

可変楕円モデルを用いた K-means トラッキング

華 春 生[†] 和 田 俊 和[†] 吳 海 元[†]

我々は、「K-means トラッキング」と名づけた対象追跡の方法を提案している。この方法は追跡対象の領域内に背景を混入しても頑健性を持つ特徴がある。一方、処理速度や、ズームと変形などさらなる安定性の問題が残っている。本論文では、これらの問題を解決するために、一つのターゲット中心点と、そのターゲット周辺の非ターゲットクラスタ中心を表す可変楕円モデルから構成されるトラッキング方法を提案する。本提案手法の特徴として：1) ターゲットのクラスタ中心は楕円の中心位置の画素とし、非ターゲットのクラスタ中心は可変楕円周上の画素から選定し、K-means クラスタリングによって楕円モデル内の各画素をターゲットと非ターゲットに分けながら追跡する。これによって、距離の計算回数を大幅に削減できると同時にターゲット検出の安定性も向上できる。2) ターゲットの検出結果に基づいて可変楕円モデルのパラメータを画像フレームごとに更新する。これによって、ターゲットの大きさや形変化に対してロバストに追跡できるようになる。元手法との比較実験を通じて、本提案手法が速度ならびに安定性の両面ともに大きく改善できることが確認された。

K-means tracking with variable ellipse model

CHUNSHENG HUA,[†] TOSHIKAZU WADA[†] and HAIYUAN WU[†]

We have proposed a K-means clustering based target tracking method, which is robust against background involution comparing with the target template. This paper presents a new method for solving the drawbacks of the previous method, i.e., speed, stability, target appearance change and size change. Our new tracking method consists of a single target point, and a variable ellipse model for representing non-target pixels. The contributions of our new method are : 1) The original K-means clustering is reduced to 2-means clustering, i.e., target and non-target clusters, and the non-target cluster center is adaptively picked up from the pixels on the ellipse circumference. This modification reduces the number of distance computation and improves the stability of the target detection as well. 2) The ellipse parameters are adaptively controlled according to the target detection result. This adaptation improves the robustness against the scale and shape changes of the target. Through the various experiments, we confirmed that our new method improves speed and robustness of the original method.

1. Introduction

Recently, object tracking has emerged as an important part of the vision-based system. It is used in many applications such as video surveillance, robot controlling, security system, etc. The main purpose of object tracking is to compute the position of the target object in each image sequence. Some techniques have been reported in object tracking, such as template matching, kernel-based object tracking, active contours, and condensation, etc.

Because of its high-speed computation, template matching method¹⁾²⁾ can be suitable for high-speed object tracking, and to solve the problem that the object shape may change while tracking. Another method called

flexible template matching has been introduced recently. When processing a new input image, the template will change according to the shape of the target object. But as we all know, template matching is sensitive to illumination changes, and template matching method is also easy to fail, if the background involution happens¹⁾.

To solve the problem above, Comaniciu has brought out a Kernel-based non-rigid object tracking method⁹⁾. In their method the target location was formulated by using the basin of attraction of the local maxima and they used the mean shift procedure to optimize the similarity measure. But when the color distribution of the target object is very narrow (the target object has only one color), this method may fail. And if the target object was lost, it could not correct itself.

The other group of algorithms based on active contours³⁾ is suitable for tracking, but before the tracking

[†] 和歌山大学
Wakayama University

procedure is initiated the contour of the object must be known.

Condensation (also called particle filter, or MCMC method⁽⁷⁾⁽⁸⁾⁽¹⁰⁾⁽¹¹⁾⁽¹³⁾⁽¹⁴⁾) has been popular in recent years. Because of using the parameters vectors and statistic calculation, this method can work quite well. But it works hardly when scales or the dimensionality increases.

Compared with the model-based method, data-driven object tracking is important for vision application, because it doesn't need any previous knowledge about the object to be tracked. Heisele has brought out a parallel K-means method⁽⁴⁾⁽⁵⁾ to track the target object in the color sequence. Heisele also used the color/position vector in the feature space, and after the clustering procedure he used the Kalman filter to predict the prototype for the next frame. Since he only used the information of the target object, if the background involution had happened that method might fail. Besides that, that method was impossible to be used in the real-time tracking. Agarwal used a K-means based autoregressive model⁽¹²⁾ to track the object, and because they only used the target information too, we wonder about that method can work robustly.

