Hindawi
Journal of Electrical and Computer Engineering
Volume 2018, Article ID 5381962, 9 pages
https://doi.org/10.1155/2018/5381962

Research Article

Target Tracking via Particle Filter and Convolutional Network

Hongxia Chu,[1,2] Kejun Wang,[1] and Xianglei Xing[1]

[1] *College of Automation, Harbin Engineering University, Harbin, China*
[2] *College of Electrical and Information Engineering, Heilongjiang Institute of Technology, Harbin, China*

Correspondence should be addressed to Hongxia Chu; chx0420@163.com

Received 3 June 2017; Revised 17 August 2017; Accepted 14 November 2017; Published 9 January 2018

Academic Editor: Tongliang Liu

We propose a more effective tracking algorithm which can work robustly in a complex scene such as illumination, appearance change, and partial occlusion. The algorithm is based on an improved particle filter which used the efficient design of observation model. Predefined convolutional filters are used to extract the high-order features. The global representation is generated by combining local features without changing their structures and space arrangements. It not only increases the feature invariance, but also maintains the specificity. The extracted feature from convolution network is introduced into particle filter algorithm. The observation model is constructed by fusing the color feature of the target and a set of features from templates which are extracted by convolutional networks without training in our paper. It is fused with the features extracted from convolutional network for tracking. In the process of tracking, the template is updated in real time, and then the robustness of the algorithm is improved. Experiments show that the algorithm can achieve an ideal tracking effect when the targets are in a complex environment.

1. Introduction

Object tracking has a wide application prospect in computer vision. Recently, many researchers have carried out a lot of research on it in the real world [1]. Detecting and tracking the target is a very difficult task in practical application [2]. Many factors can impact the performance of the tracking algorithm. These issues are made up of attitude change, appearance variation owing to illumination changes, partial occlusion, and background noise [3, 4]. To solve all these problems, we need more efficient machine learning [5] and feature extraction [6] to describe the target.

At present, the tracking algorithms mainly include two types: generative model and discriminative model [5]. Particle filter is one of the representatives of generative tracking algorithm. Particle filters have been used widely in the tracking problem. Particle filter algorithm has the advantage of simplicity and flexibility. And it is easy to handle non-Gaussian and multimodality system model. There are many related literatures presented in [7–11]. The information from different measurement sources can be used in the framework of particle filter. This has greatly improved the tracking performance. But in the actual process of tracking, there are still a lot of ways to improve the effectiveness of the tracking algorithm. In addition, the classical particle filter usually adopts the dynamic model with global information. Regardless of whether the target is blocked or deformed, it treats the target as a whole. This leads to the neglect of the local information of the target. When the target is partially occluded and local appearance of it changes, particle filter algorithm cannot accurately track the target.

Discriminative methods are used to distinguish targets and backgrounds by training classifiers. At present, most of deep learning methods are also attributed to discriminative frames in the target tracking. Deep learning has made outstanding achievements in the field of image classification and target detection. It has become one of the most powerful automatic feature extraction methods. The deep network can get high-level abstract features gradually from low-level features through learning and mapping of multiple levels. These abstract features have high dimension and strong distinction. High accuracy of classification and regression tasks can be achieved by using simple classifier. At present, some tracking methods based on learning feature have been proposed, using convolutional networks trained offline [12, 13]. In tracking, the target localization is achieved by intercepting the characteristics of the target in different layers of the network.

The key point in all of these methods is how to learn an effective feature extraction offline with a great deal of auxiliary data, and it consumes a lot of time. The methods also have given no consideration to the similar local structure and inner geometry distribution information between the targets over consequent frames, which is handy and effective in distinguishing the target from background for visual tracking. In addition, the pure use of the deep learning method does not solve the problem of tracking drift, and it needs to be combined with other methods in order to better play the role of the depth network [14, 15].

In summary, a target tracking algorithm combining particle filter and convolution network is proposed in the paper. The extracted features from convolutional networks are introduced into the particle filter framework. The target block is represented by sparse representation. The local information and spatial information of the target are fully exploited to represent the state change of the object. According to the target state, different information is dealt with. Because the global pieces of information of particle filter are combined to determine the position of the current target, the local appearance change and partial occlusion problem of the target are better solved. And in the tracking process, the template is updated according to the tracking results, which improves the robustness of the algorithm to a certain extent. Experiments show that when the target is in a complex environment, the algorithm can achieve an ideal tracking effect.

2. Particle Filtering Tracking Formula

The tracking problem for particle filter is to estimate the posterior probability density at the t moment, which is obtained by two steps [10].

Step 1 (prediction). First, suppose the initial value $p(s_0)$ of the probability density is known and the posterior probability density function $p(s_{t-1} \mid o_{1:t-1})$ is also known at the $t-1$ moment. $\mathbf{s}_t$ is described as a three-dimension vector; $\mathbf{s}_t = [s_t^x, s_t^y, s_t^s]^T$. s_t^x, s_t^y express the position of object. s_t^s expresses the size change of the object. Then the prior probability $p(s_t \mid o_{1:t-1})$ is

$$p\left(s_t \mid o_{1:t-1}\right) = \int p\left(s_t \mid s_{t-1}\right) p\left(s_{t-1} \mid o_{1:t-1}\right) ds_{t-1}, \tag{1}$$

where $p(s_t \mid s_{t-1})$ is defined by the state equation of the target.

Step 2 (updating). Finally, $p(s_t \mid o_{1:t})$ is obtained from the observation model of the system.

$$\begin{aligned} p\left(s_t \mid o_{1:t}\right) &= p\left(s_t \mid o_t, o_{1:t-1}\right) \\ &= \frac{p\left(o_t \mid s_t, o_{1:t-1}\right) p\left(s_t \mid o_{1:t-1}\right)}{p\left(o_t \mid o_{1:t-1}\right)} \\ &= \frac{p\left(o_t \mid s_t\right) p\left(s_t \mid o_{1:t-1}\right)}{p\left(o_t \mid s_{1:t-1}\right)}. \end{aligned} \tag{2}$$

The observation likelihood function $p(o_t \mid s_t)$ is determined by the observation of the target. $p(o_t \mid o_{1:t-1})$ is a normalization constant.

