The 12th Korea-Japan Joint Workshop on Frontiers of Computer Vision, 2–3, Feb.,2006, Tokushima, Japan

Tracking-Oriented Feature Extraction based on Texture Richness Analysis

Yukiyasu DOMAE, Henrik STIER, Shun'ichi KANEKO and Takayuki TANAKA

Graduate School of Information Science and Technology, Hokkaido University
Kita 14, Nishi 9, Kita-ku, Sapporo, 060-0814, JAPAN

Abstract

Feature extraction and tracking are widely applied in the industrial world today. It is still an important topic in Machine Vision. We present a feature extraction and tracking method which is based on pure rich textures in images and has robustness under irregular conditions, such as illumination change, deformation of objects and so on. The method is composed mainly of two algorithms: Entropy filter and Orientation Code Matching (OCM). The entropy filter points up areas of images being messy distribution of orientation codes. The orientation code is determined by detecting the orientation of maximum intensity change around neighboring eight pixels. It is defined as simply integral values. The areas being extracted by the entropy filter doesn't depend on edges and gradations of images and has robustness against illumination change and motion of objects. Therefore, we can say the areas have "pure rich texture" and are suitable for tracking based on visual information. And then, the OCM, a template matching method using the orientation code, is applied to track templates being centered on the features and updates the templates each frame. Because of their large robustness against illumination change and deformation of templates, we can track the templates robustly.

1 Introduction

Machine vision systems, which are based on image processing technologies effective in real world environments, are widely used in the field of inspection, monitoring, measurement and so on. Robust feature extraction and matching, in those systems, has been one of the basic technologies as pre-processing for tracking, continuation of consistent identification of the extracted features over temporal direction. The purpose of this research is subjective analysis of robust tracking and developing a robust feature extraction algorithm which performs stably in real environments.

The first step in the algorithm is acquiring or discovering visual cues or features from the scence. The problem has been addressed in computer vision for a long time. Edge detection is one of the most commonly used solutions, therefore, many edge detectors [7, 8] have been proposed. In many cases, however, edges are affected by artificial parameters in any implementations and also by change in factors such as postures. Therefore, in many researches, non-edge features have been designed for practical applications. Moravec operator [1] extracts features by evaluating the variance of intensity in neighboring pixels. Harris operator [2] is based on the variation of local auto-correlation over different orientations. KLT algorithm [3] which is based on a model of affine image change is well-suited to extract features from video sequences and tracks them. They are well-known non-edge feature detectors, but they have some weakness in robustness against local illumination change such as shading and highlighting, which may cause troubles in identifying temporal consistencies of spatially distributed individual features in the frames.

A richness (RC) is a feature detector being robust against these conditions [4]. It points up areas being *messy* distribution of the orientation codes which is designed as a robust spacial features based on local brightness distribution. Orientation code (OC) [5] is determined by detecting the orientation of maximum intensity change around neighboring eight pixels, which is applied to search tags, potentially possible landmarks, for understanding an unknown environment. We apply the method to extract features from video-sequences. And then these features are tracked by applying a kind of robust template matching method: orientation code matching (OCM) [6]. As well as the richness, OCM has robustness against the local illumination change.

Combining the feature extraction and the template matching algorithm, a tracking algorithm is developed in this paper.

The rest of the paper is organized by eight sections. OC and RC are described in Section 2 and 3. Then, we explain about our distribution density variable feature extraction method using OC and RC in Section 4. Section 5 describes the comparison of our feature extraction method with KLT. Our tracking algorithm and applying it to a planar landmark and human tracking is described in Section 6 and 7. Finally, conclusions and future works are mentioned.

2 Orientation code

OC is defined as the quantized orientation of maximum intensity change around neighboring eight pixels [5]. Let $\Delta I_x = \frac{\partial I}{\partial x}$ and $\Delta I_y = \frac{\partial I}{\partial y}$ represent the horizontal and vertical gradient. Sobel operator [7] is used to compute the gradients. Quantizing the orientation of maximum intensity change by using quantize width Δ_θ, OC c_{xy} is defined as below.

$$c_{xy} = \begin{cases} \left[\frac{\tan^{-1}\left(\frac{\Delta I_y}{\Delta I_x}\right)}{\Delta_\theta}\right] & \text{if } |\Delta I_x| + |\Delta I_y| > \Gamma \\ N = \frac{2\pi}{\Delta_\theta} & \text{otherwise} \end{cases} \quad (1)$$

If $|\Delta I_x|+|\Delta I_y|$ is smaller than the threshold value Γ, Γ excludes the pixels and define the OC of the pixels as expeptional value N. In doing so, stable extraction of OC becomes possible. In reverse case, OC is defined as the integer 0∼16 ($\Delta_\theta = \frac{\pi}{8}$).