Our group has brought out a *K-means tracking* method⁽⁶⁾. The key idea is tracking a given target by clustering the image pixels into target and non-target pixels using K-means clustering. Since the method only tracks the target cluster, it is robust against the background involution. But it could not scale, target appearance change, and takes too much calculation time.

To solve that problem, in this paper, we bring out an improvement version. Our new method consists of a single point that is the pixel of target cluster center, and a variable ellipse model for representing the pixels of non-target cluster centers. The main contributions are: 1) The original K-means clustering is reduced to 2-means clustering, i.e., target and non-target clusters, and the non-target cluster center is adaptively picked up from the pixels on the ellipse circumference. This modification reduces the number of distance computation and improves the stability of the target detection as well. 2) The ellipse parameters are adaptively controlled according to the target detection result. This adaptation improves the robustness against the scale and shape changes of the target. The frame work of this version is show in figure 1.

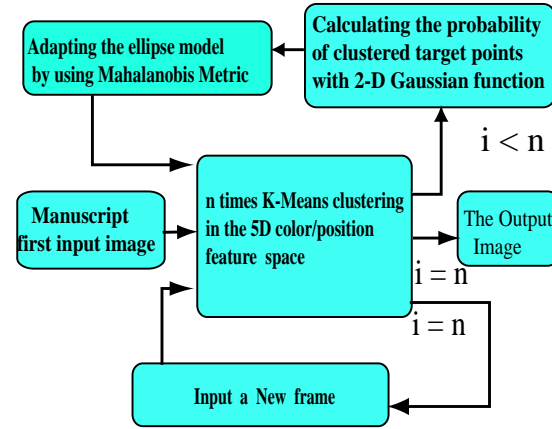


图 1 The overview of our new research

2. A review of our previous work

To track a moving object robustly even if the background involution had happened, our group has presented a *K-means tracking* method⁽⁶⁾, which divides the pixels of an input image into tracking object and non-tracking objects by using the K-means clustering algorithm⁽¹⁵⁾⁽¹⁶⁾. For clustering, each pixel $p(x, y)$ is represented by a five-dimensional feature vector $\mathbf{v}_{x,y} = (Y_{x,y}, U_{x,y}, V_{x,y}, x, y)$, which contains its color in the *YUV*-space and position in the image, here x is the row and y is the column. Therefore, as shown in figure 2, the feature vector of the center of target points is $\mathbf{v}_T = (Y_T, U_T, V_T, x_T, y_T)$, and the centers of none-target points are $\mathbf{v}_{nk} = (Y_{nk}, U_{nk}, V_{nk}, x_{nk}, y_{nk})$. Here, $k = 1 \sim N$ and N is the number of the non-target centers.

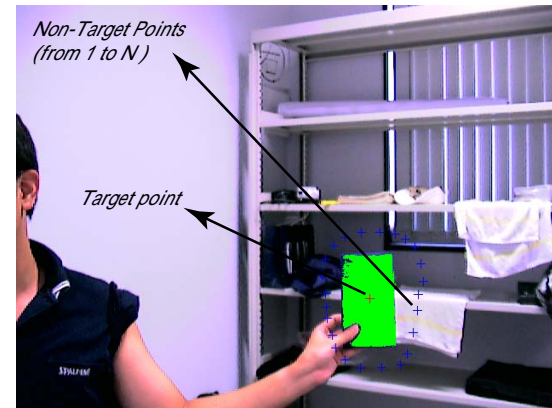


图 2 A sample of the old method

In this method⁶⁾, the distance from an unknown pixel $p(x, y)(t)$ in an input image taken at t time to the target center $d_T(t)$, and non-target centers $d_{NT}(t)$ are following, respectively:

$$d_T(t) = \|\mathbf{v}_{i,j}(t) - \mathbf{v}_T(t)\|^2, \quad (1)$$

$$d_{NT}(t) = \underset{k=1 \sim N}{\operatorname{argmin}} \|\mathbf{v}_{x,y}(t) - \mathbf{v}_{nk}(t)\|^2, \quad (2)$$

so the clustering procedure produces a set of target feature vectors $T(t)$ as :

$$T(t) = 1 \quad \text{if} \quad \{d_T(t) < d_{NT}(t)\}. \quad (3)$$

Therefore, if the target object is lost at $t + 1$ time, we can use the following method to realize self-correcting.