In fact, since the integral of formula (1) is difficult to realize, the recursive Bayesian filtering (i.e., particle filter) is simulated by the nonparametric Monte Carlo method. The basic formula is

$$p\left(s_t \mid o_{1:t}\right) \approx \sum_{i=1}^{N} w_t^i \delta\left(s_t - s_t^i\right), \tag{3}$$

where w_t^i is the weight of the corresponding particle. The weight of the particle is updated according to the observation value. That is,

$$w_t^i = w_{t-1}^i \frac{p\left(o_t \mid s_t^i\right) p\left(s_t^i \mid s_{t-1}^i\right)}{q\left(s_t^i \mid s_{t-1}^i, o_t\right)}, \quad \sum_{i=1}^{N} w_t^i = 1, \tag{4}$$

where $q(s_t^i \mid s_{t-1}^i, o_t)$ is the proposed distribution (importance density) function in Bayesian importance sampling. The optimal choice is to select the proposed distribution as a priori density. That is, $q(s_t^i \mid s_{t-1}^i, o_t) = p(s_t^i \mid s_{t-1}^i)$. Then the weight is

$$w_t^i = w_{t-1}^i p\left(o_t \mid s_t^i\right), \quad \sum_{i=1}^{N} w_t^i = 1. \tag{5}$$

Finally, the real state estimate of the target is obtained; that is,

$$\widehat{s}_t = \sum_{i=1}^{N} s_t^i w_t^i. \tag{6}$$

3. Model Construction and Implementation

3.1. Target Motion Model. For images in a new frame, each particle carries out state transition according to the following motion model:

$$X_t = X_{t-1} + \mu_t \zeta_t, \tag{7}$$

where ζ_t is the Gauss white noise and μ_t is the propagation radius the particle, which is proportional to the average state change of the target at the previous moment, $\mu_t = (1/k)\sum_{n=t-k}^{t} |X_n - X_{n-1}|$.

3.2. Target Observation Model. Each input image is divided into a fixed size of $n \times n$ pixels, denoted as $I \in \mathbb{R}^{n \times n}$. A set of local image blocks $y \in \{Y_1, \ldots, Y_l\}$ is obtained by densely sampling through sliding a window of size $k \times k$ (k is called the receptive field size). $Y_i \in \mathbb{R}^{k \times k}$ is the ith image block and $l = (n-k+1) \times (n-k+1)$. Each block Y_i is preprocessed through subtracting the mean and ℓ_2 normalization, respectively [11].

3.2.1. Color Model

(1) Target Model. s_i, $i = 1 : n$ is vectorized image patches, with zero as the center. The number of Eigen values' bin is m. The probability of the Eigen value of the target model is [16]

$$\begin{aligned} &\widehat{Q} = \left\{\widehat{Q}_a\right\}_{a=1,\ldots,m}, \\ &\widehat{Q}_a = C_1 \sum_{i=1}^{n} k\left(\left\|s_i^*\right\|^2\right) \delta\left[B\left(s_i^*\right) - a\right], \end{aligned} \tag{8}$$

where $k(s_i)$ is kernel function used to adjust the size of the weights, δ is delta function, $B(s_i^*)$ represents the color value of pixels at s_i^*, a is the color index of the histogram, and $C_1 = 1/\sum_{i=1}^{n} k(\|x_i^*\|^2)$ is normalized constant coefficient.

(2) Target Candidate Model. For empathy, taking y as the center, the probability of the target candidate model is

$$\widehat{P}(y) = \left\{\widehat{P}_a(y)\right\}_{a=1\cdots m},$$
$$\widehat{P}_a(y) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{y - s_i^*}{h}\right\|^2\right) \delta\left[B(s_i^*) - a\right], \tag{9}$$

where window radius is h and $C_h = 1/\sum_{i=1}^{n_h} k(\|(y - s_i)/h\|^2)$ is normalized constant coefficient.

(3) Similarity Function. In the paper, Bhattacharyya coefficients are used to calculate the similarity functions of two models [17]:

$$\widehat{\rho}(y) = \rho\left[\widehat{P}(y), \widehat{Q}\right] = \sum_{a=1}^{m} \sqrt{\widehat{P}_a(y)\,\widehat{Q}_a}. \tag{10}$$

It has a value of 0~1. Then we suppose the distance between the two target templates as

$$D(y) = \sqrt{1 - \rho\left[\widehat{P}(y), \widehat{Q}\right]}. \tag{11}$$

The corresponding color observation probability of the particle is obtained:

$$p_1\left(\mathbf{o}_t \mid \mathbf{s}_t^i\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{D^2\left[\widehat{P}(y), \widehat{Q}\right]}{2\sigma^2}\right), \tag{12}$$

where mean square deviation is $\sigma = 0.2$.

3.2.2. Convolutional Networks Model. In order to describe the target better, we applied convolutional networks to learn robust representations for visual tracking without offline training using a large amount of auxiliary data, which is inspired by recent studies [11, 18]. First, we use predefined convolutional filters to extract the high-order features. Second, we generate a global representation by combining local features which their structures and space arrangements need not be changed. So it increases feature invariance while maintaining specificity.

Step 1 (target layer). The k-means clustering method is used to cluster out d filters as a convolution kernel from $l = (n - k + 1) \times (n - k + 1)$ patch. The patch is $\mathsf{F}^o = \{F_1^o, \ldots, F_d^o\} \subset y$. Given the ith filter $\mathsf{F}_i^o \in \mathbb{R}^{k\times k}$, its response on the input image I is denoted with a feature map $U_i^o \in \mathbb{R}^{(n-k+1)\times(n-k+1)}$, where $U_i^o \in F_i^o \otimes I$ and $\otimes$ is the convolution operator.

