A TARGET TRACKING ALGORITHM BASED ON ADAPTIVE MULTIPLE FEATURE FUSION

Hongpeng Yin, Yi Chai
College of Automation, Chongqing University, Chongqing City, 400030, China

Simon X. Yang
School of Engineering, University of Guelph, Guelph, Ontario, N1G 2W1, Canada

David K. Y. Chiu
Dept. of Computing and Information Science, Univ. of Guelph, Guelph, ON, N1G 2W1, Canada

Keywords: Target tracking, Feature fusion, Template update, Kernel-based tracking.

Abstract: This paper presents an online adaptive multiple feature fusion and template update mechanism for kernel-based target tracking. According to the discrimination between the object and background, measured by two-class variance ratio, the multiple features are combined by linear weighting to realize kernel-based tracking. An adaptive model-updating mechanism based on the likelihood of the features between successive frames is addressed to alleviate the mode drifts. In this paper, RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are employed to implement the scheme. Experiments on several video sequences show the effectiveness of the proposed method.

1 INTRODUCTION

Visual tracking is a common task in computer vision and play key roles in many scientific and engineering fields. Various applications ranging from video surveillance, human computer interaction, traffic monitoring to video analysis and understanding, all require the ability to track objects in a complex scene. Many powerful algorithms for target tracking have yielded two decades of vision research. Frame difference and adaptive background subtraction combined with simple data association techniques can effectively track in real-time for stationary cameras target tracking (Collins et al., 2001; Shalom and Fortmann, 1988; Stauffer and Grimson, 1995). Optical flow methods using the pattern of apparent motion of objects, surfaces and edges in a visual scene caused by the relative motion between the camera and scene. These methods can achieve the target tracking in the stationary cameras scene and the mobile cameras scene (Barron and Fleet, 1994; Tal and Bruckstein, 2008). Modern appearance-based methods using the likelihood between the tracked target appearance describe model and the reference target appearance describe model can achieve the target tracking without prior knowledge of scene structure or camera motion. Modern appearance-based methods include the use of flexible template models (Wang and Yaqi, 2008; Mattews et al., 2004) and kernel-based methods that track nonrigid objects used colour histograms (Comaniciu and Meer, 2002; Comaniciu et al., 2003; Li et al., 2008). Particle filter and Kalman filter are using to achieve more robust tracking of manoeuvring objects by introducing statistical models of object and camera motion (Comaniciu et al., 2003; Pan et al., 2008; Chang et al., 2008; Maggio et al., 2007).

The major difficulty in target tracking based on computer vision is the variation of the target appearance and its background. By using a stationary camera, the background in a long image sequence is dynamic. However, the performance of a tracker can be improved by using an adaptive tracking scheme and multiple features. The basic ideal is online adaptive selection of appropriate features for tracking. Recently, several adaptive tracking algorithms (Collins et al., 2005; Wang and Yaqi, 2008; Wei and Xiaolin,

Yin H., Chai Y., Yang S. and Chiu D. (2009).
A TARGET TRACKING ALGORITHM BASED ON ADAPTIVE MULTIPLE FEATURE FUSION.
In *Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics - Intelligent Control Systems and Optimization*, pages 5-12
DOI: 10.5220/0002154300050012

2007; Dawei and Qingming, 2007; Zhaozhen et al., 2008)were proposed. Collins et al. proposed to online select discriminative tracking features from linear combination of RGB values (Collins et al., 2005). The two-class variance ratio is used to rank each feature by how well it separates the sample distributions of object and background. Top N features that have the greatest discrimination are selected to embed in a mean-shift tracking system. This approach,however only considers the RGB colour features. Actually, this approach is one feature-based tracking. Furthermore, it lacks an effective model update method to copy with the model drifts. Liang et al. extend the work of Collins et al. by introducing adaptive feature selection and scale adaptation (Dawei and Qingming, 2007). A new feature selection method based on Bayes error rate is proposed. But how to deal with the model drifts is not addressed in this paper. He et al. used a clustering method to segment the object tracking according to different colours, and generate a Gaussian model for each segment respectively to extract the colour feature (Wei and Xiaolin, 2007). Then an appropriate model was selected by judging the discrimination of the features. The Gaussian model however, not always fit each segment in practise. Recently, Wang and Yagi extended the standard mean-shift tracking algorithm to an adaptive tracker by selecting reliable features from RGB, HSV, normalized RG colour cues and shape-texture cues, according to their descriptive ability (Wang and Yaqi, 2008). But only two best discriminate features are used to represent the target. It dose not use fully all the features information it has computed and has a high time complexity.

