A novel object tracking method based on a mixture model

Dongxu Gao · Zhaojie Ju · Jiangtao Cao · Honghai Liu

Received: date / Accepted: date

Abstract Object tracking has been applied in many fields such as intelligent surveillance and computer vision. Although much progress has been made, there are still many puzzles which pose a huge challenge to object tracking. Currently, the problems are mainly caused by occlusion, similar object appearance and background clutters. A novel method based on a mixture model was proposed for solving these issues. The mixture model was integrated into a Bayes framework with the combination of locally dense contexts feature and the fundamental image information (i.e. the relationship between the object and its surrounding regions). This is because that the tracking problem can be seen as a prediction question, which can be solved using the Bayes method. In addition, both scale variations and templet updating are considered to assure the effectiveness of the proposed algorithm. Furthermore, the Fourier Transform (FT) is used when solving the Bayes equation to make the algorithm run in a real-time system. Therefore, the MMOT (Mixture model for object tracking) can run faster and perform better than existing algorithms on some challenging images sequences in terms of accuracy, quickness and robustness.

Keywords Object Tracking · MMOT · Fourier Transform · Bayes Equation

1 Introduction

Object tracking is known as locating positions of interest area over time in every frame of a video. Many targets with various features have been researched for different applications. For example, Xiang [1] utilized optical flow and sampled points within the Markov Decision Process framework for tracking pedes-

Dongxu Gao, Zhaojie Ju and Honghai Liu
School of Computing, University of Portsmouth, UK
E-mail: zhaojie.ju@port.ac.uk

Jiangtao Cao
Faculty of Information and Control Engineering, Liaoning Shihua University, China

trians. Single person and multiple people could be tracked effectively under the proposed tracking framework. Li [2] adopted the discriminative feature into a convolutional neural network (CNN) framework for tracking an arbitrary single object after learning its feature online. In addition, the trajectory and the state of the tracking target could also contribute the tracking in many application areas.

Besides, object tracking can be applied in many application areas such as human-computer interaction, self-driving vehicles and surveillance system. Many researchers contribute their time for getting a more robust and effective tracking result with a focus on object appearance modelling, model updating, optimizing algorithm and recent hot topic deep learning. Although many different kinds of object tracking algorithms have been studied for several decades, and much progress has been made in recent years [3], there are still many challenging problems such as fast movement, illumination variation, occlusion, background clutters and proceeding time.

Generally speaking, current tracking algorithms are categorized as two methods: generative tracker and discriminative tracker. The noticeable difference between them is how to build an appearance model of the tracking target [4]. A generative tracker usually focuses on the appearance of a moving object and tries to find a model to represent it. It is unnecessary to consider the background information, which makes the tracker works faster. Online updating method is often used in case that the appearance changed. However, the change of the object appearance caused by some factors such as occlusion and pose variation makes it more difficult for modelling. Some generative tracking examples can be found from a benchmark paper [5].

A discriminative method mainly emphasizes how to separate the target from the background in a video scene. Finding a decision boundary between the object and the background is the key issue for a discriminative method. It is well known that the discriminative method works better when enough training samples were given. This tracker works well even though dramatic changes of an object, but it needs more sophisticated calculation, which makes it fail to use in a real-time system as the higher process speed is needed. Due to a large number of features is necessary, the offline feature selection procedure and trained classifier make it difficult to get an arbitrary object type for tracking approaches which need online boosting.

Some methods try to combine the generative and discriminative models, which can be often treated as a semi-supervised problem. A common approach learns an online appearance model which can select features from an arbitrary object. The core idea of a combination method is to predict a classifier with the aim of enlarging the training data after obtaining two independent conditional classifiers from the same data. More detail information about the comparison of discriminative and generative models can be found in [6].

Therefore, various representative models become an important research topic. The algorithm framework will also be researched for computing different models. For example, the methods [7,8] used the Bayes theorem as a basic framework in this paper. However, a novel appearance model and the solution method for a model are quite different. Motivated by literatures about Bayes framework, the main contribution of this paper is: 1. the appearance model of an object is modelled as a prediction problem; 2. the MMOT with the combination of locally dense context feature and fundamental image information is proposed; 3. the fast Fourier Transform is introduced for solving the Bayes equation and reducing the processing time.

In this paper, advantages of both colour information and Fourier transform are utilized for effectiveness and efficiency. The rest of this paper is organized as follows: the second section reviewed some related work, the MMOT is introduced in section 3; experimental results and discussion are given in section 4, finally, conclusion is given in section5.

2 Related work

A typical tracking algorithm consists of four steps: object representation, search mechanism, model solving and model updating. For recent generative trackers and discriminative trackers, both of their key step is how to acquire a better appearance of an object. There are many papers focus on object information to find the target. Recently, there are several methods utilized context information to handle object tracking which locate the target through finding consistent information of an object. To do so, related data mining method should be introduced for extracting both object and its surrounding region as supplement information, although satisfied results have been obtained, computational cost are still needed. Not only that, templates and subspace models are also contribute to robust performance. Wang et al [9] utilized the subspace model, which can handle appearance change while online learning model can learn appearance model in IVT methods. To solve this kind of model, the optimized algorithms [10] have been proposed to meet the real time performance such as proximal gradient approach and the l1-norm related minimization method [11]. These methods seem sensitive to partial occlusion according to many experiments.