Step 2 (background layer). At the same time, the useful background information around the target is used to distinguish the target from the background. M samples are selected around the background target, and k-means is used to select a bank of filters $\mathsf{F}_i^b = \{F_{i,1}^b, \ldots, F_{i,d}^b\} \subset y$. We use the average pooling method to summarize each filter in F_i^b, and generate the background context filter set $\mathsf{F}^b = \{F_1^b = (1/m)\sum_{i=1}^{m} F_{i,1}^b, \ldots, F_{i,d}^b = (1/m)\sum_{i=1}^{m} F_{i,d}^b\}$. Then it does the convolution with the input image $U_i^b \in F_i^b \otimes I$. Finally, the simple cell feature maps are defined as

$$U_i = U_i^o - U_i^b = \left(F_i^o - F_i^b\right) \otimes I, \quad i = 1, \ldots, d. \tag{13}$$

Step 3 (convolution layer). At first, simple cell feature map consists of the filter set $F = F^o \cup F^b$. Then d different feature maps are stacked to construct a three-dimensional tensor $\mathbf{V} \in \mathbb{R}^{(n-k+1)\times(n-k+1)\times d}$, that is, the combination of the characteristic graphs. This kind of specificity has the characteristic of shift and sensitivity. In addition, the warp region is $n \times n$, which makes the characteristic of the target scale robust.

To increase the robustness of appearance change, we represent the feature $\mathbf{V}$ by using the sparse representation.

$$\widehat{\mathbf{v}} = \arg\min_{\mathbf{v}} \|\mathbf{v} - \operatorname{vec}(\mathbf{V})\|_2^2 + \lambda_1 \|\mathbf{v}\|_1 + \lambda_2 \|\mathbf{v}\|_2^2. \tag{14}$$

Then the solution of the model can be solved by using the method of soft shrinkage [19]:

$$\begin{aligned}\widehat{\mathbf{v}} &= \operatorname{soft}\left(\operatorname{vec}(\mathbf{V}), \lambda c_i\right)\\ &= \operatorname{sign}\left(\operatorname{vec}(\mathbf{V})\right) \max\left(0, |\operatorname{vec}(\mathbf{V})| - \lambda c_i\right),\end{aligned} \tag{15}$$

where $\operatorname{sign}(\cdot)$ is a sign function and λ is set to median value of it and $\mathbf{C} = [c_1, c_2, \ldots, c_i, \ldots]$, with $c_i \geq 0$.

Step 4 (model update). The update strategy is as follows. It is a low pass filtering form, in which $\mathbf{v}_t$ is the target template at frame t, $\mathbf{v}_{t-1}$ is the characteristic of the upper frame, and $\widehat{\mathbf{v}}_{t-1}$ is the sparse expression of $\mathbf{v}_{t-1}$.

$$\mathbf{v}_t = (1 - \rho)\,\mathbf{v}_{t-1} + \rho\widehat{\mathbf{v}}_{t-1}, \tag{16}$$

where ρ is a learning parameter.

Observation model is defined by (17) in convolution model.

$$p_2\left(\mathbf{o}_t \mid \mathbf{s}_t^i\right) \propto \exp^{-\|\mathbf{v}_t - v_t^i\|_2^1}, \quad i = 1, \ldots, r, \tag{17}$$

where $\mathbf{v}_t^i = \operatorname{vec}(\mathbf{v}_t^i) \odot \mathbf{W}$.

$\mathbf{v}_t^i$ is the ith candidate sample representation at t frame based on the complex cell features, where $\odot$ expresses the product of elements and $\mathbf{W}$ is an indicator function whose element is defined as

$$W_i = \begin{cases} 1, & \text{if } \mathbf{v}_t(i) \neq 0 \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

3.2.3. System Observation Model. The system observation probability density function of each particle is

$$\begin{gathered} p\left(\mathbf{o}_t \mid \mathbf{s}_t^i\right) = r \cdot p_1\left(\mathbf{o}_t \mid \mathbf{s}_t^i\right) + (1 - r)\,p_2\left(\mathbf{o}_t \mid \mathbf{s}_t^i\right) \\ 0 \leq r \leq 1. \end{gathered} \tag{19}$$

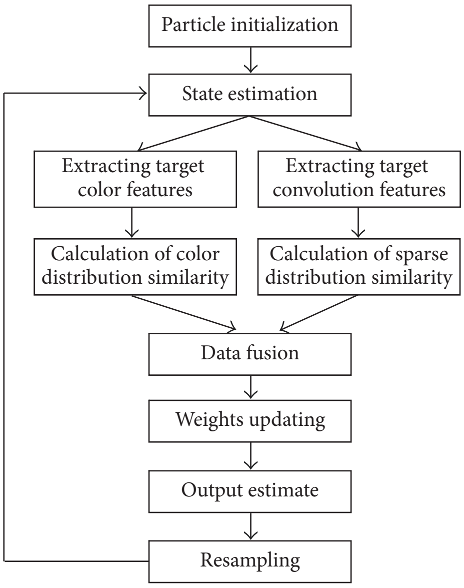

Particle initialization
State estimation
Extracting target color features
Extracting target convolution features
Calculation of color distribution similarity
Calculation of sparse distribution similarity
Data fusion
Weights updating
Output estimate
Resampling

Figure 1: Flow chart of the algorithm.

The parameter r is used to regulate the proportion of the observed probability of each feature in the total observed probability. When the background is complex and the target is partially occluded, the global positioning advantage of color distribution should be fully exploited, and the r value should be increased at this time. When the color of the target is different from the background color, the r value should be reduced, and the localization advantage of the convolution feature can be fully exploited. Under normal circumstances, we take $r < 0.5$.