A key issue addressed in this work is an online, adaptive multiple-feature fusion and template-update mechanism for target tracking. Based on the theory of biologically visual recognition system, the main visual information comes from the colour feature, edge feature and the texture feature (Thomas and Gabriel, 2007; Jhuang et al., 2007; Bar and Kassam, 2006). In this paper, RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are employed to implement the scheme. Target tracking is considered as a local discrimination problem with two classes: foreground and background. Many works have point the features that best discriminate between object and background are also best for tracking performance (Collins et al., 2005; Thomas and Gabriel, 2007; Jhuang et al., 2007). In this paper the tracked target is represented by a fused feature. According to the discriminate between object and background measured by two-class variance ratio, the multiple features are combined by linear weighting to realize kernel-based tracking. As model drafts, better performance could be achieved by using a novel up-dating strategy that takes into account the similarity between the initial and current appearance of the target. Each feature's similarity is computed. The high similarity features are given a big weight and the low similarity features are given a small weight. A good feature for tracking is a steady feature across the consecutive frames. The target update model is updated by re-weighting the multiple features based on the similarity between the initial and current appearance of the target. The proposed approach is shown as Figure 1.

The paper is organized as follows. Section 2 presents a brief introduction the feature extraction. Section 3 presents our proposed approach for target tracking. Computer simulation and results compared with related work are presented in Section 4. Concluding remarks are given in Section 5.

2 FEATURE EXTRACTION

It is important to decide what kinds of features are used before constructing the feature fusion mechanism. In Collins et al. (2005), the set of candidate features is composed of linear combination of RGB pixel. In Wang and Yaqi (2008), colour cue and shape-texture cues are employed to describe the model of the target. In this paper based on the theory of biologically visual recognition system (Thomas and Gabriel, 2007; Jhuang et al., 2007; Bar and Kassam, 2006), RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are employed to implement the scheme.

2.1 RGB Colour Feature

Colour information is an important visual feature. That is robust to the target rotary, non-rigid transformation and target shelter, widely used in the appearance model-based visual application. In this paper, colour distributions are represented by colour histograms, and RGB colour space as a very common colour space is used in this paper. The R, G and B channels are quantized into 256 bins, respectively. The colour histogram, calculated using Epanechnikov kernel, is applied (Comaniciu and Meer, 2002)(Comaniciu et al., 2003).

2.2 Prewitt Edge Feature

The edge information is the most fundamental characteristic of images. It is also included useful information for target tracking. There are many methods

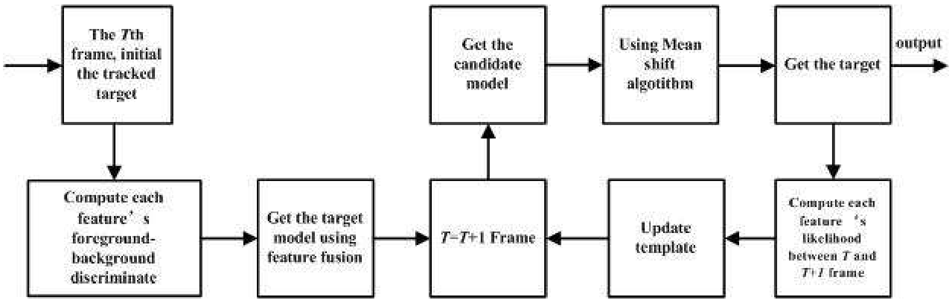

Figure 1: The Flow chart of the purposed approach.

for edge detection, but most of them can be grouped into two categories: search-based and zero-crossing based. The search-based methods detect edges by first computing a measure of edge strength, usually a first-order derivative expression, such as the gradient magnitude; and then searching for local directional maxima of the gradient magnitude using a computed estimate of the local orientation of the edge, usually the gradient direction (Yong and Croitoru, 2006; Sharifi et al., 2002). The zero-crossing based methods search for zero crossings in a second-order derivative expression computed from the image, in order to find edges, usually the zero-crossings of the Laplacian or the zero-crossings of a non-linear differential expression (Sharifi et al., 2002).