Although some algorithms [12] were proposed to manage occlusion while drift might be occur because of the offline update of template or offline subspace model. Many researchers have developed the online updating model which can deal with drift well. However, the scale of an object sometimes change which poses another challenge for these trackers. For different trackers [13], scale updating should be considered separately. Compressive tracking [14] method cannot handle scale variance well but introduce a multi-scale information in fast compressive tracking (FCT) [15]. However, there is no colour information included for FCT which might fail when the colour of the object and background are similar. Fei etc [16] introduced a perceptual hashing method which can track moving object effectively. Wang etc. [17] used a probability continuous outlier model and background information for the tracking issue. They also proved that the least soft-threshold squares can improve the tracking performance [18].

Lots of researchers who exploited colour information have achieved an excellent performance in object detection. This method not only handle similar colour problem but can also locate the first location of an object [19]. Danelljan et al. [20] analysed how the colour information contribute the performance of tracking and the experiments proved the effectiveness compared with CSK tracker and VTD tracker [18, 21].

3 Proposed algorithm

Recently, object tracking problem has been treated as a predictive problem which can be solved by the particle filter framework based on the Bayes theorem. The main difference compared with previous traditional particle filter framework is that the number of particles is not needed for solving the model while using a kernel function to obtain the probability needed. When estimating the object location, the object location likelihood is used which is shown as follows:

$$p(x) = p(x|o) \tag{1}$$

x is the output vector which includes the predicting object information and represents the current object feature in an image sequence. p(x) can be computed according to the Bayes theory.

$$\begin{aligned} p(x) = p(x|o) &= \sum_{f(\mathbf{z}) \in X^f} p(x, f(\mathbf{z}, o)) \\ &= \sum_{f(\mathbf{z}) \in X^f} p(x|f(\mathbf{z})|o) p(f(\mathbf{z})|o) \end{aligned} \tag{2}$$

Then, the problem can be transferred to compute the joint probability. $p(x)$ represents the context feature, $f(z)$ denotes image information including the location and the feature of a target, it can be represented as eq.(3).

$$M(z) = (V(\mathbf{z}), \mathbf{z}) \tag{3}$$

denotes the colour information which adopted the HSV (Hue, Saturation, Value) colour space at location $z(m, n)$,especially the value of V channel (the use of V channel makes the algorithm work well for both colour images and gray-scale images), z belongs to the neighbourhood of location X that includes target object. The target model is defined as z which includes the vectorized image patches centred at pixel position c, the distance between the surrounding pixel and the centre is assigned by applying an isotropic kernel $k(c)$, and then the target model is obtained by computing the value of the colour model histogram, in which the $j - th$ value is:

$$q_j = N_c \sum_{i=1}^{N} k(||c||^2)|\alpha_f| \tag{4}$$

where N_c is the normalisation constant to make sure the summation is 1, and α_f is the coefficient of the image patch. α_f is the learning rate, C_f is the covariance matrix of the current frame appearance, and Λ_j is the a $D_1 \times D_2$ diagonal matrix. Then we select a mapping matrix B_1 according to normalised eigenvectors of R_f, which denotes the largest eigenvalue. The mapping matrix is found by the dimensionality reduction technique to get a projection $D_1 \times D_2$ with orthogonal column vectors.

As the colour attributes normally have high-dimensional colour features, a dimensionality reduction method is used to make the algorithm preserve useful information after the colour dimensions are reduced dramatically, then the computational time will be decreased. The problem of dimension reduction is formulated to find a mapping for the current frame f, by performing an eigenvalue decomposition of the matrix in eq. (4).

The framework of our proposed algorithm is described in the following table.

Algorithm 1 The framework of the MMOT method

1. Compute the target appearance with q_j
2. Integrate the appearance into a Bayes framework
3. Compute the condition probability with the FFT
4. Compute the appearance function
5. Compute the kernel function
6. Integrate last two steps in an inverse FFT
7. Output the current the object location from the last step
8. Update the learning parameters
9. Update the appearance model
10. Detect the next frame object location

To solve the eq.(2), two conditional probability should be computed separately.

$$p(x|f(\mathbf{z}), o) = h(x - \mathbf{z}) \tag{5}$$

Where h can be seen as a kernel function with respect to the relationship between the centre location of object and its surrounding region. The object location likelihood can be computed through the confidence map as [22]

$$C_m(x) = P(f(z)|o) = ae^{\frac{|z-x^*|^\beta}{\sigma}} \tag{6}$$

In eq.(6), a denotes the normalization constant, σ represents a scale parameter and β define the shape parameter. The confidence map in the eq.(7) considers the colour information of the tracking target which improves the challenging problem effectively. The STC method guides us about how to set the parameters of β with some experimental results.

