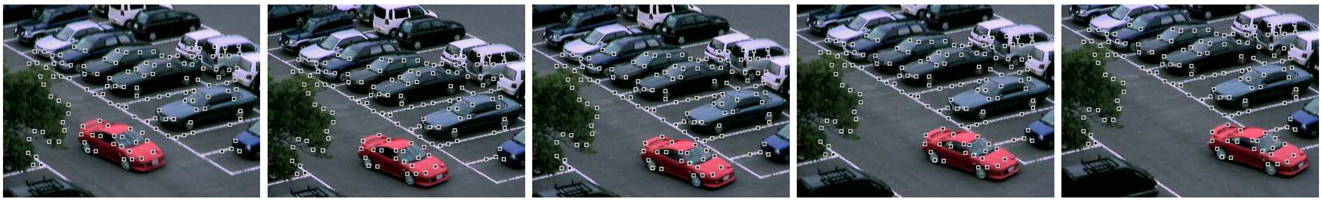
(i)

Fig. 4 (a) Five decimated frames of a 100 frame image sequence of a static scene and a moving object with 107 feature points successfully tracked. (b) The residuals of the trajectories ($\times$ indicates apparently incorrect trajectories). (c) The locations of the outliers in the first frame. (d) The segmentation with outliers ($\times$ for object points; $\square$ for background points). (e) The segmentation without outliers ($\times$ for object points; $\square$ for background points). (f) The correctness of segmentation and the classification details. (g) The trajectories of detected outliers. (h) The trajectories of detected background points. (i) The trajectories of detected object points.

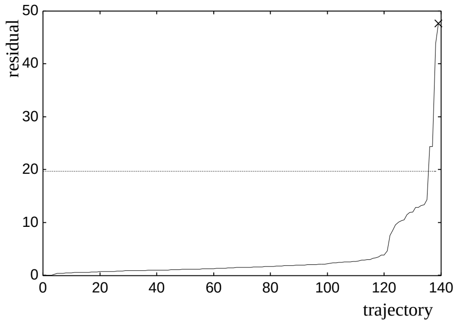
0.5 can correctly separate outliers from inliers. We confirmed by visual inspection that the rejected trajectories are all incorrect and the remaining ones are all correct. Figure 4(c) shows the locations of the detected outliers in the first frame. As can be seen, many of them are on the occluding contours of the moving object.

In order to see the effect of outliers on segmentation, we applied the subspace separation algorithm† of Kanatani [12], [13] to this sequence. Figure 4(d) shows the segmentation result without removing outliers; Fig. 4(e) shows the result after removing outliers. The symbol $\square$ indicates points classified to the back-

†The source program is publicly available from: `http://www.suri.it.okayama-u.ac.jp/e-program.html`

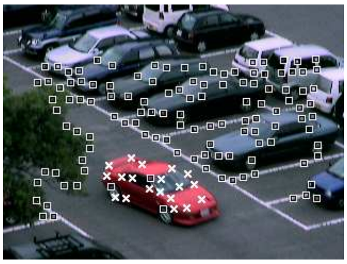


(a)

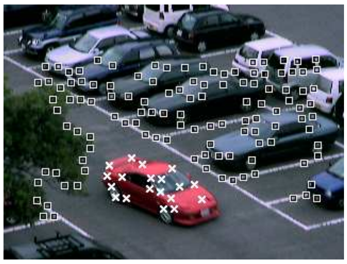


(b)

(c)



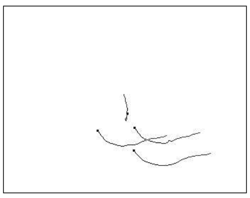
(d)



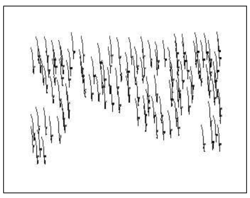
(e)

	correctness	details
with outliers	97.1%	18/22, 118/118
without outliers	100.0%	19/19, 117/117

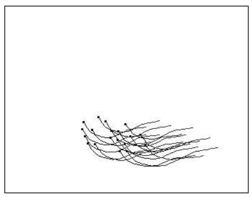
(f)



(g)



(h)



(i)

Fig. 5 (a) Five decimated frames of a 30 frame image sequence of a static scene and a moving object with 140 feature points successfully tracked. (b) The residuals of the trajectories (× indicates apparently incorrect trajectories). (c) The locations of the outliers in the first frame. (d) The segmentation with outliers (× for object points; □ for background points). (e) The segmentation without outliers (× for object points; □ for background points). (f) The correctness of segmentation and the classification details. (g) The trajectories of detected outliers. (h) The trajectories of detected background points. (i) The trajectories of detected object points.

ground; the symbols × indicates points classified to the moving object. In Fig. 4(d) some inliers are incorrectly classified, while in Fig. 4(e) all inliers are correctly classified.

