ROBUST VISUAL TRACKING VIA A COMPACT ASSOCIATION OF PRINCIPAL COMPONENT ANALYSIS AND CANONICAL CORRELATION ANALYSIS

Yuxia Wang, Qingjie Zhao

Beijing Lab of Intelligent Information Technology, School of Computer Science,
Beijing Institute of Technology, Beijing 100081, P.R. China
yuxiawang2006@163.com

ABSTRACT

We propose a novel correlation-based incremental tracking algorithm based on the combination of principal component analysis (PCA) and canonical correlation analysis (CCA), which called Principal Component-Canonical Correlation Analysis (P3CA) tracker. We utilize CCA to evaluate the target goodness, resulting in more robust tracking than using holistic information, especially in handling occlusion. PCA is adopted to solve the Small Sample Size (3S) problem and reduce the computation cost in the generation of CCA subspace. To account for appearance variations, we propose an online updating algorithm for P3CA tracker, which updates the PCA and CCA cooperatively and synchronously. Comparative results on several challenging sequences demonstrate that our tracker performs better than a number of state-of-the-art methods in handling partial occlusion and various appearance variations.

Index Terms— Principal Component-Canonical Correlation Analysis, Small Sample Size problem, Adaptive appearance model, Visual tracking

1. INTRODUCTION

While numerous tracking methods have been proposed with demonstrated success in recent years, designing a robust tracking method is still an open problem, especially considering various complicated variations that may occur in real world condition, e.g., scale and pose change, illumination variation, occlusion, cluttered scenes, etc. One of the main reasons is the lack of the effective object appearance models, which play a significant role in visual tracking, to account for the appearance variation.

In existing tracking methods, most algorithms model the object appearance either by global descriptors or local descriptors. Color histogram [1, 2] is one of the most widely used global descriptor for its effectiveness and efficiency. Other global appearance models, based on raw pixel values, are widely adopted in [3, 4, 5, 6]. Although these methods mentioned above are proved to be effective when dealing with the illumination change and pose variation, they are less effective in handling partial occlusions as a result of the adopted holistic appearance model. In order to overcome the demerit mentioned in global-descriptor based appearance models, many trackers [7, 8] adopt local descriptor-based appearance models due to their incorporation of the spatial information with the appearance models. However, most of them do not make use of the correlation between local parts, which is a very important statistics information of the object in tracking. Recently, [9] uses a correlation-based observation model with canonical correlation analysis (CCA) as the appearance model, which is robust in handling occlusion and illumination variation. Inspired by [9], our P3CA tracker also utilizes the correlation-based appearance model, but in contrast to [9], we incorporate principal component analysis (PCA) into the calculation of the CCA subspace in order to solve the Small Sample Size (3S) problem and reduce the computation cost. Moreover, considering the disadvantage of static appearance models used in many previous trackers, we propose a novel incremental updating method for our P3CA tracker to handle the appearance variations.

The reminder of the paper is organized as follows. Section 2 shortly introduces the whole tracking system. In Section 3, we propose our P3CA with the basic theory of CCA and PCA. P3CA-based appearance model is introduced in Section 4. In Section 5, we introduce our incrementally updating method for P3CA. We present our tracking results with a detailed analysis in Section 6. Finally, our work concludes with Section 7.

2. SYSTEM OVERVIEW

In our tracker, we use a rectangle to represent the target and define the object state at time t as $\mathbf{S}_t = (dx_t, dy_t, sc_t, \theta_t, sr_t, \delta_t)$, where dx_t, dy_t, sc_t, θ_t, sr_t and δ_t represent the translation in horizontal and vertical axis, scale, rotation angle, aspect ratio, and skew direction respectively. We construct our tracker in the framework of Particle Filter (PF), which is commonly

This work was supported by the National Natural Science Foundation of China (61175096).

used in many trackers[10, 5, 6] due to its excellent characteristics.