3 Richness

In order to continue stable tracking of real objects in the scene with complicated background, stable and prominent features have to be extracted and checked their consistency by canceling deceiving features in their neighbours. If the objects move in the scene, according to change in the postures, emerging and occluding feature points should be detected as the objects change their appearances. Furthermore, it is preferable for the features to be insensitive according to viewing directions from the camera. We utilize richness [4] as an evaluator for extraction effective regions that can get preferment in comparison with their neighboring ones.

Because the method points up areas being messy distribution of orientation codes, it doesn't depend on general illumination change and has robustness against local illumination change such as highlightning or shading. Therefore, the areas which have pure rich texture are easily extracted as features and makes stable tracking possible.

Let $h_{xy}(i)$ $(i = 0, 1, \cdots, N)$ represent the frequency of appearance of orientation codes which includes in a M-by-M local area I_{xy} centering on (u, v). Then, their relative frequency is computed as follows:

$$P_{xy}(i) = \frac{h_{xy}(i)}{M^2 - h_{xy}(N)} \qquad (i = 0, 1, \cdots, N-1) \quad (2)$$

where $h_{xy}(N)$ means the exceptional values and is excluded from the relative frequency. Next, an entropy E_{xy} is defined as below.

$$E_{xy} = \sum_{i=0}^{N-1} P_{xy}(i) \log_2 P_{xy}(i) \quad (3)$$

If the distribution of OC in the I_{xy} is uniform, i.e. $P_{xy}(i) = \frac{1}{N}$, the entropy becomes maximum: $E_{max} = \log_2 N$. Then, RC is defined as

$$R_{xy} = \begin{cases} \frac{E_{xy} - \alpha_e E_{max}}{E_{max} - \alpha_e E_{max}} & \text{if } E_{xy} \geq \alpha_e E_{max} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where the threshold value α_e $(0 < \alpha_e < 1)$ is defined to remove the low entropy areas. We give the process computing RC the name of the entropy filtering.

4 Distribution density-variable feature extraxtion

Using the two alogorithms above-mentioned, a feature extraction program is implemented. Fig. 1 shows the flow of the proposed program. A raw image is first transfered to the OC image, from which the RC image or the two dimensional RC distribution is made for entropy filtering or feature selection in the final step. For smoothing the RC image, the box filter is utilized in the third step. Next, a most highlighted point I_p is searced from the whole box-filtered RC image. Then, an area being centered on I_p is weighted by an inverse gaussian window function shown in Fig 2 (a). After repeating N times, N features are extracted from the raw image. The feature distribution density can be manipulated by changing the size of the weight function $w(x)$. Fig. 2 shows the effect of size variation of $w(x)$. If the size becomes smaller, the features are localized. The size S should be determined by considering the number of features and high RC area.

We should note how the effect of the parameter change on the feature extraction is. N, the quantizing number of OC, and α_e, the threshold in entropy definition, affects the RC image. But they do

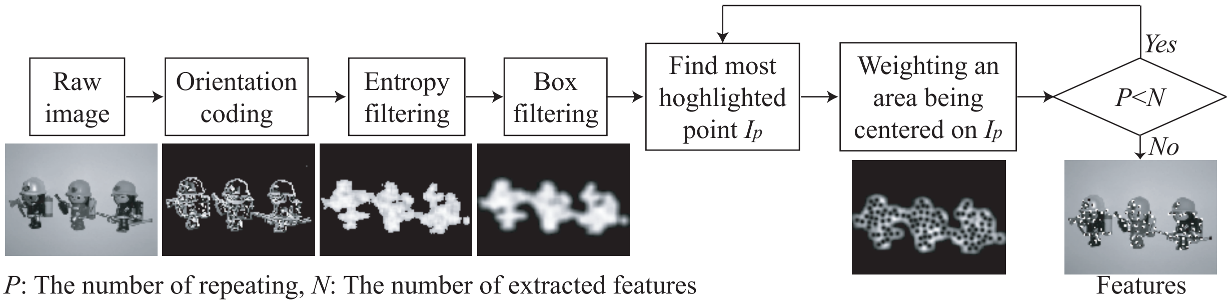

Figure 1: Procedure of feature extraction.