In this paper, Prewitt operator is employed to get the edge feature. For its low computational complexity and high performance. Prewitt operator has two convolution kernels as shown in Figure 2. Images of each point are used for the convolution kernel, the first kernel usually corresponding to the largest vertical edge, and the second corresponding to the largest horizontal edge. The maximum values of each point convoluted with the two kernels are accepted. Convolution is admitted as output value; results of operations are edge images. The Prewitt operator can be defined as

$$S_p = \sqrt{d_x^2 + d_y^2}, \tag{1}$$

$$\begin{aligned} d_x =& [f(x-1,y-1) + f(x,y-1) + f(x+1,y-1)] - \\ & [f(x-1,y+1) + f(x,y+1) + f(x+1,y+1)], \end{aligned} \tag{2}$$

$$\begin{aligned} d_y =& [f(x+1,y-1) + f(x+1,y) + f(x+1,y+1)] - \\ & [f(x-1,y-1) + f(x-1,y) + f(x-1,y+1)]. \end{aligned} \tag{3}$$

The histogram is used to represent the edge feature. Prewitt edge is also quantized into 256 bins Epanechnikov kernel like colour feature.

1	1	1
0	0	0
-1	-1	-1

1	0	-1
1	0	-1
1	0	-1

Figure 2: The two convolution kernel of Prewitt.

2.3 LBP Texture Feature

Local Binary Patterns (LBP) is basically a fine-scale descriptor that captures small texture details. It is also very resistant to lighting changes. LBP is a good choice for coding fine details of facial appearance and texture (T and T, 2007)(Aroussi and Mohamed, 2008)(Ojala et al., 1996). The Local Binary Patterns operator is introduced as a means of summarizing local gray-level structure by Ojiala in 1996 (Ojala et al., 1996). The operator takes a local neighbourhood around each pixel, thresholds the pixels of the neighbourhood at the value of the central pixel, and uses the resulting binary-valued image patch as a local image descriptor. It was originally defined for 3*3 neighbourhoods, given 8 bit codes based on the 8 pixels around the central one. Formally, the LBP operator takes the form

$$LBP(x_k, y_k) = \sum_{n=0}^{7} 2^n S(i_n - i_k), \tag{4}$$

$$S(u) = \begin{cases} 1, & u \geq 0, \\ 0, & u < 0, \end{cases} \tag{5}$$

where in this case n runs over the 8 neighbours of the central pixel k, i_k and i_n are the gray-level values at k and n, and $S(u)$ is 1 if $u \geq 0$ and 0 otherwise. The LBP encoding process is illustrated in Figure 3.

In methods that turn LBPs into histograms, the number of bins can be reduced significantly by assigning all non-uniform patterns to a single bin, often without losing too much information. In this paper, it is quantized into 256 bins with Epanechnikov kernel.

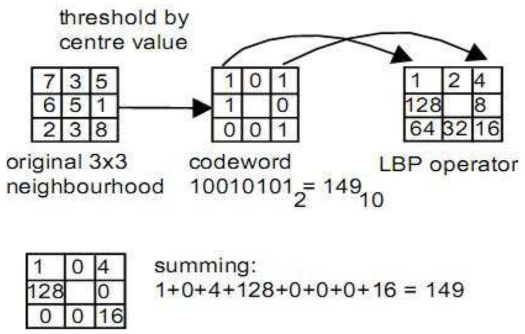

Figure 3: Illustration of the basic LBP operator.

3 ONLINE, ADAPTIVE FEATURES FUSION METHOD FOR TRACKING

There are two main components in this approach: the online adaptive features fusion based on discrimination criterion function, and the kernel-based tracking, which is used to track targets, based on the fused feature.