$$T(t+1) = 1 \quad \text{if} \quad \|\mathbf{v}_{x,y}(t+1) - \mathbf{v}_T(t)\|^2 < \underset{n=1 \sim N}{\operatorname{argmin}} \|\mathbf{v}_{x,y}(t+1) - \mathbf{v}_n(t)\|^2. \quad (4)$$

The advantages of this research are: 1) Because of using feature vectors of both target point (positive data) and non-target points (negative data), it is robust against background involution. This is one of the most important differences between our research and Heisele⁴⁾⁵⁾. 2) With equation (4), our method can achieve self-correction, that will ensure us to track the target object more robustly.

But there are still several drawbacks of our previous method: 1) The processing result is heavily dependent on the choosing of the initial non-target points, i.e., position, number N . If few points are chosen, the tracking stability will be lost. But, 2) if many points are chosen, from equation (2) we can see that the calculation cost is in proportion to N , which reduces the processing speed. Also 3) if many points are chosen, the tracking result will be easy to be influenced by the background, when the background has the similar color to the target object. For example, if each non-target point has the uniform risk r , then the whole risk is $N \times r$. 4) It scales poorly because the non-target points could only translate, and therefore it also could not deal with the rotation.

3. K-means tracking with a variable ellipse model

For solving those remained problems, in this paper, we propose a variable ellipse model. In the model, we use the center of the ellipse to represent the center of target, and use the ellipse circumference to represent centers of non-target. Still more, the ellipse parameters are adap-

tively controlled according to the target clustering result.

3.1 Improvement in the speed and stabilization by the ellipse model

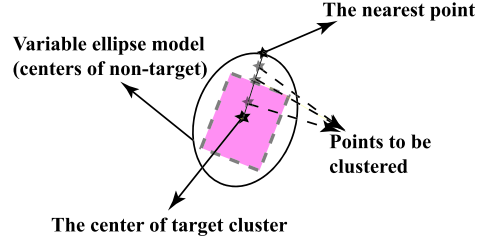


Fig 3 Using the ellipse model to cluster the unknown points

From equation (2), we can see one of the parameters that influence the speed and stability is the number N of the non-target points. So, the number N needs to be reduced as much as possible, and at the same time the stability should be maintained. As shown in the figure 3, we assume all the centers of non-target points that lie on the ellipse circumference are different from the center of target point in the color/position space. Therefore, it is not necessary to calculate minimum distance from all the centers of non-target points, only *one nearest non-target point* is necessary to be calculated. The nearest non-target point (hereafter *nearest point*) is a cross-point made by the ellipse circumference and a straight line connected by the target center and an unknown point that is within the ellipse. With this approximate assumption, the original K-means clustering is reduced to 2-means clustering, i.e., one target cluster and one non-target cluster. Then, we can convert equation (2) into

$$d_{NT}(t) = \|\mathbf{v}_{x,y}(t) - \mathbf{v}_{n1}(t)\|^2 \quad (5)$$

Here $\mathbf{v}_{n1}(t)$ is the feature vector of the nearest point at t time, and $\mathbf{v}_{x,y}(t)$ is the vector of a point needs to be clustered.

Just because of computing only one point of the centers of non-target automatic selected from the ellipse model, with this new model, we make the following improvements to the previous method: 1) The choosing of the initial non-target points becomes very easy. 2) The processing speed is N times quicker than before. 3) The failure risk is reduced to $1/N$ of the conventional method. So this new model will be seldom influenced by the background and be come more robust. Figure 4 shows an example of the new result.



图 4 The new result

3.2 Adaptation in the zoom and deformation by the variable ellipse model

To solve the problem of scaling and the deformation of the target object, we used the Mahalanobis Metric to control the parameters of ellipse model. That method needs us to compute the distribution of the clustered target points. In the following subsection, we describe the covariance matrix and Gaussian probability density of the clustered target points, which will be used in the Mahalanobis Metric. And then, we describe the controlling method of the ellipse parameters with the Mahalanobis Metric.