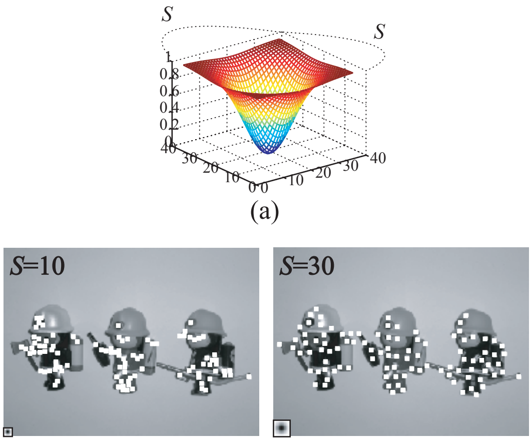

Figure 2: Distribution density-variable feature extraction, (a) An inverse gaussian window function $w(x, y)$, (b) 100 features extraction result ($S = 10$), (c) extraction result ($S = 30$).

not heavily affect the feature extraction. M, the side length of the domain for entropy definition, affects RC too. It decides whether or not the RC image becomes blurry. Γ, the threshold in the definition of OC, may be the most important parameter for the feature extraction. If it becomes bigger, many features are extracted, for example, from corners, while if it becomes smaller the features are extracted more sensitively. In fact, it means the extraction of areas which has fine texture.

5 Comparison experiment

In this section, the perfomance of the feature extraction algorithm is evaluated in comparison with a well-known extraction algorithm. To evaluate perfomance of the feature extraction method, the comparison between our method and KLT (Kanede-Lucas-Tomashi) [3] method have been tried. KLT is a well-known feature extraction and tracking method and is based on a model of affine image change. It is optimized for feature extraction from video-sequences. The implementation details based on a free program in C++ code being made public on the Internet [1]. Parameters of our method are set as $N = 17$, $\alpha_e = 0.995$, $M = 10$, $\Gamma = 17$, $S = 23$.

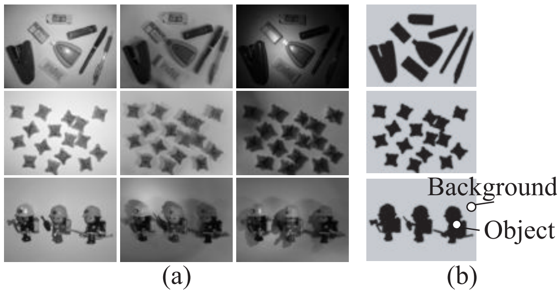

Figure 3: (a) Nine images for experiment, three scenes are taken under three different illumination conditions. (b) Pre-ordained object and background areas of the three scenes.

Fig. 3 shows the scenes for the experiments, which include different kinds of objects, stationeries, cookies and toys. They are taken under three different illumination conditions. The object area (black) and the background area, a white flat plane (gray), are defined by Fig. 3 (b). One hundred features have been extracted from each image by our method and KLT. Examples of extraction results are shown in Fig. 4. For example, KLT extracts many features from the shadow as illustrated in Fig. 4, where false features are mainly from edges and corners. We have checked a feature detection rate from the object areas. In these experiments, we could obtain by our method the detection rate 95.8%, while by KLT the ratio could be 86.5%. This result can be explained by the fact that RC is robust for brightness change caused by illumination fluctuation but KLT is not so.