3.1 Features Fusion Method

In this paper, the target is represented by a rectangular set of pixels covering the target, while the background is represented by a larger surrounding ring of pixels. Given a feature f, let $H_{fg}(i)$ be a histogram of target and $H_{bg}(i)$ be a histogram for the background. The empirical discrete probability distribution $p(i)$ for the object and $q(i)$ for the background, can be calculated as $p(i) = H_{fg}(i)/n_{fg}$ and$q(i) = H_{bg}(i)/n_{bg}$, where n_{fg} is the pixel number of the target region and n_{bg} the pixel number of the background. The weight histograms represent the features only. It does not reflect the descriptive ability of the features directly. A log-likelihood ratio image is employed to solve this problem [14, 15]. The likelihood ratio nonlinear log likelihood ratio maps feature values associated with the target to positive values and those associated with the background to negative values. The likelihood ratio of a feature is given by

$$L(i) = \max(-1, \min(1, log(\frac{\max(p(i), \varepsilon)}{\max(q(i), \varepsilon)}))), \quad (6)$$

where ε is a very small number(set in 0.001 in this work), that prevents dividing by zero or taking the log of zero. Likelihood ratio images are the foundation for evaluating the discriminative ability of the features in the candidate features set. Figure 4 shows the likelihood ratio images of different features.

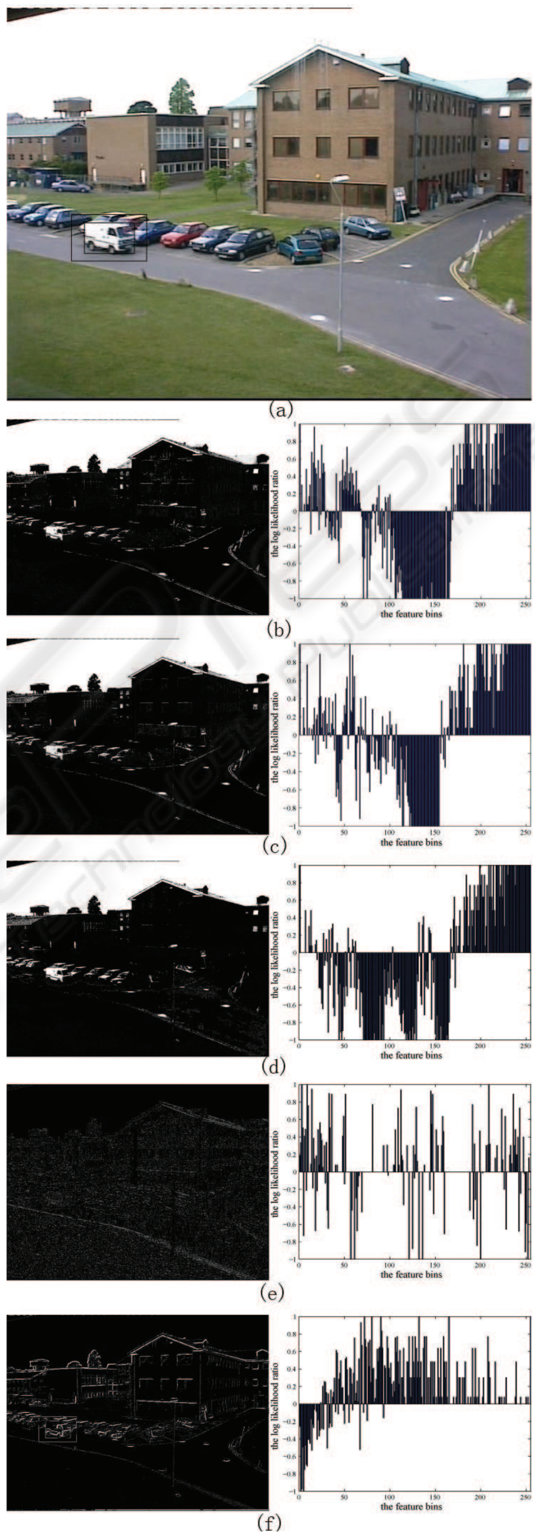

Figure 4: Different feature images after likelihood ratio process. (a) The input frame; (b) feature R and likelihood ratio; (c) feature G and likelihood ratio; (d) feature B and likelihood ratio; (e) LBP texture feature and likelihood ratio; (f) Prewitt edge feature and likelihood ratio.