3.2.1 Computation of Gaussian probability density function

In this paper, we use the Gaussian probability density function (*pdf*) to compute the distribution of the clustered target points. The Gaussian *pdf* of a random n -dimensional vector $\mathbf{Z} = [z_1, z_2, \dots, z_n]^T \in \mathcal{R}^n$ is

$$f(\mathbf{Z}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\mathbf{Z} - m_{\mathbf{Z}})^T \Sigma^{-1} (\mathbf{Z} - m_{\mathbf{Z}})\right\}. \quad (6)$$

Where, $m_{\mathbf{Z}} = [E(\mathbf{Z}_1), E(\mathbf{Z}_2), \dots, E(\mathbf{Z}_n)]^T$ is the mean vector of the random vector \mathbf{Z} , $\Sigma = E[(\mathbf{Z} - m_{\mathbf{Z}})(\mathbf{Z} - m_{\mathbf{Z}})^T]$ is the covariance matrix, and $n = \dim \mathbf{Z}$ is the dimension of the random vector.

In this research, in order to adapt to zoom and deformation of the target object, after the clustering at t time, as shown in figure 5 we can get a set of clustered target points. We make this set $\mathbf{Z}_i = \{x_i, y_i\}^T; \{i = 1, \dots, m\}$, and we take $n = 2$, so the mean vector is

$$E[\mathbf{Z}] = E \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} m_X \\ m_Y \end{bmatrix}, \quad (7)$$

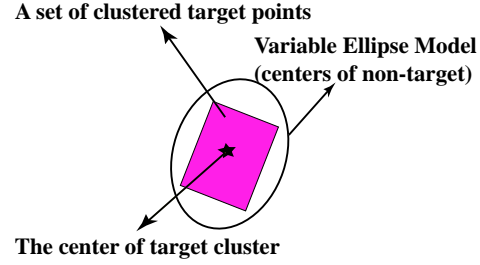


图 5 Using the clustered target points to control the parameters of the ellipse model

and the covariance matrix is

$$\Sigma = \begin{bmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{bmatrix}, \quad (8)$$

here, σ_X^2, σ_Y^2 are the variance of the random image coordinates x and y , and ρ is the correlation coefficient of the variables x and y .

So the 2-dimensional Gaussian *pdf* is

$$\begin{aligned} f(z) &= \frac{1}{2\pi\sqrt{\det\Sigma}} \exp\left\{-\frac{1}{2}[x - m_X \quad y - m_Y]\Sigma^{-1} \right. \\ &\quad \left. [x - m_X \quad y - m_Y]^T\right\} \\ &= \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}} \exp\left\{-\frac{1}{2(1-\rho^2)} \left(\frac{(x - m_X)^2}{\sigma_X^2} \right. \right. \\ &\quad \left. \left. - \frac{2\rho(x - m_X)(y - m_Y)}{\sigma_X\sigma_Y} + \frac{(y - m_Y)^2}{\sigma_Y^2} \right)\right\}, \quad (9) \end{aligned}$$

where the role played by the correlation coefficient ρ is evident. Figure 6 shows an example of Gaussian *pdf*.

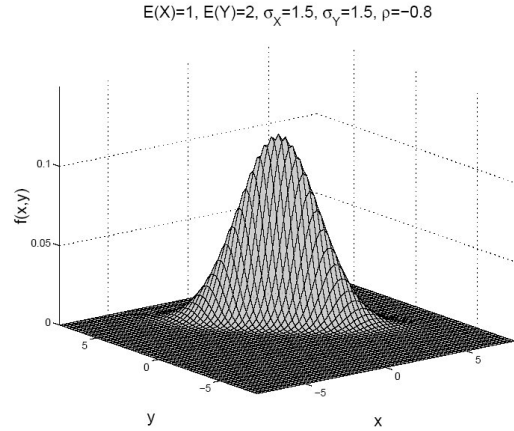


图 6 The Gaussian pdf with $m_X = 1, m_Y = 2, \sigma_X = 1.5, \sigma_Y = 1.5$

3.2.2 Refreshing the non-target model

For equation (6), we consider a constant $J_1 \in \mathcal{R}$. The locus of the *pdf* $f(\mathbf{Z})$ is greater or equal to a specified

constant J_1

$$\{\mathbf{Z} : \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} [\mathbf{Z} - m_{\mathbf{Z}}]^T \Sigma^{-1} [\mathbf{Z} - m_{\mathbf{Z}}] \geq J_1\right\} \quad (10)$$

Which is equivalent to

$$\{\mathbf{Z} : [\mathbf{Z} - m_{\mathbf{Z}}]^T \Sigma^{-1} [\mathbf{Z} - m_{\mathbf{Z}}] \leq J\} \quad (11)$$

Where $J = -2 \ln((2\pi)^{\frac{n}{2}} J_1 |\Sigma|^{\frac{1}{2}})$ is an n-dimensional ellipsoid centered at the mean $m_{\mathbf{Z}}$ and whose axis are only aligned with the Cartesian frame if the covariance matrix Σ is diagonal. The ellipsoid defined by equation (11) is the region of minimum volume that contains a given probability mass under Gaussian assumption.