[1] http://robotics.stanford.edu/~birch/klt/

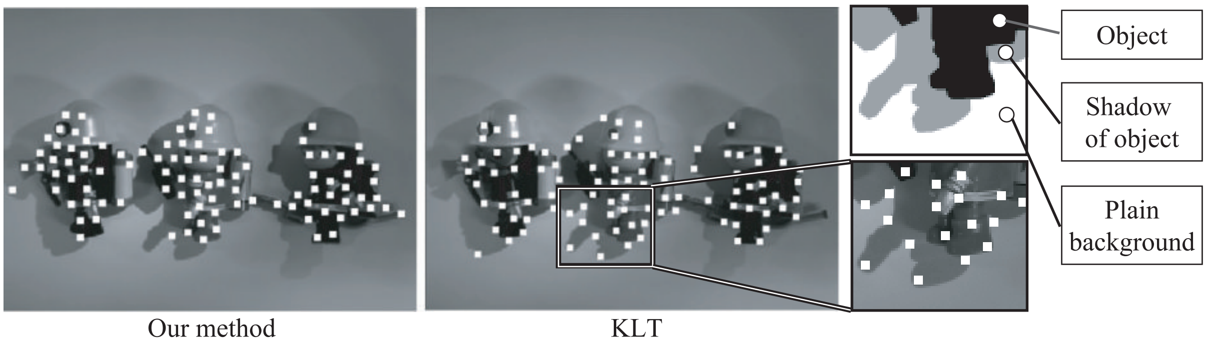

Figure 4: Examples of extraction results, KLT extracts many features from shadows of objects.

6 Application to landmarks tracking

For tracking features, we use OCM, a template matching algorithm based on OC. It has large robustness against change in illumination in nature and rather experimentally effective for matching deformed objects. For details, please refer to [6]. In order to do performance servey and evaluation of the OCM tracking argorithm, we have tried tracking planer landmarks. Five pieces of landmark of star-and-circle shape are depicted on a plate as shown in Fig. 5 (a), and they are subject to be tracked in the scene with change in posture, scaling and illumination. The landmarks have high richness, therefore they are extracted easily by using texture richness analysis. The plate is moved manually by an operators hand. Fig. 5 (b) shows the result of the plain landmark tracking. The areas which are enclosed by squares are tracked target areas. Insted of the deformation of the target areas, rotation, scaling and illumination change in the motion of the plain object, the method can keep tracking well. The parameters are set as below. $N = 16$, $M = 20$ (Face tracking: $M = 10$), $\Gamma = 23$, $\alpha_e = 0.88$ (Face tracking: $\alpha_e = 0.7$). The image size is 320×240. The frame rate of the algorithms is about 7.8fps.

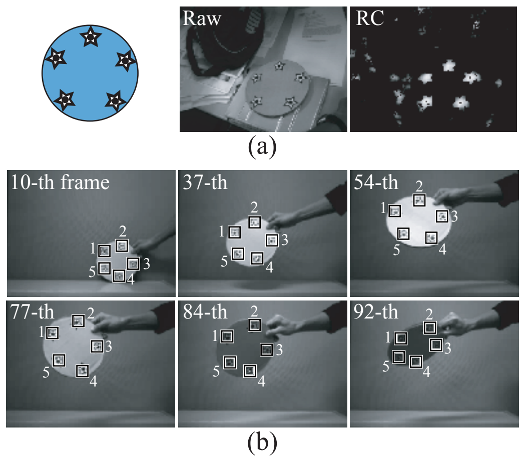

Figure 5: Planar landmark tracking using OCM, (a) An object having six high richness star-shape landmarks, (b) a tracking result.

7 Automatic template definition and update

If objects which should be tracked have no landmarks, we have to define templates (or RoI) for tracking. The features being extracted by our method are well-suited to tracking, but too much to track. Therefore, we need to generate few templates from this feature distribution. K-means [9], a well known and simple clustering algorithm to group features and calculate their centroids, is used for this purpose.

The features have been classified into a specified number of clusters, templates can be adopt to the cluster centroids. Grouping extracted features, their position information is used. Another adequate data therefore could be the richness values. The first step of the k-means algorithm is to define k initial centroids, this number is fixed and depends on the quantity of features. We set these uniformly distributed in the first frame. The centroids could be specified also by coincidence within a segmented area. Then, for each feature, the distance to these centroids is computed and the feature is assigned to the closest one. When all features have been assigned, the centers of the k clusters have to again be determined. The last two steps must be repeated until the centers do not move any longer.