For the dynamic model in PF, we assume a commonly used Gaussian random walk model:

$$p(\mathbf{S}_t|\mathbf{S}_{t-1}) = \mathcal{N}(\mathbf{S}_t; \mathbf{S}_{t-1}, \mathbf{\Psi}) \tag{1}$$

where $\mathbf{\Psi}$ is a diagonal covariance matrix which is composed of the standard deviations of each variable in the state vector $\mathbf{S}_t$.

For the appearance model in PF, we use our proposed P3CA appearance model, which is a combination of PCA and CCA. CCA is an effective method for analyzing the statistical correlation between two variables. Intuitively, using the statistical correlation relationship between the pairs of sub-patches for evaluation of the candidate objects tends to be more robust than using the holistic information, because with real environment changes (e.g. illumination variations or occlusions) the sub-patches will only be influenced by the identical and random environment noises. Possible split of a given image patch is shown in Fig. 1(a). In this paper, we adopt the vertical split. However, the calculation of the CCA subspaces in the area of pattern recognition always suffers from the 3S problem [11]. Thus, PCA is incorporated in our P3CA appearance model, which successfully solves the 3S problem and dramatically reduces the computation cost in the generation of the CCA subspace. Moreover, to account for the appearance variations, we propose an online updating algorithm for P3CA. The whole tracking system is shown in Fig. 1(b).

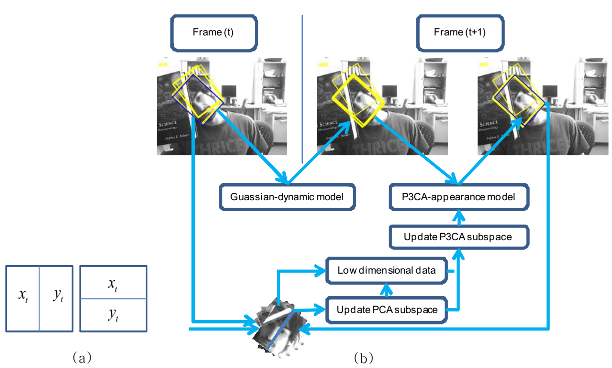

Fig. 1: (a)Possible split of a given image patch (b)System overview

3. P3CA: A COMPACT ASSOCIATION OF PCA AND CCA

The canonical correlation analysis (CCA) is an effective statistical method, which can convert the correlation between the two sets of random variables to the correlation between a few pairs of independent variables. Considering two random vectors with zero mean $\mathbf{x} \in \mathbb{R}^{d_x}$ and $\mathbf{y} \in \mathbb{R}^{d_y}$, CCA aims in finding a pair of project-vectors $\mathbf{u_x} \in \mathbb{R}^{d_x}$ and $\mathbf{u_y} \in \mathbb{R}^{d_y}$ such that $Corr(\mathbf{x}_1^*, \mathbf{y}_1^*)$ is maximized, where $\mathbf{x}_1^* = \mathbf{u}_x^T\mathbf{x}$ and $\mathbf{y}_1^* = \mathbf{u}_y^T\mathbf{y}$. Generally, the project-vectors $\mathbf{u}_x$ and $\mathbf{u}_y$ in CCA subspace can be obtained by (2):

$$max\ \mathbf{u}_x^T\mathbf{\Sigma}_{xy}\mathbf{u}_y \quad s.t.\ \mathbf{u}_x^T\mathbf{\Sigma}_{xx}\mathbf{u}_x = \mathbf{u}_y^T\mathbf{\Sigma}_{yy}\mathbf{u}_y = 1 \tag{2}$$

where $\mathbf{\Sigma}_{xy}$ represents the inter-class covariance matrix and $\mathbf{\Sigma}_{xx}$, $\mathbf{\Sigma}_{yy}$ represent the intra-class covariance matrices.