In the practise, the whole weighted images weighted by log likelihood are not needed to be calculated for the computational complexity. The corresponding variance is employed to measure the separately between target and background classes. Using the method in Collins et al. (2005) and Wang and Yaqi (2008). based on the equality $var(x) = E[x^2] - (E[x])^2$, the variance of the log likelihood is computed as

$$\mathrm{var}(L:p) = E[L(i)^2] - (E[L(i)])^2. \quad (7)$$

The discriminative ability of each feature is calculated by the variance ratio. The hypothesis in this paper is that the features that best discriminate between target and background are also best for tracking the target. So, as the target's features describe model, a highest weight is given to the best discriminate feature, and the less discriminate feature has a smaller contribution. Based on the discrimination criterion function , the target features describe model can be calculated as

$$p_t(i) = \sum_{k=1}^{n} \lambda_k p_k(i), \quad (8)$$

$$\lambda_k = \frac{var_k(L:p)}{\sum_{k=1}^{n} var_k(L:p)}, \quad (9)$$

where $p_k(i)$ is the feature $K's$ probability distribution model, $\lambda_k$is the weight and $\sum_{k=1}^{n} \lambda_k = 1$. Figure 5 shows the fusion of the five features image.

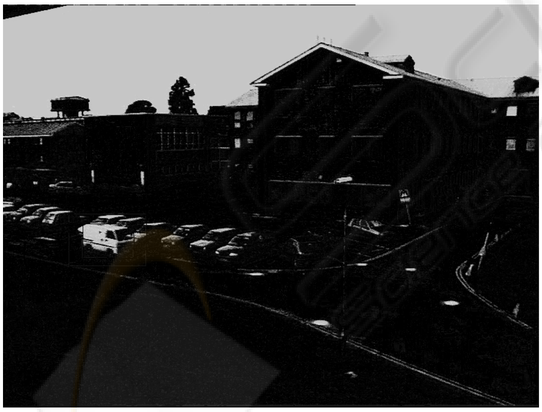

Figure 5: The image after fusion.

3.2 Kernel-based Tracking

Mean shift is a nonparametric kernel density estimator, which, based on the colour kernel density estimation, has recently gained more attention due to its low computational complexity and robustness to appearance change, however, the basic mean shift tracking algorithm assumes that the target representation is discriminative enough against the background. This assumption is not always true, especially when tracking is carried out in a dynamic background (Comaniciu et al., 2003)(Li et al., 2008). An online, adaptive features fusion mechanism is embedded in the kernel-based mean shift algorithm for effective tracking. Due to the continuous nature of video, the distribution of target and background features in the current frame should remain similar to the previous frame and the fused feature model should still be valid. The initial position of the target is given by $y0$ which is determined in the previous frame. The target model is $P = p_{t,t=1\ldots m}, \sum_{t=1}^{m} p_t = 1$, and the candidate target model is $P(y_0) = p_t(y_0)_{t=1\ldots m}, \sum_{t=1}^{m} p_t = 1$, where p_t is the fused feature model. The Epanechnikov profile [8, 9] is employed in this paper. The target's shift vector form y_0 in the current frame is computed as

$$y_1 = \frac{\sum_{i=1}^{n_h} X_i \omega_i g(\|\frac{y_0 - X_i}{h}\|^2)}{\sum_{i=1}^{n_h} \omega_i g(\|\frac{y_0 - X_i}{h}\|^2)}, \quad (10)$$

where $g(x) = -k'(x)$,$k(x)$ is Epanechnikov profile, h is bandwidth and $\omega_i$can compute as

$$\omega_i = \sum_{t=1}^{m} \sqrt{\frac{P}{P(y_0)}} \delta[b(x_i) - t]. \quad (11)$$

The tracker assigns a new position of the target by using

$$y_1 = \frac{1}{2}(y_0 + y_1). \quad (12)$$

If $\|y_0 - y_1\|$, the iteration computation stops and y_1 is taken as the position of the target in the current fame. Otherwise let $y_0 = y_1$, then using Eq. (10) get the shift vector and do position assignment using Eq. (12). From Eq. (8) and Eq. (10), the pixels' weight is assigned by two parts. One is the kernel profile, which gives high weight to the pixel nearly to the centre. The other one is the discriminative ability of each feature. Higher weight is given to the higher discriminative ability feature.