To measure the distance between a feature $f_i^{(j)}$ and a centroid c_j, we uses the Pythagorean theorem. K-means algorithm aims at minimizing a function, that is defined as

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \sqrt{\left(f_{i,x}^{(j)} - c_{j,x}\right)^2 + \left(f_{i,y}^{(j)} - c_{j,y}\right)^2}.$$

This function is an indicator of the distance of the n features to their associated cluster centers. It is proved [10] that k-means algorithm will always terminate, but it does not necessarily find the most optimal result, analogous to the global function minimum. But we obtained thereby sufficiently good results and it is not necessary to run the clustering procedure several times. But it is possible, that we create cluster without or only one feature, then we do not use them for further activities.
The complicacy is how to choose k. In our experiments, this parameter is set manually, for an automatic localization background knowledge is essential. Additive information like the number of persons in a scene or even the ken of segmented areas could be very helpful. An advancement of performance with k-means using background knowledge is described in [11]. An optimal value of k is also dependent on n.

A disadvantage of the presented method is the adverse effect by features they are not related to the interesting object. Detected features in the background causes a movement of cluster centroids to an area in the video sequence that not contains interesting objects to track. To prevent this, a real-time background subtraction, e.g. background segmentation based on the Gaussian Mixture Model [13], is needed. But the way to distinguish between objects and background depends on applications. Therefore, we don't make further reference. To give an example, moving man's face tracking under complicated background have been tried in [12] and forward subtraction is used in that algorithm to select features on the moving face. Above-mentioned we described a method to initialize templates by finding well suited positions with clustering by k-means. The templates are captured at the cluster centroids in the initial frame and tracked in the following by OCM.

Feature extraction and clustering is not necessary for the tracking run, except some special cases. Such cases can be movements of objects (e.g. persons) out of the regarded scene or head turn of a human. Under these circumstances occurs an error because the template can not be tracked any longer. But it is not possible to identify such failures by a threshold or something else relative to the OCM values, because these values are always unsteady and nearby. So we decided to use the displacement of the central point for each template. If

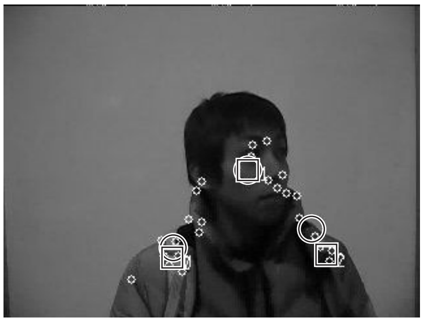

Figure 6: Extracted features and cluster centroides (bigger circles), while updating templates (squares).

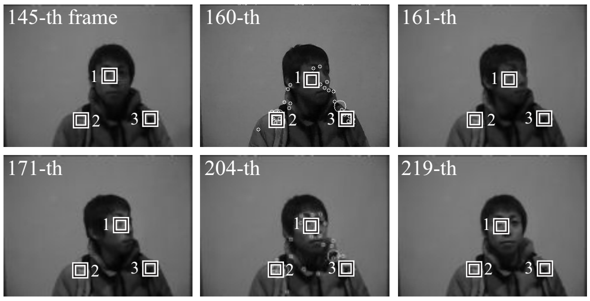

Figure 7: The result of head turn tracking using automatic template generation and update.

the distance of centers in consecutive frames exceed a defined threshold Γ we realize a misplaced template and know that we have to update this one. A new feature extraction and cluster recalculation is necessary (Fig.6). It is conceivably to use the last known template positions for cluster initialization, but there is no strong discrepancy ascertainable. The updated template is extracted at the appropriate cluster centroid, for remaining templates nothing changes.

The result of the presented method at the example of a head turn is shown in Fig.7. There are three well suited templates in the 145-th frame. Because of head turning template one can not be tracked any longer in the 160-th frame, an update is necessary. The further frames (from 204-th to 205-th) shown a similar updating process concerned the template one while back turning, there are no redefinitions of the other templates in the whole sequence. The illumination changes in the scene, brightness is slowly manipulated by covering a light source. The outcome is very good, the positions of the considered template in the first and the last appeared frame are even the same. Among these updates an adequate position for the template is found and tracked by OCM.

8 Conclusion

We have described a novel algorithm for feature extraction and tracking, which consist of OC, RC evaluation, feature distribution density modification, OCM and automatic template definition using k-means. And then, the robustness of the algorithm against change in illumination was shown formally and experimentally. The effectiveness of the algorithm is tested and recognized through comparison the algorithm to KLT. The landmark and human tracking have been performed for evaluation of the proposed algorithm.