The flow chart of particle filter algorithm based on feature fusion is displayed in Figure 1.

4. Experimental Results

4.1. Implementation Parameters Setup. We utilized the two-layer convolutional network as a feature extractor in the experiment. We made $l = (36 - 6 + 1) * (36 - 6 + 1)$ warped image patches obtained at the initial frame. The size of the warped image is set to 32×32 ($n = 32$). We set the receptive field size as 6×6 ($k = 6$) and set the number of filters as $d = 100$. The learning parameter ρ in (16) is set to 0.95 and the template is updated every frame. The standard deviations of the target state of the particle filter are set as follows: $\sigma_x = 4$, $\sigma_y = 4$, $r = 0.4$, and $N = 600$ particles are used. We compared the improved algorithm with five other tracking algorithms (CNT [11], MS [16], MTT [20], VTS [21], and CPF [22]). These five algorithms all use the particle filter as the search mechanism, and MTT, CNT, and VTS are sparse representation.

4.2. Evaluation Metrics. The tracking benchmark dataset [23] is used for the experimental validation. To better analyze the performance of the algorithm, we employ the videos with 11 attributes from complex scene [11].

In order to quantitatively evaluate the system, we use two graphs including the success plot and the precision plot. The success plot is calculated by the overlap rate: $S = \text{Area}(\text{BT} \cap \text{BG})/\text{Area}(\text{BT} \cup \text{BG})$. BT denotes the tracked boundary frame and BG denotes the ground truth. The percentage of frames with $S > t_0$ throughout all threshold $t_0 \in [0, 1]$ is used to express success rates. At the same time, the precision plot shows the percentage of frames between the given threshold distance and the ground truth over the tracked positions. In order to rank the trackers, we set the threshold to 20 pixels in the precision mark represented. One-pass evaluation (OPE) is used to express the average success and precision ratio of target state in the ground truth [23].

4.3. Quantitative Analysis

(1) Whole Performance. We give the performance of the top 6 implemented tracking algorithms in Figure 2 according to success and precision plots. It must be pointed out that all the graphics are produced applying the benchmark evaluation [23]; our proposed algorithm ranks firstly based on the success rate while it ranks secondly based on the precision rate. Note that the proposed algorithm exploits only simple sparse image representation that encodes local structural and geometric layout information of the target, and achieves competitive performance compared to other methods. Furthermore, even using only specific target information from the first frame without learning with auxiliary training data, our method performs well in contrast to the other methods. This is mainly because the generic features learned offline from numerous auxiliary data may not adapt well to object appearance variations in target tracking.

(2) Performance of Basic Attribute. To analyze the performance of the improved method, we need to use different attributes to estimate the tracker for videos. In this paper, 11 attributes are selected. Figures 3 and 4 are, respectively, the success plots and the correspondent precision plots [24]. We note that our improved algorithm ranks within top 1 on 3 out of multiple attributes in success plots. And the precision plots rank within top 1 on 3.

Our method also ranks first for the video sequences with the low resolution in all estimated trackers. It is difficult to extract useful characters of the targets when the resolution of the videos is low. In contrast, our algorithm extracts dense information across the entire target region by convolution operators to divide the target from the complex scene.

Our method ranks second which follows the CNT methods on the video sequences with background clutter attributes. The proposed algorithm uses background context information that is updated online and pooled in every frame and hence provides effective features to precisely locate objects from the clutters.

4.4. Qualitative Analysis

(1) The Variation of the Illumination and Posture. Figure 5 shows the successful tracking results on shaking and skating_1 seq. When great changes have taken place in the stage lighting conditions, the posture of the target is drastically changed owing to dancing or head shaking. Our algorithm effectively

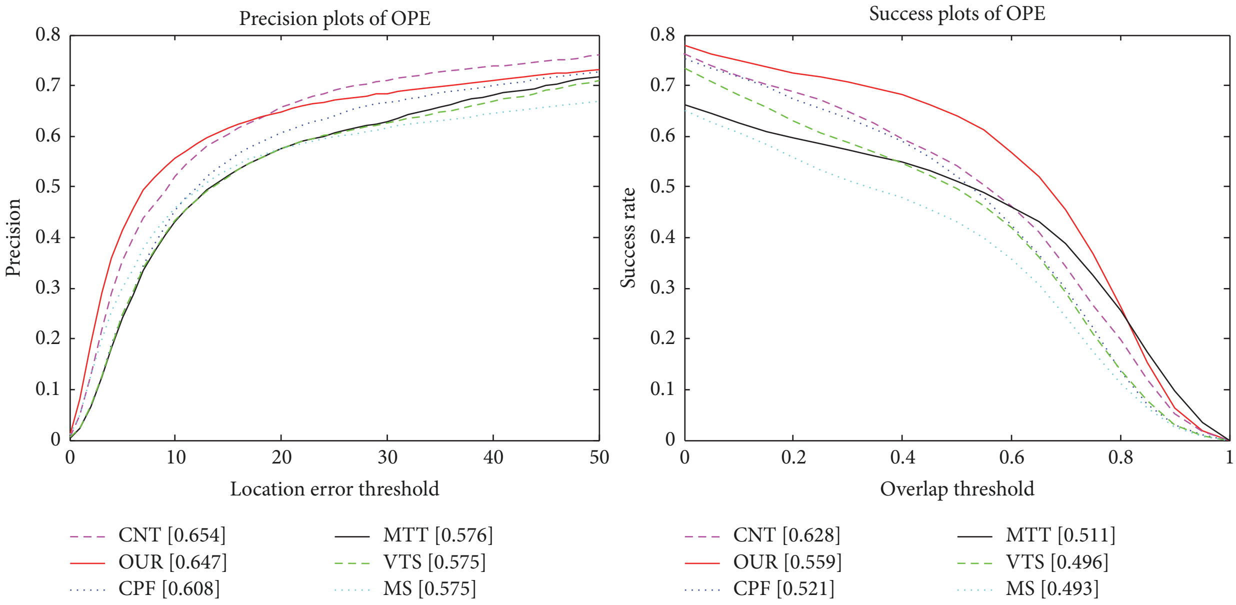


Figure 2: OPE of the success plots and precision plots in the top 6 trackers.