Then, take (3)(4)(5)(6) into account, the eq.(2)can be formulated as:

$$\begin{aligned} p(x) &= \sum_{f(\mathbf{z})\in X^f} h(x-\mathbf{z})V(\mathbf{z})\omega_\sigma(\mathbf{z}-x^*) \\ &= h(x)\bigotimes V(x)\omega_\sigma(x-x^*) \end{aligned} \tag{7}$$

As the $\bigotimes$ is a convolution operator, so the Fast Fourier Transform (FFT) can be applied for ensuring the computing speed fast, the location of an object can be determined by the maximum value of $p(x)$ at the $(t+1)$th frame, which can be represented as:

$$F(be^{|\frac{(x-x^*)}{\alpha}|^\beta}) = F(h(x))\bigodot F(I(x)\omega_\sigma(x-x^*)) \tag{8}$$

Therefore, the appearance model can be obtained by:

$$h(x) = F^{-1}(\frac{F(be^{-|\frac{x-x^*}{\alpha}|^\beta})}{F(I(x)\omega_\sigma(x-x^*))}) \tag{9}$$

In addition, it is well known that the visual tracking could fail when the target appearance changes. So it is necessary to update the target model over time. For the MMOT tracker, the appearance model considers the learned target x and the transformed classifier coefficient A computed using the current appearance, and then we use a simple linear interpolation method to update the classifier coefficients:

$$A^t = (1-\rho)A^{t-1} + \rho A \tag{10}$$

where t indicates the the current frame and ρ means the learning rate parameter, thus a sub-optimal problem is introduced. A scheme, allowing the model to be updated without storing the previous target appearances, is introduced to ensure the fast computing speed. Then not all previous frames are considered when computing the current model.

$$A_C^t = (1-\rho)A_D^{t-1} + \rho O^t(O^t+\rho) \tag{11}$$

$$x_C^t = (1-\rho)x_C^{t-1} + \rho x_C^t \tag{12}$$

O^t is the output of the Fourier transformed kernel, the weight is set with a learning rate ρ, x^t denotes the learned target appearance to calculate the detection scores for the next frame appearance. Therefore, only A_C^t and x^t need to be stored with updating method in above equations.

4 Experiments

We have successfully integrated our method into a real-time system which is used for the task of not only tracking the Autism children but also need tracking some objects that children are grasping. To prove the efficiency of the algorithm, we evaluate our method on eight challenging image sequences and compare its performance with some other methods which represent the most common tracking framework. For the convenience of comparison, the algorithm is implemented in Matlab and achieves at least 25 frames per second on a PC with Intel E7500 CPU (2.93GHz).

Some key parameters setting in our experiments are: α is 3 and β is 1; the scale factor is set as 1 and the learning parameter is 0.05. The scale is updated every 5 frame. For the colour information, the tracker normalises the scale values to [-0.5,0.5], which can counter the distortion as an effect of the window operation, thus avoid to affect the kernel. The kernel is introduced as we extend the colour feature to multi-dimensional features, which are extracted from an image patch and it is set to 6 to get the best result.

In order to illustrate the qualitative comparison more clearly, some methods most used to be compared are introduced as they used different object representing methods to locate the very first object appeared in the first frame, and various computing methods are used to solve their models. These methods are described briefly here. The Visual Tracking Decomposition (VTD) [21] method used the observation model, which is decomposed into multiple basic observation models that are constructed by sparse principal component analysis (SPCA) [10] of a set of feature templates. The MIL [23] method put all ambiguous positive and negative samples into bags to learn a discriminative model for tracking. The $L1$ method [24] adopted the holistic representation of the object as the appearance model and then track the object by solving the $L1$ minimization problem. The assessment of several methods above in different situations are shown as below:

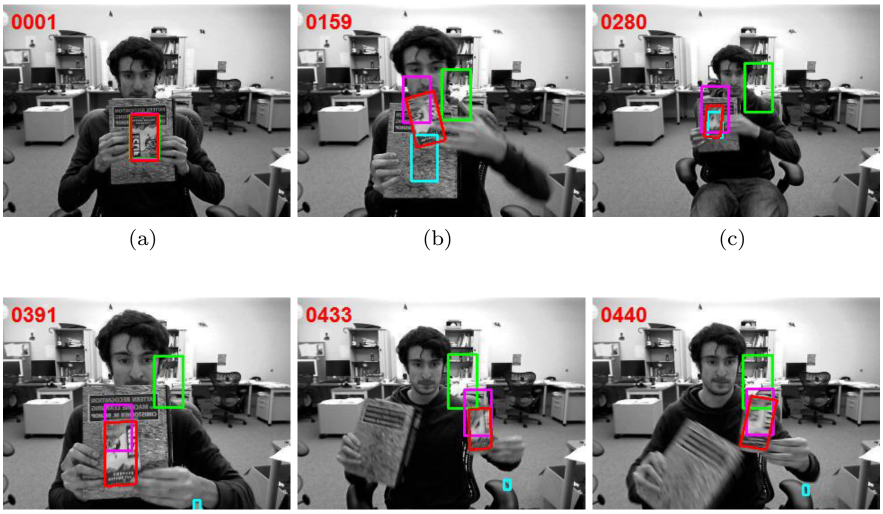
a. Qualitative and quantitative evaluation Figure 2-9 show the tracking results of the proposed method and three different algorithms including L1, VTD and MIL in eight diverse images sequences for tracking. We use red for our method, green for L1, blue for VTD and pink for the MIL. These images sequences are extremely challenging because they contain various difficulties for tracking such as occlusion, scale change, similar objects, illumination change, fast motion , camera angles and cluttered background. The tracking rectangle with different colours represents the compared methods which is shown in Figure 1.

In the sequence of Cliffbar, the L1 and VTD trackers drift away from the object and could not track the target again when the object is on the top of the book shown in Figure 2, the major challenge is the object and background share the similar appearance sometimes. The results show our method performs good even the background information is similar to the target.