When in equation (11) rather than having an inequality, there is equality as:

$$\{\mathbf{Z} : [\mathbf{Z} - m_{\mathbf{Z}}]^T \Sigma^{-1} [\mathbf{Z} - m_{\mathbf{Z}}] = J\} \quad (12)$$

The scalar quantity of equation (12) is known as the Mahalanobis distance of the vector \mathbf{Z} to the mean $m_{\mathbf{Z}}$.

The Mahalanobis distance, is a normalized distance where normalization is achieved through the covariance matrix. The surfaces on which J is constant are ellipsoids that are centered about the mean $m_{\mathbf{Z}}$, and whose semi-axis are \sqrt{J} times the eigenvalues of Σ , as seen before. In the special case where the random variables that the components of \mathbf{Z} are uncorrelated and the same variance, i.e., the covariance matrix Σ is a diagonal matrix with all its diagonal elements equal, these surfaces are spheres, and the Mahalanobis distance becomes equivalent to the Euclidean distance.

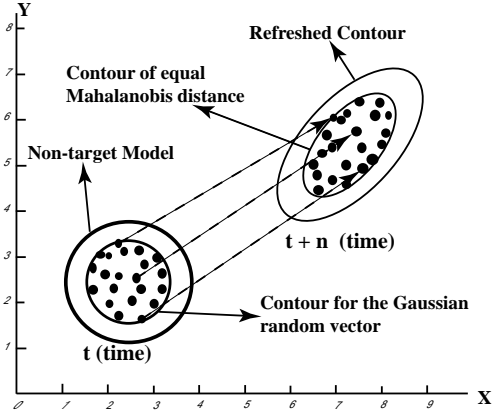


Figure 7 The contours of the equal Mahalanobis distance around means for a second order Gaussian random vector

Figure 7 represents the contour of the equal Maha-

lanobis distance. In another words, any point $p(x, y)$ in the ellipse is at the same Mahalanobis distance to the center of the ellipse. This plot enhances the face that the Mahalanobis distance is weighted by covariance matrix.

As to $n = 2$, we can write equation (12) in the following format

$$\begin{aligned} g(\mathbf{Z}) &= \{[\mathbf{Z} - m_{\mathbf{Z}}]^T \Sigma^{-1} [\mathbf{Z} - m_{\mathbf{Z}}]\}^{\frac{1}{2}} \\ &= c_{22}(x - m_X)^2 - 2c_{12}(x - m_X)(y - m_Y) \\ &\quad + c_{11}(y - m_Y)^2 \\ &= c_{22}x^2 - 2c_{12}xy + c_{11}y^2 + 2x(c_{12}m_Y - c_{22}m_X) \\ &\quad + 2y(c_{12}m_X - c_{11}m_Y) + c_{22}m_X^2 + c_{11}m_Y^2 \\ &\quad - 2c_{12}m_Xm_Y \end{aligned} \quad (13)$$

Where

$$\Sigma = \begin{bmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \quad (14)$$

here $c_{12} = c_{21}$.

The ellipse can be written in the following equation:

$$g(z) = Ax^2 + 2Bxy + Cy^2 + 2Dx + 2Ey + F \quad (15)$$

So compared with equation (13), we can get the following parameters as the ellipse:

$$\begin{aligned} A &= c_{22} & B &= -c_{12} & C &= c_{11} \\ D &= c_{12}m_Y - c_{22}m_X & E &= c_{12}m_X - c_{11}m_Y \\ F &= c_{22}m_X^2 + c_{11}m_Y^2 - 2c_{12}m_Xm_Y \end{aligned}$$

With these parameters, we could control the ellipse model to be kept up with the main distribution of the clustered target points at t time (see figure 5). Therefore, the variable ellipse model are adaptively controlled according to the target's zoom and deformation from t time to $t + n$ time as shown figure 7. Some experiment results are shown in figure 8.