Using the Lagrangian multipliers, the optimal project-vectors in (2) are the eigenvectors corresponding to the largest eigenvalues of the following generalized eigensystems:

$$\begin{aligned} \mathbf{\Sigma}_{xy}\mathbf{\Sigma}_{yy}^{-1}\mathbf{\Sigma}_{yx}\mathbf{u}_x &= \lambda_x\mathbf{\Sigma}_{xx}\mathbf{u}_x \\ \mathbf{\Sigma}_{yx}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xy}\mathbf{u}_y &= \lambda_y\mathbf{\Sigma}_{yy}\mathbf{u}_y \end{aligned} \tag{3}$$

Naively, assuming that all covariance matrices are invertible, by calculating the q largest eigenvalues and the corresponding eigenvectors of (3), one can easily obtain the current CCA subspace. However, the assumption mentioned above always cannot be satisfied in object tracking due to the observations being high-dimensional, whereas the number of samples always being small size, which will lead to 3S problem.

To obtain the inverse covariance matrices while keeping all valid information, Sun et al. [11] propose to incorporate PCA, which is widely used in many applications[12, 13, 14], into the calculation of CCA for feature extraction[15]. By using PCA, one can analyze the canonical correlation in low dimensional space where the covariance matrices always being invertible, which would dramatically reduce the computation cost. Moreover, the authors also give a detailed proof to prove that calculating the CCA subspace in the low-dimensional space would not lose any valid information [11].

Inspired by the above method, we combine PCA and CCA in our P3CA tracker to solve the 3S problem and reduce the computation cost in object tracking. The initial P3CA subspace$(\mathbf{P}, \mathbf{U}_x, \mathbf{U}_y)$ can be obtained by calculating CCA on low-dimensional data which is gained by projecting observations to the initial PCA subspaces, where $\mathbf{P} = diag([\rho_1, \rho_2, ..., \rho_q])$ is a diagonal matrix whose elements are the q largest canonical correlation scores. And $\mathbf{U}_x = [\mathbf{u}_{x1}, \mathbf{u}_{x2}, ..., \mathbf{u}_{xq}]$, $\mathbf{U}_y = [\mathbf{u}_{y1}\mathbf{u}_{y2}, ..., \mathbf{u}_{yq}]$ represent the projection matrices with columns as the project-vectors.

Since the combination of PCA and CCA in [11] is used to extract image features, it cannot be used in object tracking directly. Therefore, after obtaining the initial P3CA subspace, the next two important issues are how to use P3CA in object tracking and how to update it, which we will discussed in the next two parts.

4. P3CA-BASED APPEARANCE MODEL

Our appearance model in the framework of PF is the P3CA subspace $(\mathbf{P}, \mathbf{U}_x, \mathbf{U}_y)$. And the log-likelihood of particle i at time t can be obtained by (4):

$$log(p(\mathbf{Z}_t^{i''}|\mathbf{S}_t)) = -\frac{1}{2}(\bar{\mathbf{x}}_t^{i''}\bar{\mathbf{y}}_t^{i''})\mathbf{\Gamma}^{-1}(\bar{\mathbf{x}}_t^{i''}\bar{\mathbf{y}}_t^{i''})^T \tag{4}$$

where $\bar{\mathbf{x}}_t^{i''} = \mathbf{x}_t^{i''} - \mathbf{m}_x$, $\bar{\mathbf{y}}_t^{i''} = \mathbf{y}_t^{i''} - \mathbf{m}_y$ represent the centered observation data in low dimensional space obtained by projecting the original data into the current PCA subspaces and $\mathbf{m}_x$, $\mathbf{m}_y$ represent the mean vectors of the low dimensional data.

$$\mathbf{\Gamma} = \begin{pmatrix} \mathbf{\Sigma}_{xx} & \mathbf{\Sigma}_{xx}\mathbf{U}_x\mathbf{P}\mathbf{U}_y^T\mathbf{\Sigma}_{yy} \\ \mathbf{\Sigma}_{yy}\mathbf{U}_y\mathbf{P}\mathbf{U}_x^T\mathbf{\Sigma}_{xx} & \mathbf{\Sigma}_{yy} \end{pmatrix} \tag{5}$$

5. ONLINE UPDATING OF P3CA

There are many methods for subspace online learning and updating. In this paper, we employ the Sequential Karhunen–Loeve algorithm(SKL) [16, 5] to incrementally update the PCA subspaces corresponding to the left sub-patch and the right sub-patch. And we denote the current PCA subspaces by $(\mathbf{U}_{pcax}, \mathbf{\Sigma}_{pcax})$ and $(\mathbf{U}_{pcay}, \mathbf{\Sigma}_{pcay})$.