There is a huge illumination change in the sequence of DavidIndoor which is supposed to be one of the main challenges to track. However, from the

Figure 3, it is clear to see that all these four trackers can handle the challenges but the MIL method seems more sensitive to the scale change. Our method is adaptive when the light is changing, the camera is moving and the appearance is changing because of the glasses and the face angle.

To the DavidOutdoor in Figure 4, the L1 tracker performs the worst after the person appears in the back of a tree and could not track it again. Even though the VTD and MIL tracker fails to track the person when the occlusion occurs but they can track it afterwards. Our method can track the person from the beginning to the end even though the occlusion is occur.



(a) (b) (c)

(d) (e) (f)

Fig. 1: The sequences of Cliffbar

The four trackers try to track a girl's face in Figure 5, only our method can track the face from the beginning to the end though the similar face appears and blocks the target. The other three cannot handle this problem and the scale change. But they perform well when only occlusion occurs as seen in Figure 6 and Figure 7. The only problem is that they could not locate the target particular accurate when the target changes the angles.

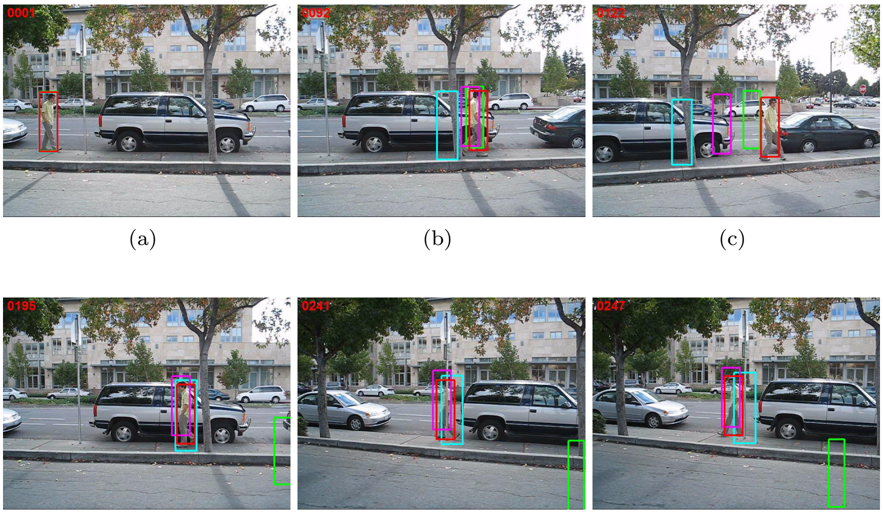
Fast motion is an extremely difficult problem in object tracking, both our method and VTD method achieve a satisfied performance all along as shown in Figure 8. All these three trackers except ours could not track an indicated object when the object is quite similar with the background in Figure 9. Our method has the ability to handle different tracking difficulties no matter they appear individually or in distinct combinations.

b. Discussion

(a) (b) (c)

(d) (e) (f)

Fig. 2: The sequences of DavidInDoor



(a) (b) (c)

(d) (e) (f)

Fig. 3: The sequences of DavidOutdoor

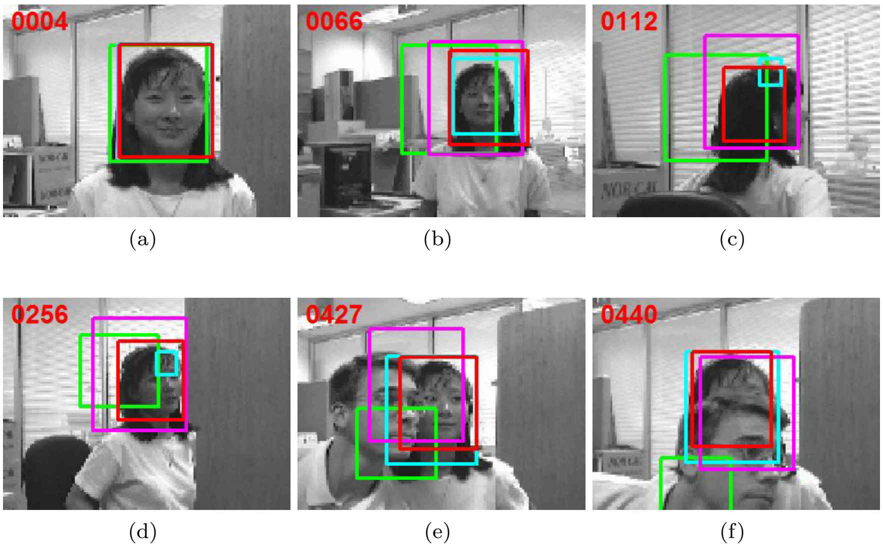


Fig. 4: The sequences of Girl

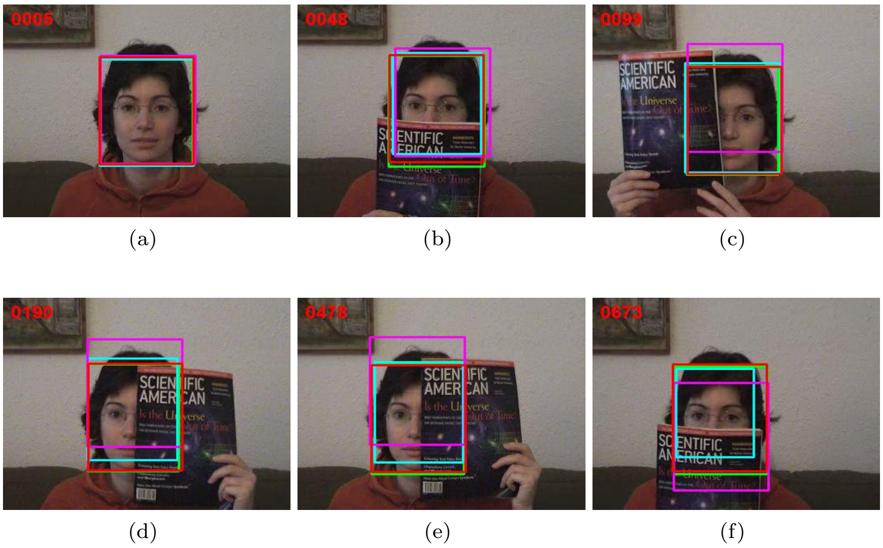
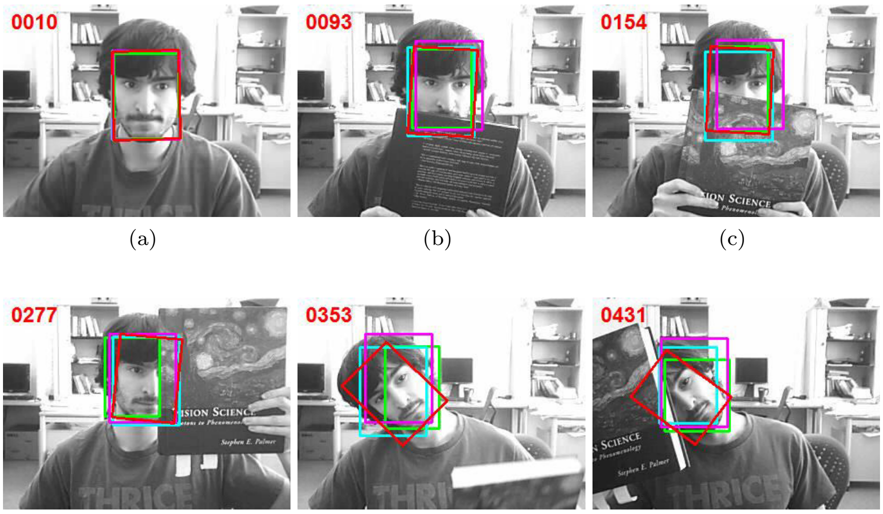


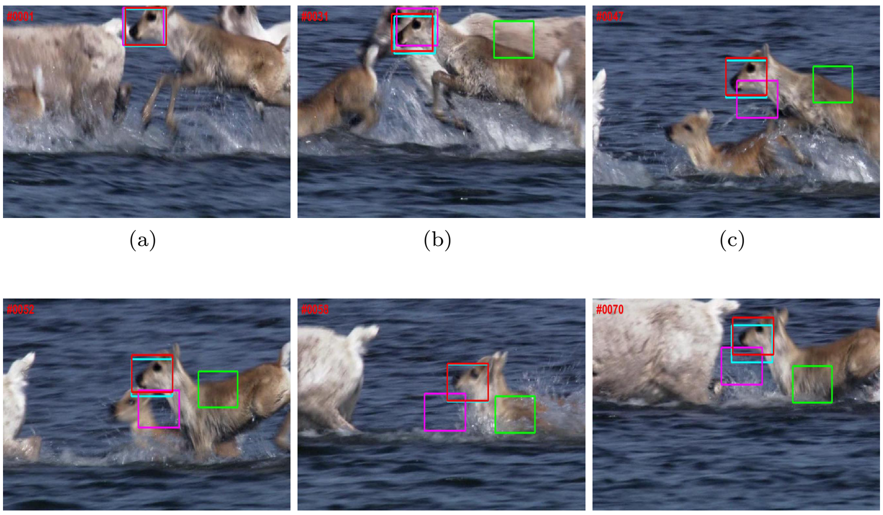
Fig. 5: The sequences of Occlusion1



(a) (b) (c)

(d) (e) (f)

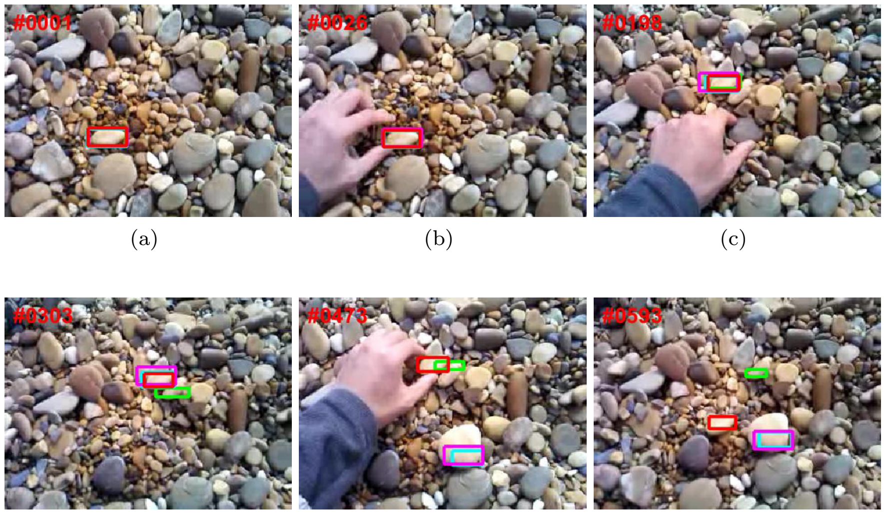
Fig. 6: The sequences of Occlusion2



(a) (b) (c)

(d) (e) (f)

Fig. 7: The sequences of Deer



(a) (b) (c)

(d) (e) (f)

Fig. 8: The sequences of Stone

Although these methods can track the object both for the sequences Davidindoor and Occlusion1 as some occlusion occurs, if there are some rotation for the Occlusion2 sequences occurs, only our method performs well.

L1 and VTD could not handle the severe occlusion like Girl sequences, while only the VTD method could not track the object again if there is a drift when tracking, other methods could keep tracking after temporally drift. For the sequences of Cliffbar, the colour of moving object is nearly same with its surrounding region, only our method can keep tracking over the time,as both color information and context information were adopted when modelling appearance. So the experimental results show our method is robust to the current tracking challenges including the occlusion and rotation, and performs best compares with other methods.

In addition,as both center error evaluation and overlap evaluation, which are defined by the PASCAL VOC, have been used to evaluate the performance of the proposed algorithms. We use the same evaluation criterion in this paper.

Table 1 and Table 2 summarizes the experimental results in terms of the average center error and the average tracking overlap. It is clearly to see that our method achieves the lowest tracking errors compared with the others in Table 1, and the highest overlap rate in Table 2. The overlap rate is one of the evaluations to verify the tracking success. According to the PASCAL VOC criterion, given the tracking result of each frame R_T and the corresponding ground truth R_G, the $score = \frac{area(R_T \cap R_G)}{area(R_T \cup R_G)}$, indicates the tracking performance. The tracking results are regarded as being valid when the score is over 0.5. The average overlap rate of our tracker is 0.75 while the highest is 0.50

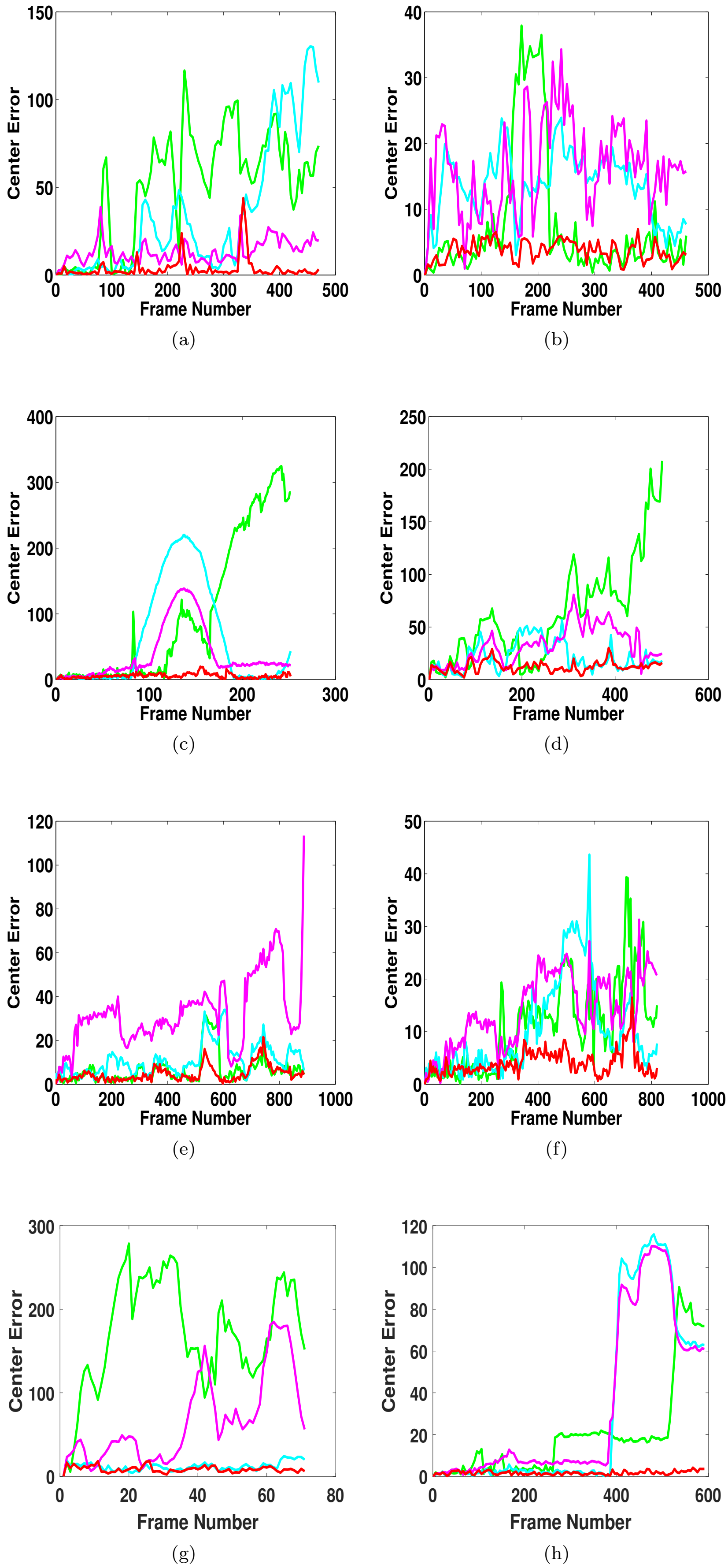


Fig. 9: The center error result

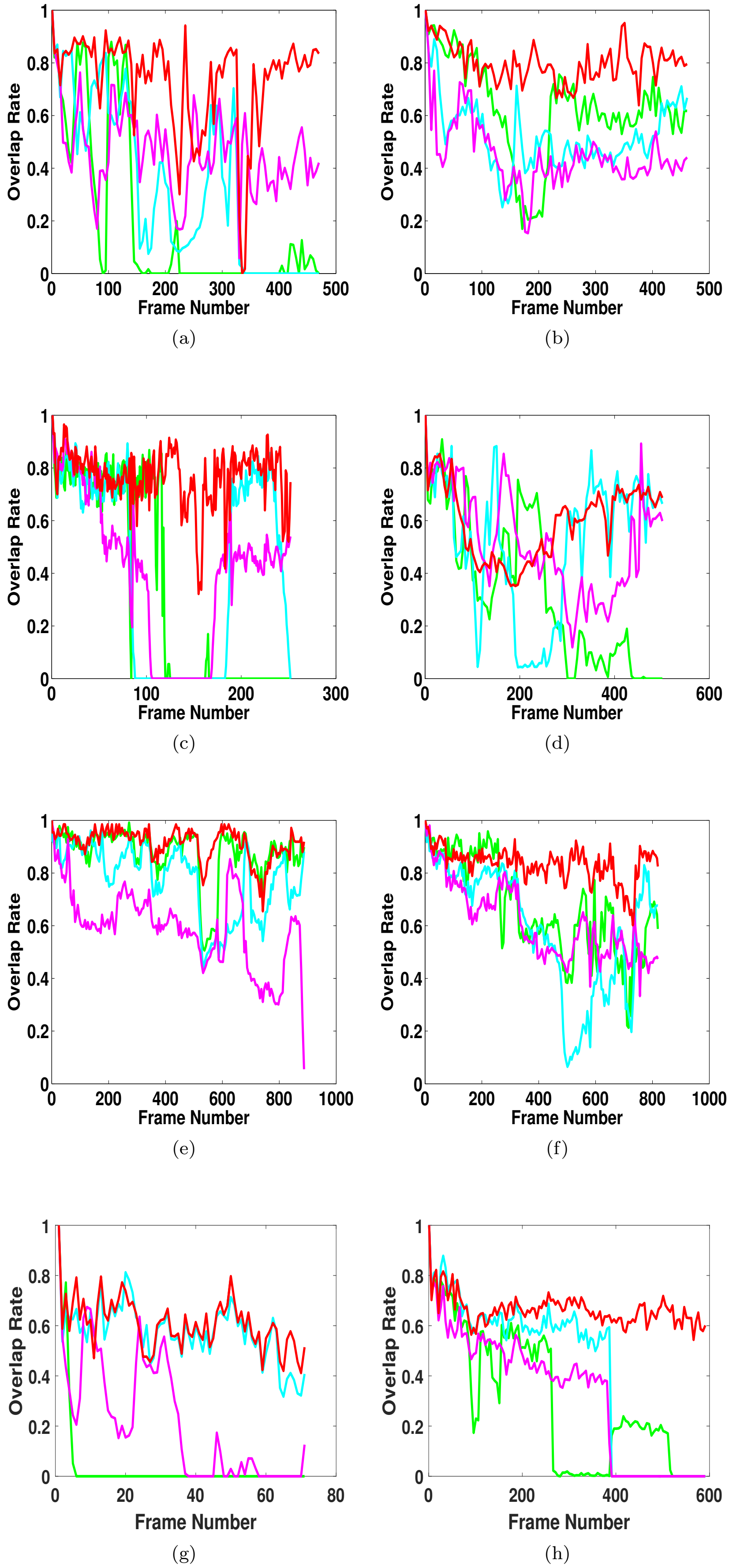


Fig. 10: The overlap result

overlap	L1	VTD	MIL	Ours
Cliffbar	24.8	34.6	13.4	3.3
David Indoor	7.6	13.6	16.2	3.2
DavidOutdoor	100.3	61.9	38.3	4.6
Girl	62.4	21.4	32.3	10.2
Occlusion1	6.5	11.1	32.3	3.8
Occlusion2	11.1	10.4	14.1	3.9
Deer	171.5	11.9	66.5	8.3
Stone	19.2	31.3	32.3	1.3
Average	50.4	24.5	30.7	4.8

Table. 1: The average center error

center error	L1	VTD	MIL	Ours
Cliffbar	0.2	0.3	0.5	0.7
David Indoor	0.7	0.6	0.5	0.8
DavidOutdoor	0.3	0.4	0.4	0.7
Girl	0.3	0.5	0.5	0.6
Occlusion1	0.8	0.7	0.6	0.9
Occlusion2	0.6	0.6	0.6	0.8
Deer	0.04	0.5	0.2	0.7
Stone	0.3	0.4	0.3	0.8
Average	0.37	0.50	0.45	0.75

Table. 2: The average overlap

at present. In addition, the average processing time of the proposed method is 52 fps and the slowest is more than 30 fps. Therefore, our method is valid and can run in a real time system.

5 Conclusion

The method in this paper combines the colour information and context feature, which makes it have robustness for appearance change of the object. It can work well even though occlusion and similar colour occur. Not only that, scale update information and online update are considered to make it perform better. In addition, it can run in a real-time system as the algorithm computed in frequency domain through Fourier Transform. Qualitative and quantitative experiments prove the effectiveness and efficiency of MMOT algorithm compared with existing methods. The next step of this work will try to compare the results on benchmark sequences with the VOT.

Acknowledgements The authors would like to acknowledge the support from the EU Seventh Framework Programme (FP7)-ICT under Grant No. 611391, Natural Science Foundation of China under Grant No. 51575412, 51575338 and 51575407, China Scholarship Council (grant no. 201508060340) and Research Project of State Key Lab of Digital Manufacturing Equipment & Technology of China under Grant No. DMETKF2017003.

References

1. Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 4705–4713, 2015.
2. H. Li, Y. Li, and F. Porikli, "DeepTrack: Learning Discriminative Feature Representations by Convolutional Neural Networks for Visual Tracking," in *Proceedings of the British Machine Vision Conference 2014*, pp. 56.1–56.12, 2014.
3. Y. YU, G. K. I. MANN, and R. G. GOSINE, "A SINGLE-OBJECT TRACKING METHOD FOR ROBOTS USING OBJECT-BASED VISUAL ATTENTION," *International Journal of Humanoid Robotics*, vol. 09, p. 1250030, dec 2012.
4. M. Heber, M. Godec, M. Rüther, P. M. Roth, and H. Bischof, "Segmentation-based tracking by support fusion," *Computer Vision and Image Understanding*, vol. 117, pp. 573–586, jun 2013.
5. Y. Wu, J. Lim, and M. H. Yang, "Online object tracking: A benchmark," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2411–2418, 2013.
6. L. Zilka, D. Marek, M. Korvas, and F. Jurcicek, "Comparison of Bayesian Discriminative and Generative Models for Dialogue State Tracking," in *Proceedings of the SIGDIAL 2013 Conference*, pp. 452–456, 2013.
7. W. Brendel and S. Todorovic, "Video object segmentation by tracking regions," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 833–840, IEEE, sep 2009.
8. S. Avidan, "Ensemble Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 261–271, feb 2007.
9. D. Wang, H. Lu, and Y.-W. Chen, "Incremental MPCA for Color Object Tracking," in *2010 20th International Conference on Pattern Recognition*, pp. 1751–1754, IEEE, aug 2010.
10. 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, pp. 125–141, may 2008.
11. R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. van den Hengel, "Part-Based Visual Tracking with Online Latent Structural Learning," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2363–2370, IEEE, jun 2013.
12. Y. Wu, H. Ling, J. Yu, F. Li, X. Mei, and E. Cheng, "Blurred target tracking by blur-driven tracker," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1100–1107, IEEE, nov 2011.
13. J. A. Lasserre, C. M. Bishop, and T. P. Minka, "Principled hybrids of generative and discriminative models," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 87–94, IEEE, 2006.
14. K. Zhang, L. Zhang, and M.-H. Yang, "Real-Time Compressive Tracking," pp. 864–877, 2012.
15. K. Zhang, L. Zhang, and M.-H. Yang, "Fast Compressive Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, pp. 2002–2015, oct 2014.
16. M. Fei, J. Li, and H. Liu, "Visual tracking based on improved foreground detection and perceptual hashing," *Neurocomputing*, vol. 152, pp. 413–428, mar 2015.
17. D. Wang, H. Lu, and C. Bo, "Fast and Robust Object Tracking via Probability Continuous Outlier Model," *IEEE Transactions on Image Processing*, vol. 24, pp. 5166–5176, dec 2015.
18. D. Wang, H. Lu, and M.-H. Yang, "Robust Visual Tracking via Least Soft-Threshold Squares," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, pp. 1709–1721, sep 2016.
19. B. Liu, J. Huang, L. Yang, and C. Kulikowsk, "Robust tracking using local sparse appearance model and K-selection," in *CVPR 2011*, pp. 1313–1320, IEEE, jun 2011.
20. M. Danelljan, F. S. Khan, M. Felsberg, and J. Van De Weijer, "Adaptive color attributes for real-time visual tracking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1090–1097, IEEE, jun 2014.
21. J. Kwon and K. M. Lee, "Visual tracking decomposition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1269–1276, IEEE, jun 2010.

22. K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M. H. Yang, "Fast visual tracking via dense spatio-temporal context learning," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8693 LNCS, pp. 127–141, Springer, Cham, 2014.
23. B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online Multiple Instance Learning," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 983–990, IEEE, jun 2009.
24. X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai, "Minimum error bounded efficient 1 tracker with occlusion detection," in *CVPR 2011*, pp. 1257–1264, IEEE, jun 2011.