With equation (9) we can get the probability of all the clustered target points in the input image, and when $f(\mathbf{Z}) < Thr$ we will eliminate these points \mathbf{Z} because these points may contain more noise according to the probability distribution. Here, Thr is a confidence level. With this processing, our new system can be adapted to the target appearance change and size change, and more robust against the noise due to background involution or the illumination changes.

4. Comparison experiment

To evaluate the performance of our new K-means tracking method, we express the comparison experiment results against other methods.

Method [1]: Sum of Absolute Difference (SAD),



(a) Tracking a CD-ROM having a hole and high-light patterns in translation, rotation in depth and zoom



(b) Tracking a pen in translation, rotation in plane and zoom

图 8 Some results of new method.

which is a variable template matching method by using the gap criterion.

Method [2]: Kernel-based object tracking.

Method [3]: Previous K-means tracking (with N non-target points).

Method [4]: Presented K-means tracking (with one non-target point and variable ellipse model)

Here, the samples of tracked objects are an incense stick and a comb. Since these objects have holes, which make the distribution of data become non-linear and with the changes of the background it will be more difficult to track these things by the presented methods until now.

To catch up with the changes of the color, template matching method (Method [1]) will refresh the template. But, while refreshing, the error will be accumulated and the accumulated error will finally lead to the lost of the object. Method [2] evaluate the likelihood of the clustered points in the color histogram, and the when the illumination changes or the object rotates the color of the object changes also, therefore, it may lose the object. Compared with Method [4], Method [3] can not deal with rotation and is slower.

We took the experiments on a desktop, whose CPU is Intel 1.0Ghz and the memory is 512 MB. Our new method 's processing speed is $0.07 \sim 0.08$ sec/frame when the image size is 640×480 pixels.

The results of comparison experiments are shown in figures 9 and 10, these results have shown the efficiency

of the proposed method.

5. Conclusion

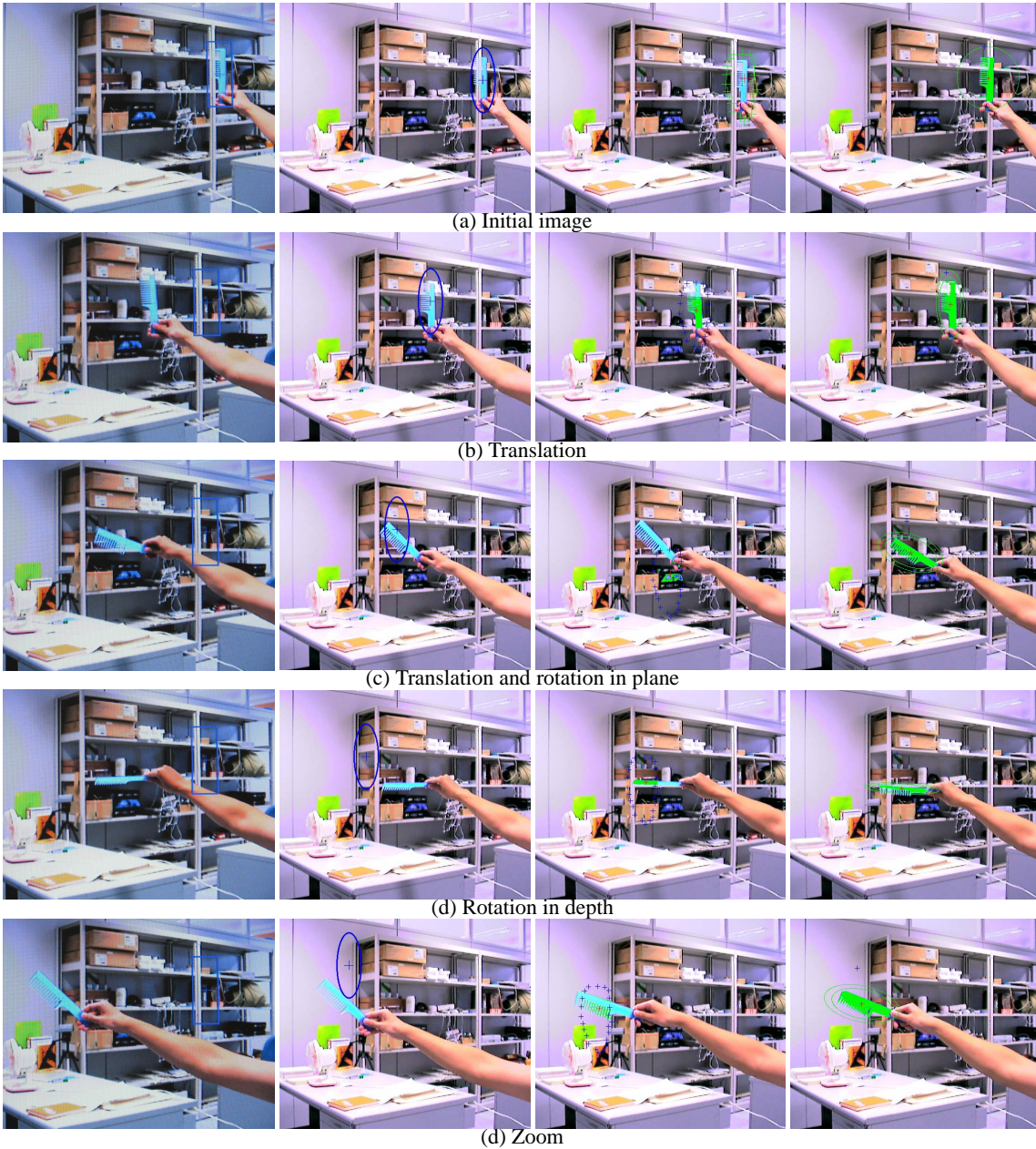
In this paper, we present an improved K-means tracking method in color image sequence. Inheriting the advantages of our previous system, with a single target point and a variable ellipse model which is adaptively controlled by the target detection result, we improve the speed, robustness, size and shape change of our new system. Various experiments show that real-time K-means tracking becomes possible.