3.3 Template Update Mechanism

It is necessary to update the target model, because the variation of the target appearance and its background. When the target appearance or the background changes, the fixed target model can not accurate describe the target, so it can not obtain the right position of target. But using an inaccurately tracking result to update the target model may lead to the wrong update of the target model. With the error accumulate, it finally results in track failure.

In order to alleviate the mode drifts, an adaptive model update mechanism based on the likelihood of the features between successive frames is proposed in this paper. During the initialization stage, the target is obtained by a hand-draw rectangle and the target model is computed by the fusion method introduced in the previous subsection. The fused target model is used for tracking in the next frame and is also kept to use in subsequent model updates. Following the method in (Wang and Yaqi, 2008), the updated target model M can be computed as

$$M = (1-L_{ic})M_i + L_{ic}M_c, \quad (13)$$

$$L_{ic} = \sum_{u=1}^{m} \sqrt{M_i M_c}, \quad (14)$$

where L_{ic} is the likelihood between the initial model and current model measured by Bhattacharyya coefficient (Comaniciu and Meer, 2002)(Comaniciu et al., 2003); M_i is the initial model; M_c is the current target model computed as

$$M_c = (1-L_{pc})M_p + L_{pc}M_a, \quad (15)$$

where M_p is the previous target model, M_a is the current target fused-feature introduced in section 3, and L_{pc} is computed the likelihood between the M_a and M_p. The proposed updating method considers temporal coherence by weighing the initial target model, previous target model and current candidate. It can be more robust for the target appearance and the background change.

4 EXPERIMENTS

To illustrate the benefits of the proposed approach, experiments on various test video sequences using the proposed approach and other algorithms are conducted. The experiment was done using Pentium core 1.8G, Win XP, MATLAB 7.0. The Epanechnikov profile was used for histogram computations. The RGB colour feature, LBP texture feature and Prewitt edge feature were taken as feature space and it was quantized into 256 bins. The public dataset with ground truth is used to test the proposed method (Collins et al., 2005). The tracking results are compared with the basic mean-shift (Comaniciu and Meer, 2002), fore/background ratio, variance ratio, peak difference (Comaniciu et al., 2003), and multiple feature (Wang and Yaqi, 2008) trackers. The initialization of the tracking is given by a hand-draw rectangle. The same initializations are given to all the trackers. In this experiment, sequences EgTest01, EgTest02, EgTest03, EgTest04 and EgTest05 in the database are chosen. This challenging tracking sequences is made by various factors, such as different viewpoints, illumination changes, reflectance variations of the targets, similar objects nearby, and partial occlusions.

The tracking success rate is the most important criterion for target tracking, which is the number of successful tracked frames divided by the total number of frames. The bounding box that overlaps the ground truth can be considered as a successful track. To demonstrate the accuracy of tracking, the average overlap between bounding boxes (Avg overlap BB) and average overlap between bitmaps within the overlapping bounding box area (Avg overlap BM) are employed. Avg overlap BB is the percentage of the overlap between the tracking bounding box and the bounding box identified by ground truth files. Avg overlap BM is computed in the area of the intersection between the user bounding box and the ground truth bounding box. The comparison results are shown in Table 1.

From the comparison results that show the successful tracking ratio, the proposed tracker gives the best results in five of the test sequences. The basic mean-shift tracker dose not have a good performance in EgTest01, EgTest02 and EgTest05, because the basic mean-shift tracker dose not use the multi-features information and lacks adaptive strategy. Although an adaptive strategy is employed in Collins's approach, it dose not have good performance in EgTest01, EgTest02 and EgTest03. The peak difference algorithm has a better performance in the first sequences. The others sequences, however, do not demonstrate a good performance either. The multi-features methods that Integrate Colour and Shape-Texture features have a higher performance in all the sequences. Although the proposed approach has the best performance than the other trackers. But in EgTest03, EgTest04 and EgTest05 the successful tracking ratio is only 25.30%, 12.03% and 24.31%. The failed tracking examples are show as Figure 6.The main reason leading to tracking failure includes the similar feature distribution nearby as (a) in Fig. 6, the lower discrimination between foreground and background as (b) in Fig. 6 and long time occlusions as (c) in Fig. 6.