After obtained the current PCA subspaces, the last remaining issue is how to effectively update the P3CA subspace $(\mathbf{P}, \mathbf{U}_x, \mathbf{U}_y)$ using CCA on current PCA subspaces. Since the updating of the right sub-patch $\mathbf{y}$ is similar as the left sub-patch $\mathbf{x}$ and can be done by simply interchanging $\mathbf{x}$ with $\mathbf{y}$, in the following, we will only discuss the P3CA updating for the left sub-patch. In Section 3, we know that P3CA subspace can be obtained by solving (3) on low-dimensional space, which can be written in matrix form as (6):

$$\mathbf{A}\mathbf{U}_x = \mathbf{U}_x\mathbf{P} \quad where, \quad \mathbf{A} = \mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xy}\mathbf{\Sigma}_{yy}^{-1}\mathbf{\Sigma}_{yx} \tag{6}$$

From (6), we know that, for updating P3CA subspace, the main task is to update the current covariance matrices to the new ones, when the new observations are available. Suppose that the current mean vectors and covariance matrices are denoted by $\mathbf{m}_x$, $\mathbf{m}_y$, $\mathbf{\Sigma}_{xy}$, $\mathbf{\Sigma}_{xx}^{-1}$, $\mathbf{\Sigma}_{yy}^{-1}$. When the new observations $\mathbf{Z}_1 = \{(\mathbf{x}_{t+1}, \mathbf{y}_{t+1}), (\mathbf{x}_{t+2}, \mathbf{y}_{t+2}), ...,$ $(\mathbf{x}_{t+m}, \mathbf{y}_{t+m})\}$ come in (empirically, we choose $m = 5$), the new covariance matrices, $\mathbf{\Sigma}_{xy}^{new}$, $(\mathbf{\Sigma}_{xx}^{-1})^{new}$ and $(\mathbf{\Sigma}_{yy}^{-1})^{new}$, corresponding to the entire dataset, $\mathbf{Z} = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2),$ $..., (\mathbf{x}_{t+m}, \mathbf{y}_{t+m})\}$ can be obtained as follows:

Step 1: Get the updated PCA subspaces $(\mathbf{U}_{pcax}^{'}, \mathbf{\Sigma}_{pcax}^{'})$ and $(\mathbf{U}_{pcay}^{'}, \mathbf{\Sigma}_{pcay}^{'})$ using SKL algorithm.

Step 2: Project the current mean vectors and covariance matrices obtained from the old PCA subspaces to the updated PCA subspaces, denote them by $\mathbf{m}_x^{'}$, $\mathbf{m}_y^{'}$, $\mathbf{\Sigma}_{xx}^{'}$, $\mathbf{\Sigma}_{yy}^{'}$, $\mathbf{\Sigma}_{xy}^{'}$.

Step 3: Project the new observation data $\mathbf{Z}_1$ to the updated PCA subspaces and get corresponding mean vectors and the covariance matrices: $\mathbf{m}_x^{''}$, $\mathbf{m}_y^{''}$, $\mathbf{\Sigma}_{xy}^{''}$, $\mathbf{\Sigma}_{xx}^{''}$, $\mathbf{\Sigma}_{yy}^{''}$.