6. Acknowledge

This research is partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Grantin-Aid for Scientific Research (A)(2), 16200014 and (C) 16500112.

参 考 文 献

- 1) Christoph Grüβl, Timo Zimβer, and Heinrich Niemann *Illumination Insensitive Template Matching with Hyperplane*, DAGM (German), 2003.
- 2) Alexander C. Gerg, Jitendra Malik *Geometric Blur for Template Matching*, CVPR, 2001.
- 3) Kang, D., Kweon, I. *Fast and Stable Snake Algorithm for Medical Images*, Pattern Recognition Letters 20 (1999) 507-512.
- 4) Bernd Heisele *Motion-based Object Detection and Tracking in Color Image Sequence*, ACCV, 2000.
- 5) B.Heisele, U.Kreβel, W.Ritter *Tracking Non-Rigid, Moving Objects Based on Color Cluster Flow*, CVPR, 1997.
- 6) Toshikazu Wada, Toshiaki Hamatuka, Takekazu Kato *K-means Tracking: A Robust Target Tracking against Background Involution*, MIRU, 2004.
- 7) M. Isard, A. Blake *CONDENSATION—conditional density propagation for visual tracking*, Int.J. Computer Vision, Vol. 29, No.1 pp. 5-28, 1998.
- 8) M. Isard, A. Blake *Contour tracking by stochastic propagation of conditional density*, Proc. European Conf. on Computer Vision, vol. 1, pp.343-356, 1996.
- 9) D. Comaniciu, V. Ramesh, P.Meer *Kernel-Based Object Tracking*, IEEE Trans. PAMI, Vol.25, No.5, 2003.
- 10) Zia Khan, Tucker Balch, Frank Dellaert *An MCMC-Based Particle Filter for Tracking Multiple Interacting Targets*, ECCV, 2004.
- 11) Tao Zhao, Ram Nevatia *Tracking Multiple Humans in Crowed Environment*, CVPR, 2004.
- 12) Ankur Agarwal, Bill Triggs *Tracking Articulated Motion using a Mixture of Autoregressive Models*,



(1) Template matching (2) Kernel-based tracking (3) previous method (4) proposed method

☒ 9 Some results of comparison experiments with a comb under complex background and illumination variance

ECCV, 2004.