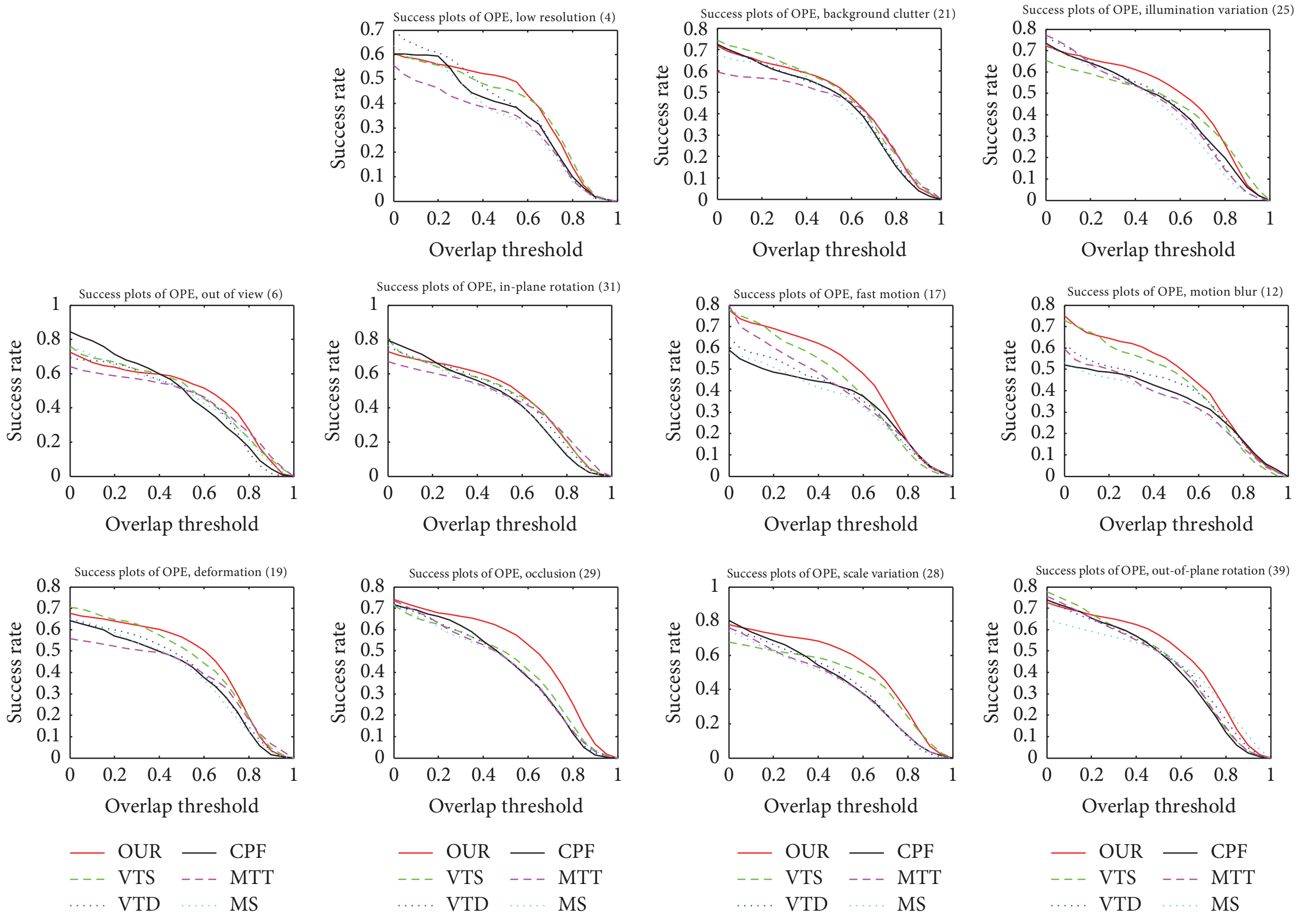


Figure 3: The success plots of videos with different attributes.