Step 4: Use the mean vectors and covariance matrices gained from Step 2 and Step 3 to compute the new covariance matrices corresponding to the entire dataset as Eqs.(7)-(9):

$$\begin{aligned} \mathbf{\Sigma}_{xy}^{new} &= \frac{t}{t+m}\mathbf{\Sigma}_{xy}^{'} + \frac{m}{t+m}\mathbf{\Sigma}_{xy}^{''} \\ &+ \frac{tm}{(t+m)^2}(\mathbf{m}_x^{'} - \mathbf{m}_x^{''})(\mathbf{m}_y^{'} - \mathbf{m}_y^{''})^T \end{aligned} \tag{7}$$

$$(\mathbf{\Sigma}_{xx}^{-1})^{new} = \mathbf{D}_x^{-1} - \frac{\tilde{a}\tilde{a}^T}{\alpha}, \quad (\mathbf{\Sigma}_{yy}^{-1})^{new} = \mathbf{D}_y^{-1} - \frac{\tilde{b}\tilde{b}^T}{\beta} \tag{8}$$

where,

$$\begin{aligned} \mathbf{D}_x &= \frac{t}{t+m}\mathbf{\Sigma}_{xx}^{'} + \frac{m}{t+m}\mathbf{\Sigma}_{xx}^{''}, \\ \mathbf{D}_y &= \frac{t}{t+m}\mathbf{\Sigma}_{yy}^{'} + \frac{m}{t+m}\mathbf{\Sigma}_{yy}^{''}, \\ \alpha &= 1 + a^T\tilde{a}, \quad \beta = 1 + b^T\tilde{b}, \\ a &= \frac{\sqrt{mt}}{t+m}(\mathbf{m}_x^{'} - \mathbf{m}_x^{''}), \quad b = \frac{\sqrt{mt}}{t+m}(\mathbf{m}_y^{'} - \mathbf{m}_y^{''}), \\ \tilde{a} &= \mathbf{D}_x^{-1}a, \quad \tilde{b} = \mathbf{D}_y^{-1}b \end{aligned} \tag{9}$$

After we get the updated covariance matrices, we can obtain the updated P3CA subspace using (6).

6. EXPERIMENT RESULTS

In this section, we validate our proposed algorithm on four challenging video sequences and compare it with five state-of-the-art methods proposed in recent years. All of these sequences, *Car11*[5], *Occ2*[17], *Dollar*[17], *Girl*[17] are publicly available benchmark video sequences and can be download from their corresponding project homepage. The challenges of these sequences include severe occlusions and drastic variations of illumination, pose and scale. In order to test the effectiveness and robustness of our proposed tracker, we compare it with FragT[8], MILT[17], L1T[6], IVT[5] and CCA[9], where IVT and CCA respectively use one component of our P3CA tracker. The number of particles is set to 600 for all particle filter-based methods and we empirically choose the PCA subspace dimension 8 and CCA subspace dimension 8 in our P3CA tracker and trackers using PCA or CCA alone. For fair comparison, we use the source code provided by the authors and assume that the initial bounding box of the target object for all methods is same and is specified manually beforehand.

Comparative tracking results for selected frames are shown in Fig. 2, from which we can see that our P3CA tracker performs very well on all these challenging sequences. The CCA tracker implemented by ourselves, which also adopts the correlation-based appearance model, solves the 3S problem by adding a random disturbance to the covariance matrix to make it invertible [18], which will lose accuracy due to approximation. Thus, CCA tracker achieves the second place on average performance over all sequences. IVT performs well in dealing with variations of illumination in *Car11* shown in Fig. 2(a), but it is less effective in handling heavy occlusion for adopting holistic appearance model as in sequences of *Occ2*, *Girl* and *Dollar* shown in Fig. 2(b), Fig. 2(c) and Fig. 2(d). FragT performs well in handling occlusion caused by dissimilar object because it is specifically designed to handle occlusions via part-based model. But, it is not very effective when there exists occlusions caused by similar object and appearance variations caused by wearing a hat as in sequences of *Dollar* and *Occ2* shown in Fig. 2(d) and Fig. 2(b). Moreover, tracking results on *Car11* demonstrate that FragT is unable to handle severe variation of illumination. L1T is a template-based method using sparse representation, which is robust to partial occlusion and illumination variation. But it is less effective when the scenario is very complex as shown in Fig. 2(a), Fig. 2(b) and Fig. 2(d). MILT models the object appearance using online Multiple Instance Learning method and it can handle occlusion, illumination change and pose change. But it cannot achieve very accurate tracking results when there exists rotation and scale change for lacking the representation of rotation angle and scale in the object state. Moreover, tracking results on *Girl* sequence demonstrates that MILT cannot handle the long duration and severe occlusion caused by object with similar appearance.