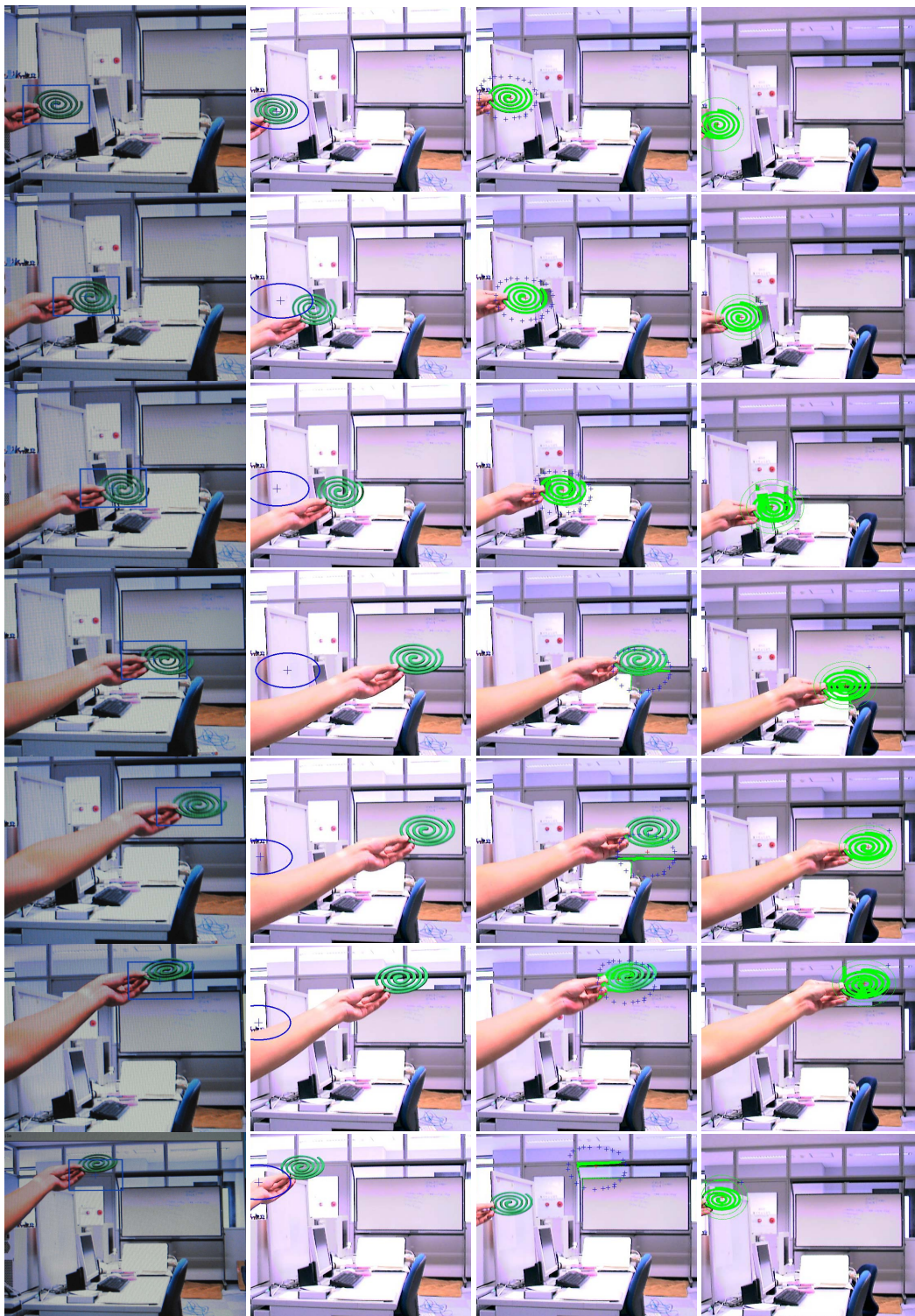
13) M. Isard, J. MacCormick, BraMBle *A Bayesian Multiple-Blod Tracker*, Proc ICCV, 2001.

14) H. Tao, H.S. Sawhney, R. Kumar *A sampling algorithm for tracking multiple objects*, Proc. Workshop Vision Algorithms with ICCV, 1999.

15) J.Hartigan, John Wiley, Sons *Clustering Algorithms*, New York, 1975.

16) J. Hartigan, M. Wong *Algorithm ASI36: A k-means*

clustering algorithm, Applied Statistics, Vol.28, pp. 100-108, 1979.



(1) Template matching (2) Kernel-based tracking (3) previous method (4) proposed method

⊗ 10 Some results of comparison experiments with an incense stick under complex background and illumination variance