As future works, we are going to improve the algorithm in computation cost and robustness, and as an important research topic some formalization of tracking and template modification will have to be considered with experiments using real world data.

References

[1] H. Moravec: Towards Automatic Visual Obstacle Avoidance, Proc. 5th Int. Joint Conf. Art. Intell, pp.584, 1977.

[2] C.Harris and M.Stephens: A combined corner and edge detector, Proc. 4th Alvey Vision Conf.., pp.147-151, 1988.

[3] J. Shi and C. Tomasi: Good features to track, Proc. IEEE Conf. Comput. Vision Patt. Recogn, pp.593-600, 1994.

[4] H. Takauji, S. Kaneko and T. Tanaka: Robust tagging in strange circumstance, The Transactions of the Institute of Electrical Engineers of Japan (C), vol.125, pp.926, 2005 (in japanese)

[5] F. Ullah, S. Kaneko and S. Igarashi: Orientation code matching for robust object search, IEICE Trans. of Inf. & Sys, E84-D(8), pp.999-1006, 2001.

[6] F. Ullah, S. Kaneko and S. Igarashi: Tracking looming and receding objects by orientation code matching, The Journal of the IIEEJ, vol.31. No. 1, pp.94-102, 2002.

[7] I. Sobel: Neighbourhood coding of binary images for fast contour following and general array binary processing, Computer Graphics and Image Processing, vol.8, pp.127-135, 1978.

[8] J. Canny: A Computational Approach to Edge Detection, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.8. No. 6, 1986.

[9] J. B. MacQueen: Some Methods for classification and Analysis of Multivariate Observations, Proc. Fifth Berkeley Sympos. Math. Statist. and Probability, vol.1, pp.281-297, 1967.

[10] T. Kanungo, D. Mount, N. S. Netanyahu, C. Piatko, R. Silverman, A. Y. Wu: The Analysis of a Simple k-Means Clustering Algorithm, Symposium on Computational Geometry 2000, pp.100-109, 2000.

[11] K. Wagstaff, C. Cardy, S. Rogers, S. Schroedl: Constrained K-means Clustering with Background Knowledge, Proc. 18th International Conference on Machine Learning, pp.577-584, 2001.

[12] Y. Domae, S. Kaneko, T. Tanaka: Robust Tracking based on Orientation Code Matching under Irregular Conditions, Proc. of SPIE, Vol.6051, 0S, 2005.

[13] Y. Dedeoglu: Human Action Recognition Using Gaussian Mixture Model based Background Segmentation, http://www.cs.bilkent.edu.tr/ ˜guvenir/courses/cs550/workshop/, 2005.

Domae, Yukiyusu: received the B.S. degree in systems engineering from Hokkaido University, Japan, in 2004. He pursue currently a postgraduate degree at the Graduate School of Information Science and Technology, Hokkaido University. His research interest includes machine vision and robust image recognition.

Stier, Henrik: received the certificate of apprenticeship as a technical assistant in computer science in 2000, the intermediate diploma in information technology in mechanical engineering from Technical University Berlin, Germany, in 2004. He is a participant of the DeMaMech exchange program at the Graduate School of Information Science and Technology, Hokkaido University, from 2005 to 2006.

Kaneko, Shun'ichi: He was a Research Assistant in the Department of Computer Science from 1980 to 1991, an Associate Professor in the Department of Electronic Engineering from 1991 to 1995, and an Associate Professor in the Department of Bioapplication and Systems Engineering in 1996, at the TUAT, Tokyo, Japan. He is currently a Professor at the Graduate School of Information Science and Technology, Hokkaido University. His research interests include machine and robot vision, image sensing and understanding, and robust image registration.

Tanaka, Takayuki: received the B.S., M.S. and Ph.D. degree in Mechanical Engineering and Intelligent Systems from The University of Electro-Communications, Japan, in 1991 , 1993 and 1996. He was a Research Associate in the University of Electro- Communications from 1996 to 2001, a Visiting Scholar in University of California, Irvine from 2001 to 2003, an Associate Professor in The University of Electro-Communications from 2003 to 2004, and currently an Associate Professor in Hokkaido University. His research interests include intelligent mobile robot, wearable human assisting robot and biological real-time pattern extraction.