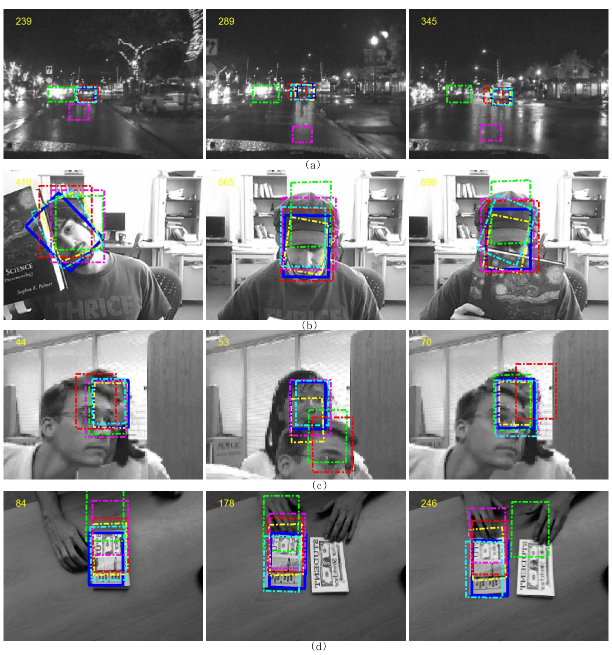

Fig. 2: Comparison tracking results of P3CA tracker (blue solid box) with the CCA tracker (cyan dashed box), the IVT tracker (yellow dashed box), the Fragment-based tracker (magenta dashed box), the MIL tracker (red dashed box), and the $\ell 1$ tracker (green dashed box) on four video sequences. (a)*Car11* (b)*Occ2* (c)*Girl* (d)*Dollar*

We also measured the quantitative tracking error, the Euclidean distance from the tracking center to the ground-truth (labeled by ourselves), to evaluate the robustness of a method. The center error plots of these 6 methods on 4 representative sequences are shown in Fig. 3, which demonstrate that our tracker is very robust in handling occlusions, illumination changes, pose and scale changes even in very complex scenario. Moreover, we show the average center errors in Table 1, which shows that our tracker gives the best tracking results on 3 sequences and the second place on the other sequence.

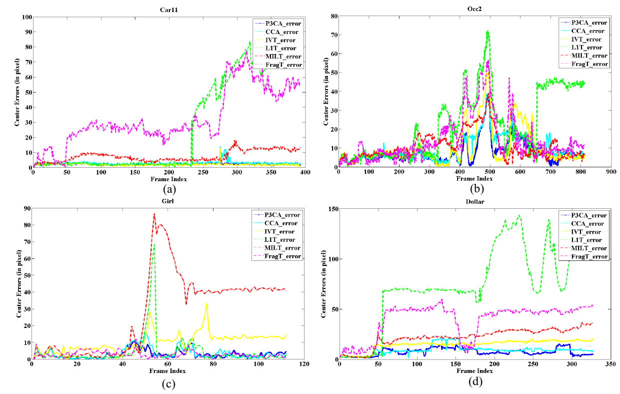

Fig. 3: Center error plots.

Table 1: Location error(in pixel, the bold font indicate the best performance)

seq	P3CA	CCA	IVT	FragT	MILT	L1T
Car11	2.2	2.8	**1.9**	32.7	7.5	26.9
Occ2	**7.7**	9.4	10.0	14.1	10.8	22.5
Girl	**2.9**	3.2	10.0	2.9	28.5	4.4
Dollar	**7.2**	9.1	14.5	39.5	22.0	75.8

Running on a standard 3.4GHz machine, our P3CA tracker costs 79.7ms per frame on average. Although it is not as effective as IVT (40.3ms) and MILT (73.8ms), it is much faster than the other three methods, which are: CCA (139.8ms), FragT (267.5ms), L1T (5322.5ms).