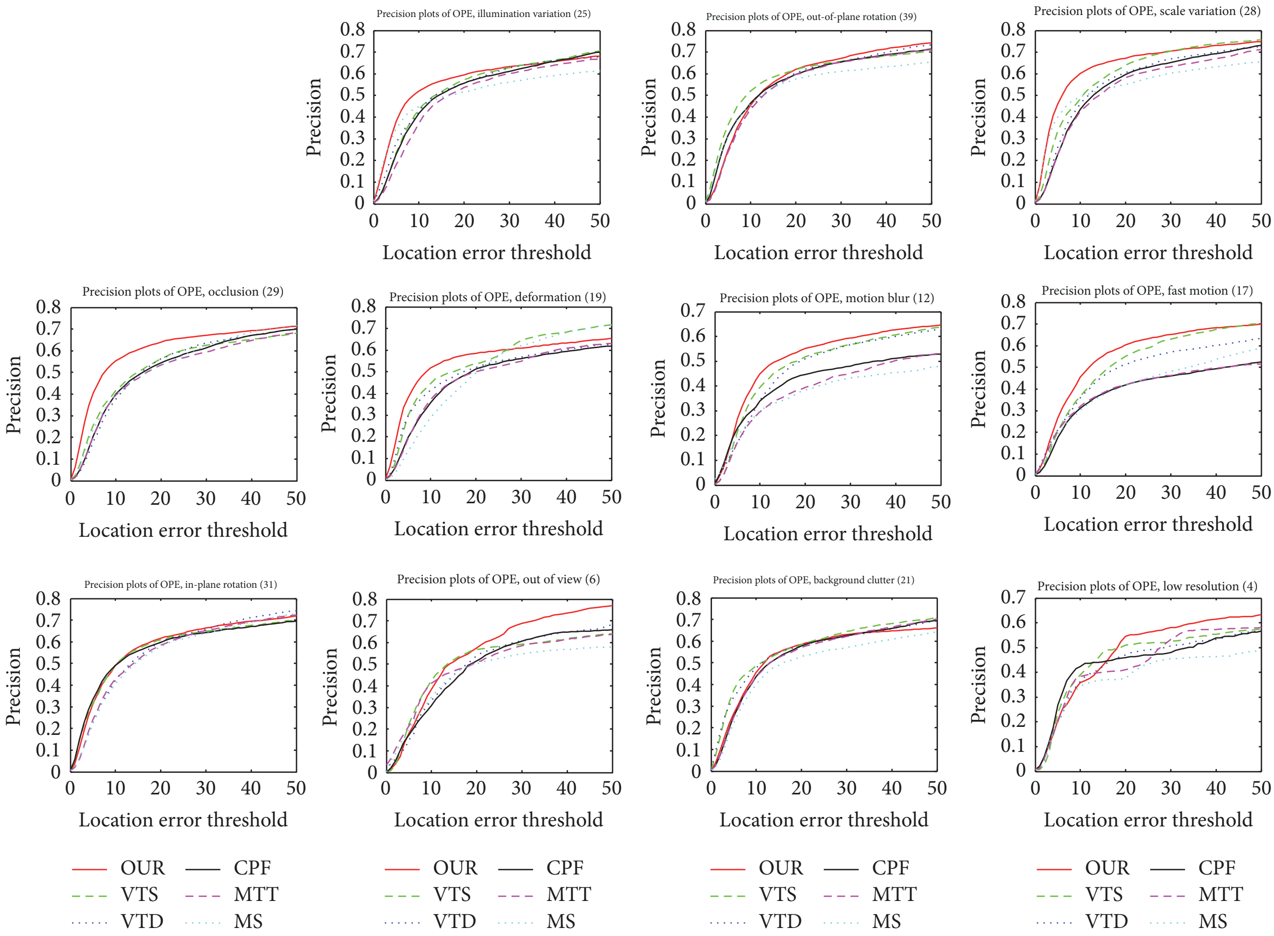


Figure 4: The precision plots of videos with multiple attributes.

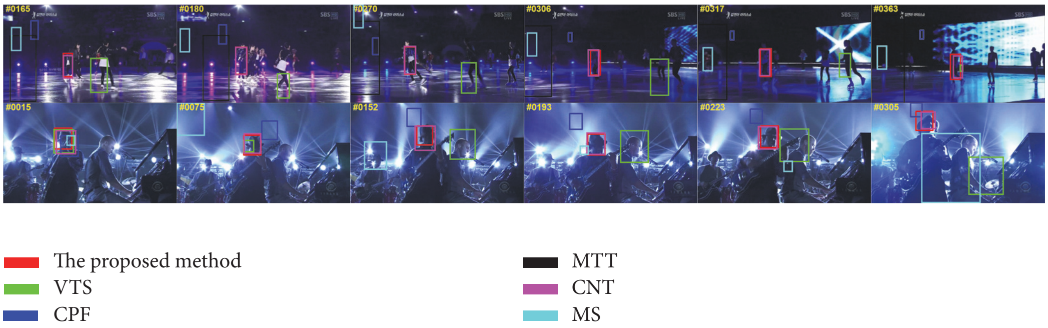


Figure 5: Qualitative evaluation of the tracker on sequences shaking_1 and skating 1.

solved the posture changes because the observation model is developed through online updating. In addition, the proposed algorithm is robust with regard to illumination changes because the observation model uses a hybrid template. But other algorithms fail to track the target while the light changes exist simultaneously.

(2) Occlusion. Figure 6 gives the successful tracking results of the improved algorithm while the object is occluded seriously by other objects. Our method and CNT can track robustly woman and jogging-1 seq. And the positioning of our method is more accurate. The two methods are robust in the presence of occlusion because of the efficient observation models. The model used local features and recorded the appearance of the target over time including the occlusion, appearance change, and the mixing of the two. Furthermore, it is responsible for various occlusions. But other algorithms fail to accurately track the targets.

(3) Background Clutters. carDark is tested in Figure 7. Background clutter is drastic, and its appearance is similar to the target itself. Under these circumstances, other tracking

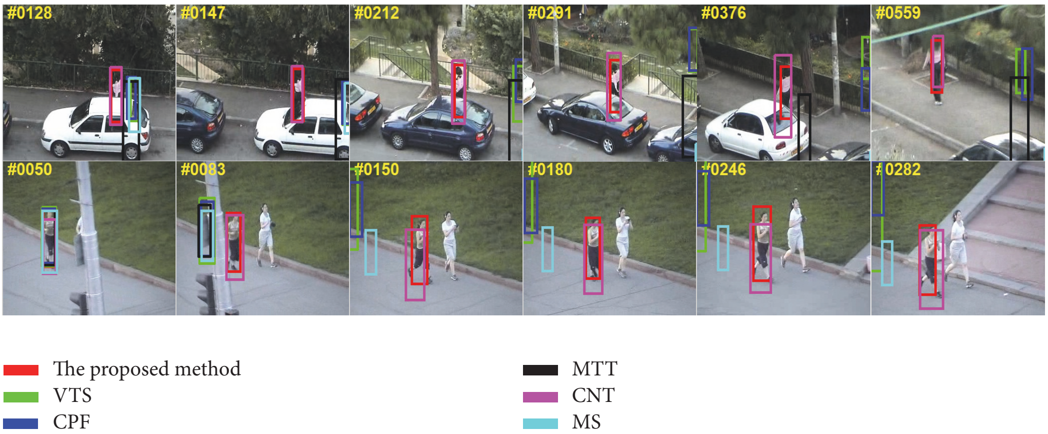


Figure 6: Qualitative evaluation about the trackers on sequences woman and jogging-1 seq.