In the table in Fig. 4(f), the second column lists the correctness of the segmentation: (the number of correctly classified trajectories)/(the total number of trajectories) in percentage for Figs. 4(d) and (e), respectively. The third column lists (the number of correct object points)/(the number of points classified to the object) and (the number of correct background points)/(the number of points classified to the background). Figures 4(g), (h), and (i) show, respectively, the trajectories of the detected outliers, the inliers classified to the background, and the inliers classified to the moving object.

Table 1 Computation time (sec).

	Fig. 2	Fig. 3	Fig. 4	Fig. 5
Number of frames	100	100	100	30
Number of points	126	155	107	140
Computation time	33.17	36.83	32.57	1.95

Figure 5 shows another example similarly arranged. We successfully tracked 140 feature points over 30 frames. In this case, only one outlier can be found by visual inspection. Figure 5(b) shows that the value $\sigma = 0.5$ can detect that outlier, although some unstable trajectories are also removed. We can see that some inliers are incorrectly classified when outliers are included as shown in Fig. 5(d), while all inliers are correctly classified after outliers are removed as shown in Fig. 5(e).

Figure 5(b) clearly indicates that correct trajectories consist of those with very small residuals and those with relatively large residuals. This clear distinction implies that the detected feature points are divided into two types: unambiguous and ambiguous. An unambiguous point is correctly tracked throughout the sequence, while an ambiguous point is always ambiguous in the course of the tracking. This phenomenon can be observed more or less in all the previous examples but is particularly strong for this sequence. This is probably because the scene is very far away and the range of the gray levels is relatively narrow.

This is also the reason why the threshold cannot be set automatically as Huynh and Heyden [4] did. If the noise in the coordinates of the feature points were Gaussian and independent for each point and each frame, we could use LMedS [19] and estimate the noise level σ from the estimated median (although we cannot use the formula of Rousseeuw and Leroy [19], because the residual is subject to a χ^2, not Gaussian, distribution). In reality, however, it is difficult to set the threshold automatically because of to the existence of strong temporal correlations.

We adopted the value $\sigma = 0.5$ from careful observations of the noise characteristics of actual video tracking. We have confirmed that this value can work very well for all image sequences that we tested including those not shown here.

The computation time for the examples of Figs. 2, 3, 4, and 5 is summarized in Table 1. It mainly depends on the number of frames. We used Pentium IV 1.8GHz for the CPU and Linux for the OS.

5. Concluding Remarks

In this paper, we have proposed a technique for removing outliers from the trajectories of feature points detected over a video sequence. Our algorithm fits a subspace to the trajectories by RANSAC and removes outliers by considering the error behavior of actual video tracking. Using real video sequences, we have demonstrated that our method is superior to the method of Huynh and Heyden [4] and can also be applied if multiple motions exist in the scene. We have also confirmed that the separation accuracy is indeed improved by our method.

Our method is based on an affine camera model. Also, feature points must be tracked throughout the sequence. These limit the use of our method to a relatively short sequence of images. For a long sequence, we must divide it into overlapping segments and apply our method separately. The affine camera model is a good approximation only when the depth of the scene does not vary very much. How to cope with strong perspective effects, if they exist, is left for future research.

Our approach is based on the *geometric* constraint that the image motion should be interpreted to be rigid motions in the scene. In contrast, the use of nonlinear filtering proposed by Ichimura [7] and Ichimura and Ikoma [8] is based on the *stochastic* constraint that the image motion should be "smooth" with a strong temporal coherence. Since these two approaches are complementary in nature, it is expected that the segmentation accuracy will be further increased by combining them.

Acknowledgments

This work was supported in part by the Ministry of Education, Culture, Sports, Science and Technology, Japan, under a Grant in Aid for Scientific Research C(2) (No. 15500113), the Support Center for Advanced Telecommunications Technology Research, and Kayamori Foundation of Informational Science Advancement.