7. CONCLUSION

In this paper, we have presented a novel correlation-based online tracker called P3CA, which takes full advantages of PCA and CCA and avoids their demerits at the same time. In contrast to methods based-on holistic information, our tracker is more robust in handling occlusion for using the canonical correlation score to evaluate the target goodness. Moreover, unlike the previous correlation-based approaches, our P3CA tracker adopts PCA into the generation and updating of CCA, which can enable the tracker to escape from 3S problem and reduce the computation cost at the same time. In addition, we proposed an incrementally learning method to update our P3CA appearance model online, leading our tracker to be robust to various appearance variations. Qualitative and quantitative experiment results on different challenging sequences demonstrate that our tracker is very robust to the environments with partial occlusion and various appearance variations.

8. REFERENCES

[1] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, 2003.

[2] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proc. European Conf. Computer Vision*. 2002, pp. 661–675, Springer.

[3] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *Int'l J. Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.

[4] Li Xi, Hu Weiming, Zhang Zhongfei, Zhang Xiaoqin, and Luo Guan, "Robust visual tracking based on incremental tensor subspace learning," in *Proc. IEEE Int'l Conf. Computer Vision*, 2007, pp. 1–8.

[5] David Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang, "Incremental learning for robust visual tracking," *Int'l J. Computer Vision*, vol. 77, no. 1, pp. 125–141, 2008.

[6] X. Mei and H. Ling, "Robust visual tracking using l1 minimization," in *Proc. IEEE Int'l Conf. Computer Vision*, 2009, pp. 1436–1443.

[7] S. Avidan, "Ensemble tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 2, pp. 261–271, 2007.

[8] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006, vol. 1, pp. 798–805.

[9] Minyoung Kim, "Correlation-based incremental visual tracking," *Pattern Recognition*, vol. 45, no. 3, pp. 1050–1060, 2012.

[10] F.S. Wang and Q.J. Zhao, "A new particle filter for nonlinear filtering problems," *Chin. J. of Computers*, vol. 3, no. 2, pp. 346–352, 2008.

[11] Q.S. Sun, S.G. Zeng, P.A. Heng, and D.S. Xia, "The theory of canonical correlation analysis and its application to feature fusion," *Chin. J. of Computers*, vol. 28, no. 9, pp. 1524–1533, 2005.

[12] Q.J. Zhao and Z.Q. Sun, "Image-based robot motion simulation," *Optics communications*, vol. 205, no. 4-6, pp. 257–263, 2002.

[13] J. Shlens, "A tutorial on principal component analysis," *Systems Neurobiology Laboratory, University of California at San Diego*, 2005.

[14] Q.J Zhao, Z.Q. Sun, F.C. Sun, and J.H Zhu, "Appearance-based robot visual servo via a wavelet neural network," *Int'l J. Control, Automation, and Systems*, vol. 6, no. 4, pp. 607–612, 2008.

[15] Zhiquan Feng, Bo Yang, Yuehui Chen, Yanwei Zheng, Tao Xu, Yi Li, Ting Xu, and Deliang Zhu, "Features extraction from hand images based on new detection operators," *Pattern Recognition*, pp. 1089–1105, 2011.

[16] A. Levy and M. Lindenbaum, "Sequential karhunen-loeve basis extraction and its application to images," in *Proceedings of International Conference on Image Processing*. 1998, vol. 2, pp. 456–460, IEEE.

[17] B. Babenko, M.H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009, pp. 983–990.

[18] Z.Q. Hong and J.Y. Yang, "Optimal discriminant plane for a small number of samples and design method of classifier on the plane," *Pattern Recognition*, vol. 24, no. 4, pp. 317–324, 1991.