For accuracy of tracking, the proposed tracking algorithm is not the best in some of the sequences. There is not obvious correlation between the tracking accuracy criterion and the tracking successful ratio.The proposed approach does not have the highest accuracy, because in most frame-sequences,the background and the target are not always separated accurately.

Table 1: Tracking performance of different algorithms. (a) EgTest01; (b) EgTest02; (c) EgTest03; (d) EgTest04; (e) EgTest05.

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	17.58%	100%	29.12%	100%	100%	100%
Avg overlap BB	65.50%	62.87%	76.87%	61.76%	61.62%	74.28%
Avg overlap BM	66.26%	49.15%	61.30%	57.76%	68.38%	76.56%

(a)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	39.23%	39.23%	27.69%	30.77%	100%	100%
Avg overlap BB	91.09%	89.13%	85.19%	90.54%	93.32%	94.21%
Avg overlap BM	74.69%	66.98%	73.32%	65.91%	72.70%	73.53%

(b)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	20.62%	17.90%	12.06%	12.06%	20.23%	25.30%
Avg overlap BB	86.96%	87.01%	93.74%	92.27%	88.66%	90.11%
Avg overlap BM	66.65%	54.04%	70.79%	67.20%	69.37%	71.23%

(c)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	9.84%	8.74%	9.84%	3.83%	9.84%	12.03%
Avg overlap BB	66.78%	67.92%	66.03%	63.60%	69.52%	71.00%
Avg overlap BM	59.70%	52.75%	66.74%	66.42%	56.34%	68.43%

(d)

Criterion	MeanShift	FgBgRatio	VarianceRatio	PeakDiff	Multi-feature	The Proposed
Successful ratio	13.64%	13.64%	13.64%	13.64%	21.22%	24.31%
Avg overlap BB	94.58%	88.75%	86.46%	86.98%	72.65%	75.41%
Avg overlap BM	84.02%	71.12%	85.12%	69.90%	64.45%	70.03%

(e)

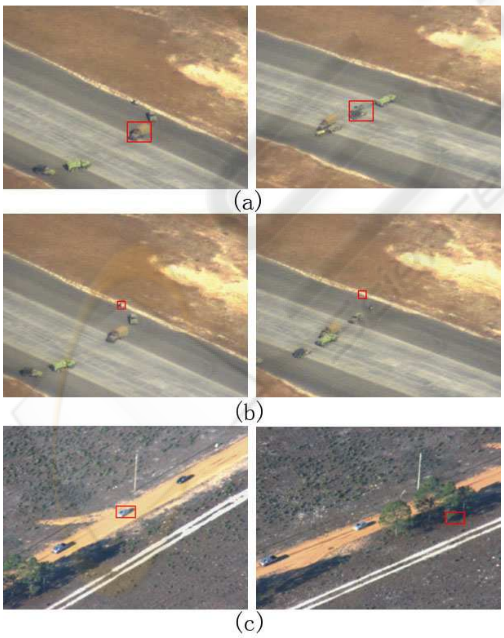

Figure 6: The failure tracking examples of the proposed approach. (a) and (b) the failure frame in EgTest03; (c) the failure result in EgTest04.

5 CONCLUSIONS

An online, adaptive multiple-feature fusion and template update mechanism for kernel-based target tracking is presented in this paper. Based on the theory of biological visual recognition system, RGB colour features, Prewitt edge feature and local binary pattern (LBP) texture feature are utilized to implement the proposed scheme. Experiment results show that the proposed approach is effective in target tracking. The comparison studies with other algorithms show that the proposed approach performs better in tracking of moving targets.

ACKNOWLEDGEMENTS

This work is partially supported by China Scholarship Council.