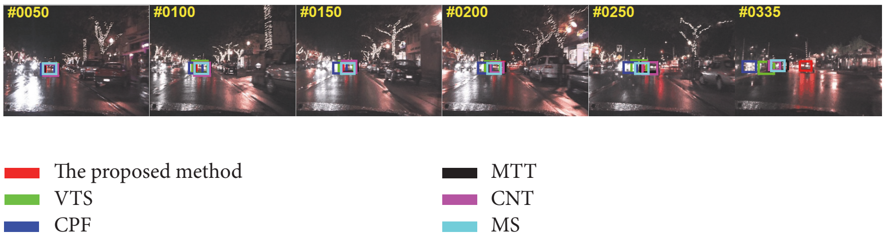


Figure 7: Qualitative evaluation about the tracker for sequences carDark seq.

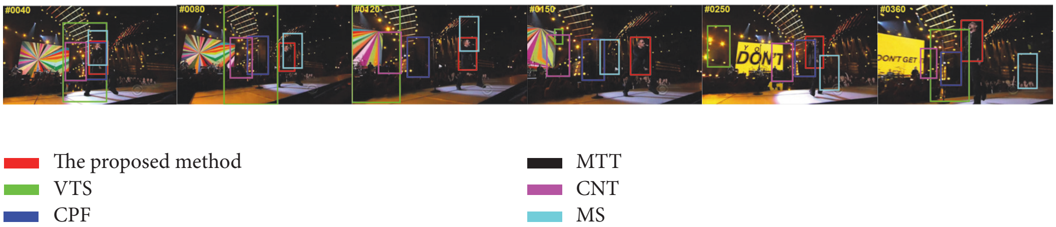


Figure 8: Qualitative evaluation about of the tracker over sequences singer2 seq.

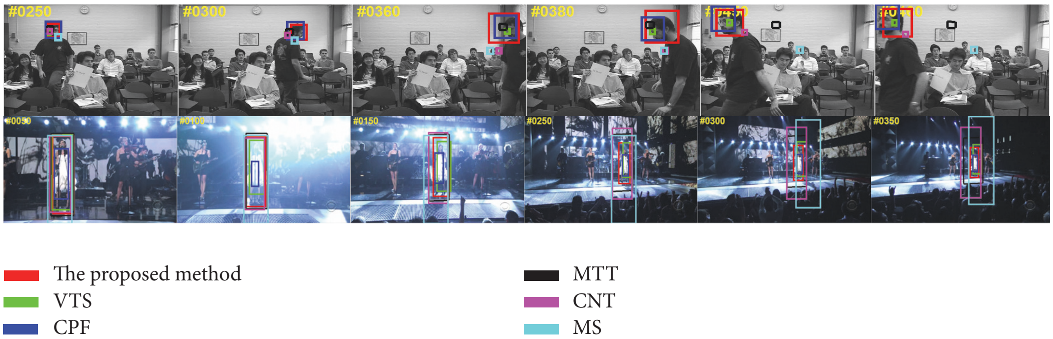


Figure 9: Qualitative evaluation about the tracker over sequences freeman3 and singer1 seq.

methods fail because of the interference of similar targets around the car. But our algorithm succeeds in tracking.

(4) Deformation. Some successful tracking results of Singer2 seq are shown in Figure 8. There are illumination changes and deformation in sequences. Only our method performs well in all of the sequences.

(5) Scale Changes. The tracking results of freeman3 and singer1 seq are shown in Figure 9. In sequences, tracked

targets have serious scale changes. For the freeman3 sequence, a person moves towards the camera with a large scale variation in his face appearance. Furthermore, the appearance also varies as the posture changes. The MTT, CPF, MS, and VTS algorithms deviate from tracked targets from #330, whereas the proposed and CNT algorithms succeed in tracking. The MTT, CPF, MS, and VTS algorithms do not efficiently complete tracking when the target has a large scale change in the singer1 sequence. But our proposed method and CNT achieve better performance. The proposed algorithm effectively solves scale variation because the representation of model is built on scale-invariant complex cell features.

5. Conclusion

We put forward an effective tracking method by using particle filter and convolutional network. Deep learning method is used to extract effective features for robust tracking. The algorithm efficiently solves the problems for appearance changing and occlusion severely. The experimental results showed that the improved method is better than traditional tracking methods in drastic tracking surroundings. Since the algorithm is extended easily by adding some more effective feature information, the tracking results could be enhanced further.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the Research Foundation of Young Resherve Talents Project in Scientific and Technical Bureau in Harbin (no. 2017RAQXJ134) and the project of the National Natural Science Foundation of China (no. 61573114).

References

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, article 13, 2006.

[2] X. Ben, W. Meng, and R. Yan, "Dual-ellipse fitting approach for robust gait periodicity detection," *Neurocomputing*, vol. 79, pp. 173–178, 2012.

[3] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1296–1311, 2003.

[4] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 125–141, 2008.

[5] T. Liu and D. Tao, "Classification with Noisy Labels by Importance Reweighting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 3, pp. 447–461, 2016.

[6] T. L. Liu, Q. Yang, and D. C. Tao, "Understanding how feature structure transfers in transfer learning," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pp. 2365–2371, Melbourne, Australia, 2017.

[7] M. Nieto, A. Cortés, O. Otaegui, J. Arróspide, and L. Salgado, "Real-time lane tracking using Rao-Blackwellized particle filter," *Journal of Real-Time Image Processing*, vol. 11, no. 1, pp. 179–191, 2016.