References

[1] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," Int. J. Comput. Vision, vol.29, no.3, pp.159–179, Sept. 1998.
[2] M. A. Fischer and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," Commun. ACM, vol.24, no.6, pp.381–395, June 1981.
[3] C. W. Gear, "Multibody grouping from motion images," Int. J. Comput. Vision, vol.29, no.2, pp.133–150, Aug./Sept. 1998.
[4] D. Q. Huynh and A. Heyden, "Outlier detection in video sequences under affine projection," Proc. IEEE Conf. Computer Vision Pattern Recog., vol.1, pp.695–701, Kauai, Hawaii, U.S.A., Dec. 2001.
[5] N. Ichimura, "Motion segmentation based on factorization method and discriminant criterion," Proc. 7th Int. Conf. Comput. Vision, vol.1, pp.600–605, Kerkyra, Greece, Sept. 1999.
[6] N. Ichimura, "Motion segmentation using feature selection and subspace method based on shape space," Proc. 15th Int. Conf. Pattern Recog., vol.3, pp.858–864, Barcelona, Spain, Sept. 2000.
[7] N. Ichimura, "Stochastic filtering for motion trajectory in image sequences using a Monte Carlo filter with estimation of hyper-parameters," Proc. 16th Int. Conf. Pattern Recog., vol.4, pp.68–73, Quebec City, Canada, Aug. 2002.
[8] N. Ichimura and N. Ikoma, "Filtering and smoothing for motion trajectory of feature point using non-gaussian state space model," IEICE Trans. Inf. & Syst., vol.E84-D, no.6,

pp.755–759, June 2001.
[9] K. Inoue and K. Urahama, "Separation of multiple objects in motion images by clustering," Proc. 8th Int. Conf. Comput. Vision, vol.1, pp.219–224, Vancouver, Canada, July 2001.
[10] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, Cambridge, U.K., 2000.
[11] K. Kanatani, Statistical Optimization for Geometric Computation: Theory and Practice, Elsevier, Amsterdam, 1996.
[12] K. Kanatani, "Motion segmentation by subspace separation and model selection," Proc. 8th Int. Conf. Comput. Vision, vol.2, pp.301–306, Vancouver, Canada, July 2001.
[13] K. Kanatani, "Motion segmentation by subspace separation: Model selection and reliability evaluation," Int. J. Image Graphics, vol.2, no.2, pp.179–197, April 2002.
[14] K. Kanatani, "Evaluation and selection of models for motion segmentation," Proc. 7th Euro. Conf. Comput. Vision, vol.3, pp.335–349, Copenhagen, Denmark, June 2002.
[15] K. Kanatani and C. Matsunaga, "Estimating the number of independent motions for multibody segmentation," Proc. 5th Asian Conf. Comput. Vision, vol.1, pp.7–12, Melbourne, Australia, Jan. 2002.
[16] A. Maki and K. Hattori, "Illumination subspace for multibody motion segmentation," Proc. IEEE Conf. Comput. Vision Pattern Recog., vol.2, pp.11–17, Kauai, Hawaii, U.S.A., Dec. 2001.
[17] A. Maki and C. Wiles, "Geotensity constraint for 3D surface reconstruction under multiple light sources," Proc. 6th Euro. Conf. Comput. Vision, vol.1, pp.725–741, Dublin, Ireland, June/July 2000.
[18] N. Otsu, "A threshold selection method from gray-level histograms," IEEE Trans. Sys. Man Cyber., vol.9, no.1, pp.62–66, Jan. 1979.
[19] P. J. Rousseeuw and A. M. Leroy, Robust Regression and Outlier Detection, Wiley, New York, 1987.
[20] Y. Sugaya and K. Kanatani, "Automatic camera model selection for multibody motion segmentation," Proc. IAPR Workshop Machine Vision Applications, pp.412–415, Tokyo, Japan, Dec. 2002.
[21] C. Tomasi and T. Kanade, Detection and Tracking of Point Features, CMU Tech. Rep. CMU-CS-91-132, April 1991; `http://vision.stanford.edu/~birch/klt/`.
[22] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography—A factorization method," Int. J. Comput. Vision, vol.9, no.2, pp.137–154, Nov. 1992.
[23] Y. Wu, Z. Zhang, T. S. Huang, and J. Y. Lin, "Multibody grouping via orthogonal subspace decomposition, sequences under affine projection," Proc. IEEE Conf. Computer Vision Pattern Recog., vol.2, pp.695–701, Kauai, Hawaii, U.S.A., Dec. 2001.

Yasuyuki Sugaya received his M.S. and Ph.D. from the University of Tsukuba, Ibaraki, Japan, in 1998 and 2001, respectively. He is currently Assistant Professor of information technology at Okayama University, Okayama, Japan. His research interests include image processing and computer vision.

Kenichi Kanatani received his M.S. and Ph.D. in applied mathematics from the University of Tokyo in 1974 and 1979, respectively. After serving as Professor of computer science at Gunma University, Gunma, Japan, he is currently Professor of information technology at Okayama University, Okayama, Japan. He is the author of many books on computer vision. He is an IEEE Fellow.