REFERENCES

Aroussi, E. and Mohamed (2008). Combining dct and lbp feature sets for efficient face recognition. In *3rd Inter-*

national Conference on Information and Communication Technologies: From Theory to Applications, 3rd International Conference on Information and Communication Technologies: From Theory to Applications, pages 1–6.

Bar, M. and Kassam, K. (2006). Top-down facilitation of visual recognition. In *Proceedings of the National Academy of Sciences*, volume 103 of *Proceedings of the National Academy of Sciences*, pages 449–454.

Barron, J. and Fleet, D. (1994). Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77.

Chang, W., Chen, C., and Jian, Y. (2008). Visual tracking in high-dimensional state space by appearance- guided particle filtering. *IEEE Transactions on Image Processing*, 17(7):1154–1167.

Collins, R., Lipton, A., Fujiyoshi, H., and Kanade, T. (2001). Algorithms for cooperative multi-sensor surveillance. In *Proceedings of the IEEE*, volume 89 of *Proceedings of the IEEE*, pages 1456–1477.

Collins, R., Liu, Y., and Leordeanu, M. (2005). Online selection of discriminative tracking features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1631–1643.

Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619.

Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577.

Dawei, L. and Qingming, H. (2007). Mean-shift blob tracking with adaptive feature selection and scale adaptation. In *IEEE Conference on Image Processing*, volume 3 of *IEEE Conference on Image Processing*, pages 369–372.

Jhuang, H., Serre, T., Wolf, L., and Poggio, T. (2007). A biologically inspired system for action recognition. In *IEEE 11th international conference on Computer Vision*, volume 14 of *IEEE 11th international conference on Computer Vision*, pages 1–8.

Li, Z., Tang, L., and Sanq, N. (2008). Improved mean shift algorithm for occlusion pedetrian tracking. *Electronics Letters*, 44:622–623.

Maggio, E., Smerladi, F., and Cavallaro, A. (2007). Adaptive multifeature tracking in a particle filtering framework. *IEEE transactions on Circuits and Systems for Video Technology*, 12(10):1348–1359.

Mattews, I., Iashikawa, T., and Baker, S. (2004). The template update problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):810–815.

Ojala, T., Pietikainen, M., and Harwood, D. (1996). A comparative study of texture measures with classifcation based on feature distributions. *Pattern Recognition*, 29(1):51–59.

Pan, J., Ho, B., and Zhang, J. Q. (2008). Robust and accurate object tracking under various types of occlusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(2):223–236.

Shalom, Y. and Fortmann, T. (1988). *Tracking and Data Association*. Academic Press, Boston.

Sharifi, M., Fathy, M., and Mahmoudi, M. (2002). A classified and comparative study of edge detection algorithms. In *International Conference on Information Technology: Coding and Computing*, International Conference on Information Technology: Coding and Computing, pages 117–120.

Stauffer, C. and Grimson, W. (1995). Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):814–820.

T, X. and T, B. (2007). *Fusing Gabor and LBP Feature Set for Kernel-Based Face Recognition*, volume 4778 of *Lecture Notes in Computer Science*. Springer Berlin, Heidelberg.

Tal, N. and Bruckstein, A. (2008). Over-parameterized variational optical flow. *International Journal of Computer Vision*, 76(2):205–216.

Thomas, S. and Gabriel, K. (2007). A quantitative theory of immediate visiual recognition. *Program on Brain Research*, 167:33–56.

Wang, J. and Yaqi, Y. (2008). Integrating color and shape-texture features for adaptive real-time object tracking. *IEEE Transactions on Image Processing*, 17(2):235–240.

Wei, H. and Xiaolin, Z. (2007). Online feature extraction and selection for object tracking. In *IEEE Conference on Mechatronics and Automation*, IEEE Conference on Mechatronics and Automation, pages 3497–3502.

Yong, Y. and Croitoru, M. (2006). Nonlinear multiscale wavelet diffusion for speckle suppression and edge enhancement in ultrasound images. *IEEE transactions on Medical Imaging*, 25(3):297–311.

Zhaozhen, Y., Poriki, F., and Collins, R. (2008). Likelihood map fusion for visual object tracking. In *IEEE workshop on Applications of Computer Vision*, IEEE workshop on Applications of Computer Vision, pages 1–7.