[8] M. Lucena, J. M. Fuertes, and N. P. de la Blanca, "Optical flow-based observation models for particle filter tracking," *PAA. Pattern Analysis and Applications*, vol. 18, no. 1, pp. 135–143, 2015.

[9] M. E. Yildirim, I. F. Ince, Y. B. Salman, J. K. Song, J. S. Park, and B. W. Yoon, "Direction-Based Modified Particle Filter for Vehicle Tracking," *ETRI Journal*, vol. 38, no. 2, pp. 356–365, 2016.

[10] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1269–1276, June 2010.

[11] K. Zhang, Q. Liu, Y. Wu, and M.-H. Yang, "Robust visual tracking via convolutional networks without training," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1779–1792, 2016.

[12] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 4293–4302, July 2016.

[13] L. J. Wang, W. L. Ouyang, X. G. Wang, and H. C. Lu, "Visual tracking with fully convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '15)*, pp. 3119–3127, Santiago, Chile, December 2015.

[14] C. Ma, J. B. Huang, X. K. Yang, and M. H. Yang, "Hierarchical convolutional features for visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '15)*, pp. 3074–3082, Santiago, Chile, December 2015.

[15] H. Li, Y. Li, and F. Porikli, "DeepTrack: learning discriminative feature representations online for robust visual tracking," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1834–1848, 2016.

[16] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)*, pp. 142–149, Hilton Head Island, SC, USA, June 2000.

[17] X. Ben, P. Zhang, W. Meng et al., "On the distance metric learning between cross-domain gaits," *Neurocomputing*, vol. 208, pp. 153–164, 2016.

[18] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 FPS with deep regression networks," in *European Conference on Computer Vision*, vol. 9905, pp. 749–765, Springer International Publishin, 2016.

[19] H. Zaimaga and M. Lambert, "Soft shrinkage thresholding algorithm for nonlinear microwave imaging," *Journal of Physics: Conference Series*, vol. 756, no. 1, Article ID 012011, 2016.

[20] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2042–2049, Providence, RI, USA, June 2012.

[21] J. Kwon and K. M. Lee, "Tracking by sampling trackers," in *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV '11)*, pp. 1195–1202, November 2011.

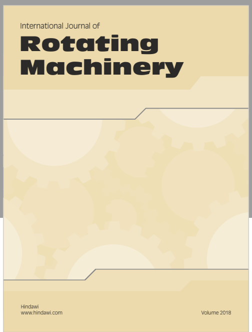
[22] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," *IEEE Conference Computer Vision Pattern Recognition*, pp. 661–675, 2002.

[23] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: a benchmark," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 2411–2418, Portland, Ore, USA, June 2013.

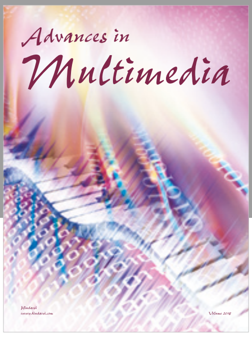
[24] W. Chen, K. Zhang, and Q. Liu, "Robust visual tracking via patch based kernel correlation filters with adaptive multiple feature ensemble," *Neurocomputing*, vol. 214, pp. 607–617, 2016.

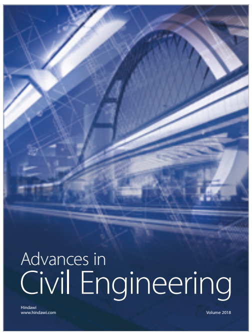
Journal of
Engineering

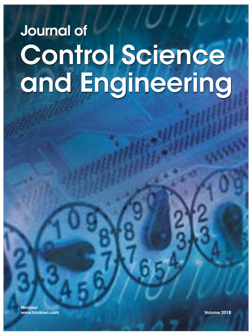
The Scientific
World Journal


International Journal of
Rotating
Machinery

Journal of
Sensors

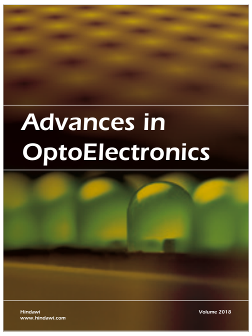

Advances in
Multimedia


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

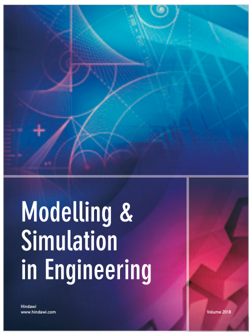
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

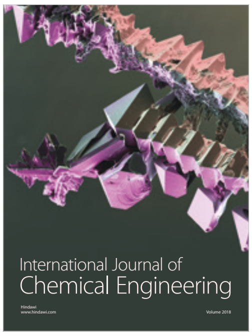

Advances in
OptoElectronics

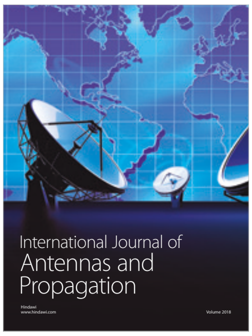
VLSI Design

International Journal of
Navigation and
Observation

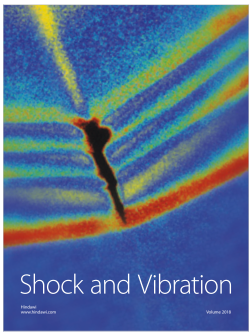

Modelling &
Simulation
in Engineering

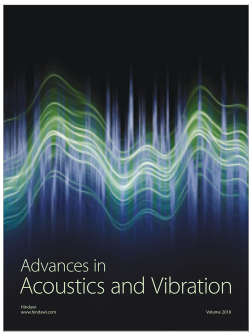
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration