2017 International Symposium on Nonlinear Theory and Its Applications,
NOLTA2017, Cancun, Mexico, December 4-7, 2017

Model Free Object Tracking Using CNN features and Color information

Keiichiro Adachi† and Kazuhiro Hotta†

†Department of Electrical and Electronic Engineering, Meijo University
1–501 Shiogamaguchi, Nagoya 468-8502, Japan
Email: 130442001@ccalumni.meijo-u.ac.jp, kazuhotta@meijo-u.ac.jp

Abstract Object tracking under occlusion is a challenging problem. In this paper, we propose a robust object tracking method under occlusion by adaptive integration of 2 kinds of trackers. Our tracking method is based on CNN features. However, the maximum value in the response map obtained by CNN features becomes small under occlusion. Thus, we can judge the occurrence of occlusion from the change of maximum value. When the target is occluded, we change the learning rate of a tracker using CNN features. A tracker using color information is also integrated because the tracker is robust to occlusion. We evaluate our method on the OTB100 dataset and the robustness to occlusion is evaluated using 49 videos which contain occlusion in the OTB100. Experimental results show that the proposed method improves the accuracy in comparison with conventional methods.

1. Introduction

In recent years, tracking function is useful in various applications such as digital cameras and surveillance cameras. If computer can track targets well, we obtain traffic information automatically. But visual object tracking is a challenging problem because occlusion frequently occurs in practical applications. Conventional methods used Histogram of Oriented Gradient (HOG) and color histogram [3, 6]. But those methods were low accuracy because HOG features and color histogram are not robust to appearance changes and fast motion.

In recent years, Convolutional Neural Network (CNN) gave good accuracy in variation kinds of recognition tasks. The features extracted by CNN are more robust to appearance changes than HOG features and color histogram. The tracker using CNN features called HCF [1] gave high accuracy, and it is robust to appearance changes. Thus we use HCF which gave high accuracy on the OTB dataset [9] as a baseline tracker. However, HCF is difficult to track the target accurately under occlusion because the learning rate for updating HCF model is fixed. When the target appears from occlusion again, the tracker cannot find true target. In this paper, we address this issue.

The maximum value in the response map obtained by HCF becomes small when occlusion occurs. This means that we can judge the occurrence of occlusion. We compute the average of the maximum value in response map of previous 10 frames. We compute difference between the average value and the maximum value in the response map at current frame. When the difference exceed threshold, we judge that the target is occluded and we change leaning rate of HCF.

When we detect occlusion, we also combine the response map obtained by TIPF [2] with that of HCF because TIPF is robust to occlusion. However, TIPF gave lower accuracy on the OTB100 than HCF because TIPF uses color histogram which is not robust to appearance changes. Although the accuracy of TIPF is not so high, it helps to track the target under occlusion. We detect the target location with the maximum value in the combined response map.

We evaluate our method on the OTB100 dataset which contains 100 videos. Since 49 videos in OTB100 includes occlusion, we use them to evaluate the robustness to occlusion. We use three evaluation measures; distance precision (DP), overlap precision (OP) and center location error (CLE). We see that the proposed method gives the best result in all evaluation measures in comparison with HCF. In particular, the accuracy for 49 videos with occlusion is much improved. The results show that our proposed method is robust to occlusion.

This paper is constructed as follows. We explain two conventional methods and the details of our proposed method in section 2. Experimental results on the OTB100 dataset and comparison results with conventional methods are shown in section 3. Conclusion and future works are described in section 4.

2. Proposed Method

HCF [1] gave high accuracy on the large benchmark dataset called OTB100. But HCF is not robust to occlusion because the learning rate for updating model is fixed. In this paper, we address this issue. The similarity score decreases drastically when the tracking target is occluded. If we use the fact effectively, we make HCF a robust tracker under occlusion.

TIPF [2] is a robust tracker to occlusion because TIPF updates the model according to the occlusion adaptively and uses previous information. But TIPF gave lower accuracy on the OTB100 than HCF because TIPF uses color histogram which is not robust to appearance changes. We detect the target location with the maximum value in the combined response map.

The overview of our approach is shown in Figure 5. When the maximum value of the response map obtained by HCF is small, we judge that the target is occluded and change the learning rate. We also combine the response

map obtained by TIPF with that of HCF to be robust to occlusion.

We explain tracking methods based CNN features and color information in section 2.1 and 2.2. We explain the judgement of occlusion in section 2.3. Section 2.4 explains the combination of two response maps. How to change learning rate is explained in section 2.5.

2.1 Tracking based CNN features

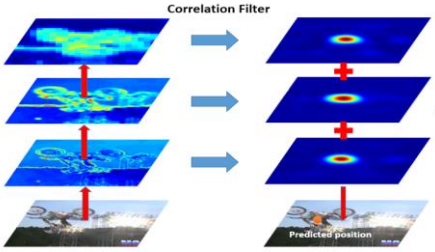

Figure 1: HCF algorithm

HCF uses the correlation filter of features extracted from the VGG net which is trained by ImageNet dataset. VGG net has 16 layers for feature extraction and 3 layers for classification. The feature vector extracted from the k-th convolutional layer is denoted $\boldsymbol{x^k}$ whose size is M×N×D where M, N, D indicate the width, height, and the number of channels. We consider the circular shifts of $\boldsymbol{x}$ as training samples. Each shifted sample $\boldsymbol{x}_{m,n,d}$ has label $y(m,n)$ with a 2D Gaussian shape. The weight w of the d-th channel in the k-th convolutional layer is obtained by solving the following equation.

$$\boldsymbol{w}^d = \underset{w}{\operatorname{argmin}} \sum_{m,n} \left\| \boldsymbol{w} \cdot \boldsymbol{x}_{m,n} - y(m,n) \right\|^2 + \lambda \|\boldsymbol{w}\|_2^2. \quad (1)$$

The weight vector in Fourier domain for the d-th channel can be written as

$$W^d = \frac{Y \odot \bar{X}^d}{\sum_{i=1}^{D} X^i \odot \bar{X}^i + \lambda}. \quad (2)$$

In Eq. (2), Y, X are the Fourier transformation form of y, $\boldsymbol{x}$. The bar means complex conjugation and $\odot$ is the element-wise product. The feature vector of a search region is denoted as $\mathbf{z}$ whose size M × N × D. The response map of the k-th correlation filter is computed by

$$S^k = F^{-1}(W^d \cdot \bar{Z}^d), \quad (3)$$

where F^{-1} denotes inverse Fourier transform. We predict the target position with the maximum value in the response map.

Model is updated with the fixed learning rate η as

$$W_t^d = (1-\eta) W_{t-1}^d + \eta W^d. \quad (4)$$

2.2 Tracking based color information

TIPF [2] gave high accuracy on the PETS2009S2L1 dataset. TIPF uses the color histogram to calculate the probability and predicts target location by using three probability maps.

The first probability map is Object-Surroud Model which compares the current target region with the surrounding reigions. The second map is Object-Distracors Model which compares the current target region with the similar regions (distractors). The third map is Object-Past Model which compares the target region with previous target regions. The three models are shown as

$$P_{O,S}(\mathrm{b}) = \frac{H_O(b)}{H_S(b)}, \quad (5)$$

$$P_{O,D}(\mathrm{b}) = \frac{H_O(b) \times d}{H_O(b) \times d + H_D(b)}, \quad (6)$$

$$P_{O,P}(\mathrm{b}) = 1 - \frac{H_O(b) \times f}{H_O(b) \times f + H_P(b)}, \quad (7)$$

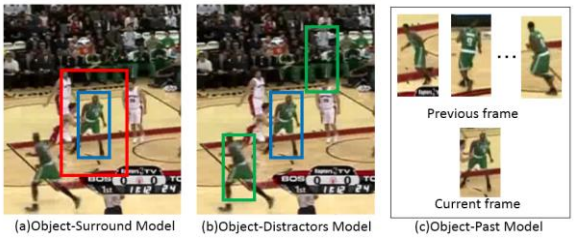

Figure 2: Three models used in TIPF

where b is the bin of the histogram assigned to the color components and d is the number of the distractors. $H_0(b)$ is a target region shown as blue rectangles in Figure 2. $H_s(b)$ is a surrounding region shown as red rectangle in Figure 2 and $H_D(b)$ are distractors shown as green rectangles in Figure 2. $H_p(b)$ is the number of votes in past targets and f is the number of frames whose similarities are high. We compute the sum of three probability maps as

$$P(\mathrm{b}) = \frac{1}{3} P_{O,S}(\mathrm{b}) + \frac{1}{3} P_{O,D}(\mathrm{b}) + \frac{1}{3} P_{O,P}(\mathrm{b}). \quad (8)$$

Finally, TIPF predicts the target location with the maximum probability in a search window. TIPF updates a model as

$$P_t(b) = (1-\gamma) P_{t-1}(b) + \gamma P(b), \quad (9)$$

where t is current frame and γ is learning rate. Learning rate is changed according to the similarity.

2.3 Judgment of Occlusion

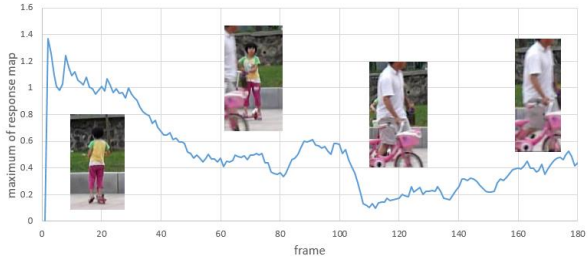

Figure 3: Change of the maximum value in response map

We use only the response map obtained by HCF to judge the occurrence of occlusion. When the target is occluded, the maximum value of response map obtained by HCF becomes small as shown in Figure 3.

In experiments, the maximum value in response map obtained by HCF is not stable in the first 10-20 frames. Thus, we combine two response maps obtained by HCF and TIPF for only the first 20 frames. After 20 frames, we compute the average of the maximum values in the response map obtained by HCF for previous 10 frames. If the difference between the average and the maximum value at current frame is more than threshold, we judge that occlusion occurs. We set the threshold to 0.22 empirically.

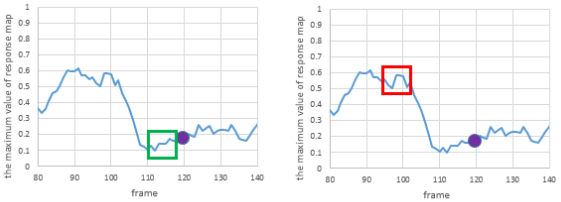

Figure 4: The maximum value in response map

If the target is occluded for over 10 frames, the average value of previous 10 frames (green rectangle in Figure 4) becomes small. When the target is occluded at purple point in Figure 4 (a), the difference from the average value is small and we cannot detect occlusion. Thus, we compute the average of 10 previous frames except for frames that occlusion is detected. Namely, we use the average of red rectangle in Figure 4 (b) when we detect occlusion in green rectangle. The difference between the average and the maximum value at current frame is large. Thus, when the target appears from occlusion again, we can track it correctly.

2.4 Combination of two response maps

When we judge occlusion, we combine probability maps obtained by TIPF and HCF. We combine two response maps after the following three processes. The first process is that we change the size of response map obtained by HCF to be the same as TIPF because initial two response maps are different size. The second process is that we normalize two response maps because the maximum value in a response map obtained by TIPF is much different. The maximum of response map obtained by TIPF is bigger than that of HCF. By the normalization, we can make the range of two response maps similar. The third process is that the weights for integrating two response maps are changed. Since TIPF is robust to occlusion, the weight of TIPF is set to two times larger value than that of HCF.

2.5 Selection of learning rate

HCF updates a model using learning rate shown in Eq. (9). The learning rate η of original HCF was set to 0.01. However, when the target is occluded, HCF updates a model with fixed learning rate. Figure 3 shows that the maximum value of the response map obtained by HCF when the learning rate is fixed. The similarity between the

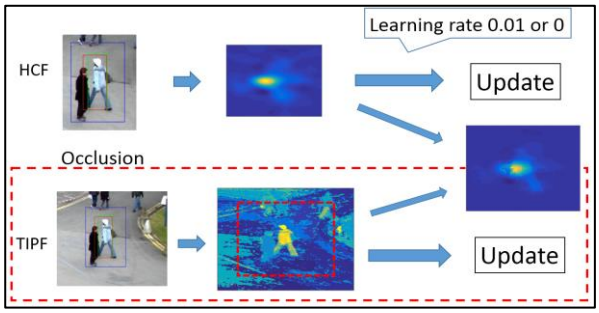

Figure 5: The proposed approach

model and the cropped patch is large because the model trains the occluded target. Thus, if the target appears again from the occlusion, the tracker cannot find the target. To avoid this problem, learning rate is set to 0 when we detect the occurrence of occlusion. Namely we do not update HCF model under occlusion.

TIPF changes learning rate automatically according to similarity. Thus TIPF updates the model with large learning rate when the target is not occluded and TIPF does not update when the target is occluded.

4. Experiments

We use the OTB100 dataset which contains 100 video sequences for evaluation. The target size and the length of 100 videos are different. Since 49 videos in the OTB100 contain occlusion in OTB100, they are used to evaluate the robustness to occlusion.

We use Distance Precision (DP), Overlap Precision (OP) and Center Location Error (CLE) as evaluation measures. DP is the number of frames that Euclidean distance between predicted center of target and ground truth (center) is within 20 pixels. OP is the number of frames that overlapping ratio between predicted bounding box and ground truth (bounding box) is below 0.5. CLE is the average Euclidean distance between predicted center and ground truth. We compute DP, OP and CLE of all sequences, and average DP, OP and CLE are used for evaluation. Higher DP and OP are better and lower CLE score is better.

We compare the proposed method with conventional HCF [1] and PITF [2] on the OTB100 dataset. Proposed method uses both adaptive learning rate and integration of 2 response maps. We also compare the proposed method (Proposed) with the proposed method without selection of learning rate (Proposed-SLR) and the proposed method without combination of two response maps (Proposed-CTR).

TABLE1: The result on the OTB100 dataset

Tracker	DP[%]	OP[%]	CLE[pixels]
HCF[1]	84.2	66.1	22.2
TIPF[2]	46.9	55.0	70.7
Proposed	85.3	68.4	19.6
Proposed-SLR	84.1	66.3	20.4
Proposed-CTR	84.8	67.7	20.2

TABLE2: The result on 49 videos which contains occlusion in the OTB100 dataset

Tracker	DP[%]	OP[%]	CLE[pixels]
HCF[1]	77.6	61.3	31.4
TIPF[2]	47.2	55.5	74.3
Proposed	80.2	64.7	26.4
Proposed-SLR	77.3	60.6	28.3
Proposed-CTR	78.7	63.1	28.5

Experimental results on the OTB100 are shown in Table 1. We see that the proposed method gives the best result in all evaluation measures. DP of our method improves 1.1%, OP of our method improves 2.3% and CLE of our method improves 2.6 pixels in comparison with conventional HCF. The proposed method improved in comparison with our method without selection of learning rate (Proposed-SLR) and without combination of two response maps (Proposed-CTR). This result shows that both selection of the learning rate and combination of two response maps are effective.

Experimental results on only 49 videos which contain occlusion in OTB100 are shown as Table 2. We see that the proposed method also gives the best result in all evaluation measures. Our method improves 2.6%, 3.4%, and 5.0 pixels in comparison with conventional HCF on the same dataset. The improvement of the proposed method is larger than the result on all 100 videos. This means that the robustness to occlusion is much improved.

The results demonstrate that the usage of both adaptive learning rate and combination with TIPF is effective to improve the tracking accuracy. When target appears after occlusion, our proposed tracker can find the target because color similarity is effective to this case and HCF tracker is not updated when we detect occlusion. Examples improved by the proposed method are shown in Figure 6. Our proposed method finds the target after occlusion.

We consider why our method gave the best accuracy on the OTB100 and 49 videos with occlusion. The combination of two response maps is effective when the search region has the similar targets after occlusion. For example, there is the similar object with the target in a search region in Figure 6. All evaluation measures are improved because CNN features cannot distinguish between any similar objects and color information can distinguish. The selection of learning rate is effective when the target is occluded for a long time. In the video called FaceOCC1, the target is occluded for a long time. However, HCF model is not updated under occlusion and our method can track it well.

We implement our tracker by MATLAB and MatConvNet on the PCA with Intel I7-6700T 2.80GHz CPU. The speed of our tracker was 0.62 FPS and that of HCF was 1.60 FPS.

5. Conclusion

We proposed the robust tracking method under occlusion. When we detect occlusion, we combine the response map obtained by HCF and TIPF and we change the learning rate. We demonstrated the effectiveness of the proposed method through the comparison with conventional methods on the OTB100 dataset.

Figure 6: The example improved by our method

Our proposed tracker has two issues. The first issue is that the speed of the proposed tracker is slow because it takes much time for computing features in the VGGnet. Thus we make a shallower network by using distillation [10] from the VGGnet.

The second issue is that OP of our tracker is low because the size of bounding box is fixed. Thus we make various sizes of search regions and extract the features by CNN. We search the most similar features with the features of the target at first frame. We make the same size of bounding box with the search region that has the best similar features of the target.

References

[1] C.Ma et al. “Hierarchical Convolutional Features for Visual Tracking,” International Conference on Computer Visual, pp.3074-3082, 2015.

[2] H . Takada , K.Hotta and . Janney, “Human Tracking in Crowded Scenes Using Target Information at Previous Frames,” International Conference on Pattern Recognition, pp.3074-3082, 2016.

[3] H.Possegger, T.Mauthner and H.Bischof, “In Defense of Color-based Model-free Tracking,” Computer Vision and Pattern Recognition, pp.2113-2120, 2015.

[4] H.Takada and K.Hotta, “Robust Human Tracking to Occlusion in Crowded Scenes,” International Conference on Digital Image Computing Techniques and Applications, pp.645-652, 2015.

[5] S.Zokai and G.Wolberg, “Image Registration Using Log-Polar Mappings for Recovery of Large-Scale Similarity and Projective Transformations,” IEEE Trans. on Image Processing, pp.1422-1434, 2005.

[6] N. Dalal and B. Triggs. “Histograms of oriented gradients for human detection,” Conference on Computer Vision and Pattern Recognition, 2005.

[7] G.Wolberg and S.Zokai, “Robust image registration using log-polar transform,” International Conference on Image Processing, pp.493-496, 2000.

[8] G.Hinton, V.Oriol and D.Jeff, "Distilling the knowledge in a neural network," arXiv preprint arXiv:1503.02531, 2015.

[9] Y.Wu, J. Lim, and M.-H. Yang. "Object tracking benchark,” IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.1834-1848, 2015.