Sophisticated Tracking Framework with Combined Detector

Gwangmin Choe[1,2], Tianjiang Wang[1(✉)], Qi Feng[1], Chunhwa Choe[2], Sokmin Han[2], and Hun Kim[2]

[1] School of Computer Science and Technology,
Huazhong University of Science and Technology,
Wuhan 430074, People's Republic of China
cca2005@foxmail.com, {tjwang,qfeng}@hust.edu.cn
[2] Kim Il Sung University, Pyongyang, Democratic People's Republic of Korea
hunkim64@163.com

Abstract. This paper proposes a combined detector containing the background subtraction and the object appearance model-based detector. This is used to solve such problems as linking, overlapping, false object detecting etc. Then, we give a non-linear multi-mode tracker with the combined detector to solve such problems as sudden appearance changes and long-lasting occlusions, etc. Finally, we test our proposed person tracking framework in multi-object tracking scenario. Experimental results demonstrate that our proposed approaches have promising discriminative capability in comparison with other ones.

Keywords: Detector · Tracker · Model of object appearance · Background subtraction · Shadow removal · Combined detector · Non-linear multi-mode tracker · Particle filter · Person re-identification

1 Introduction

Modeling an object appearance in tracking are mainly classified to two approaches: static and adaptive. Static models is proposed in the context of using assumption that the object appearance change is limited and known [1]. From this assumption, it is clear that unexpected changes of the object appearance can not be tracked. Adaptive methods are proposed to address this drawback, which update the object model during tracking [2]. These approaches assume that every update is correct. Under this underlying assumption, error of the model accumulated over time and drift are caused by every incorrect update. The drift problem has been addressed by introduction of so called visual constraints [3]. Even though this approach demonstrated increased robustness and accuracy, its performance was tested only on videos where the object was in the field of view. In scenarios where an object moves in and out of the frame, object re-detection is essential. Object detection have been extensively studied

T. Wang – He is currently a Professor with the School of Computer Science, Huazhong University of Science and Technology, Wuhan, China.

Y.-J. Zhang (Ed.): ICIG 2015, Part III, LNCS 9219, pp. 402–409, 2015.
DOI: 10.1007/978-3-319-21969-1_35

[4] and a range of ready-to-use object detectors are available [5] which enable tracking-by-detection. Apart from expensive off-line training, the disadvantage of tracking-by-detection is that all objects have the same model and therefore the identities can not be distinguished. An object tracking algorithm that splits the object model into three parts with different lifespan, is proposed to solve this problem [6]. This makes the tracker suitable for low-frame rate videos but the longest period the face can disappear from the camera view is limited. Another class of approaches for face tracking was developed as part of automatic character annotation in video [7]. These systems can handle the scenario considered in this paper, but they have been designed for off-line processing and adaptation for real-time tracking is not straightforward.

An approach called Tracking-Learning-Detection (TLD) has been designed for long-term tracking of arbitrary objects in unconstrained environments [8]. Learning part of TLD was analyzed in [9]. The object was tracked and simultaneously learned in order to build a detector that supports the tracker once it fails. The detector was build upon the information from the first frame as well as the information provided by the tracker. Apart from this, the detector was build upon the information from the gray-scale distribution, i.e. the model of object appearance. This means that, given a moving object, these approaches discard the information from the motion of object, i.e. the variation of foreground. These approaches, therefore, may result in tracking an object far away from the location with the real variation of foreground. Beside this problem, shadow removal was not considered in this detector. Also, the tracker can not guarantee the non-linear multi-mode tracking, i.e. this can not very well adapt to sudden appearance changes, long-lasting occlusions etc.

This work has three major contributions. First, a combined detector containing the background subtraction and the object appearance model-based detector is proposed to solve such problems as linking, overlapping, false object detecting etc. Second, a non-linear multi-mode tracker with the combined detector is used to solve such problems as sudden appearance changes and long-lasting occlusions, etc. The non-linear multi-mode tracker is chosen as the particle filter with spline resampling and global transition proposed in [12]. Also, a person re-identification is used to numbering person in the context of multi-target tracking.

2 The Proposed Method

In this section, we propose first a combined detector containing the background subtraction and the object appearance model-based detector. Then we give a non-linear multi-mode tracker with the combined detector.

2.1 Combined Detector

Background Subtraction. Background subtraction involves calculating a reference image, subtracting each new frame from this image and thresholding the result. What results is a binary segmentation of the image which highlights

regions of non-stationary objects. Here, a color or gray-scale video frame is compared with a background model to determine whether individual pixels are part of the background or the foreground.

Given a series of either gray-scale or color video frames, methods based on background mixture model compute the foreground mask using Gaussian mixture models (GMM). In our framework, the adaptive background mixture model is used to compute the foreground mask. This method allows system learn faster and more accurately as well as adapt effectively to changing environments.

Shadow Removal Based on Level-Thresholding. The strategy of our proposed person tracking framework is to:perform first the shadow removal using a strong threshold even if foreground pixels are also removed simultaneously, then implement shadow removal using a weakness threshold in bounded local regions. The first step prevents background pixels retained by the shadow removal. Some parts foreground pixels will still lost when the only combination of the above two filters, and the lost pixels tend to be the contact parts among outlines. The main reasons result in losing foreground pixels are that the texture of background is unobvious and current pixels are in the penumbra. The second step guarantees regain of foreground pixels lost in the first step. In first step, the intensity ratio between the background and the current frame is calculated by Eq. (1), then the pixel is a shadow when it meets the formula Eq. (2).

$$\begin{cases} E_r(i,j) = \dfrac{\min(B_r(i,j), Cur_r(i,j))}{\max(B_r(i,j), Cur_r(i,j))} \\ E_g(i,j) = \dfrac{\min(B_g(i,j), Cur_g(i,j))}{\max(B_g(i,j), Cur_g(i,j))} \\ E_b(i,j) = \dfrac{\min(B_b(i,j), Cur_b(i,j))}{\max(B_b(i,j), Cur_b(i,j))} \end{cases} \tag{1}$$

$$M(i,j) = \begin{cases} 1, E_r(i,j) < T_1 \; and \; E_g(i,j) < T_1 \; and \; E_b(i,j) < T_1 \\ 0, otherwise \end{cases} \tag{2}$$

where $E_r(i,j)$, $E_g(i,j)$ and $E_b(i,j)$ are the intensity ratio images or the difference images for three channels; $B_r(i,j)$, $B_g(i,j)$ and $B_b(i,j)$ are background images; $Cur_r(i,j)$, $Cur_g(i,j)$ and $Cur_b(i,j)$ are current frames; $M(i,j)$ is the binary mask; T_1 is the threshold for the first level of shadow removal. In the binary mask, pixels with a value of 1 correspond to the foreground, and pixels with a value of 0 correspond to the background. Then, morphological operations on the resulting binary mask are performed to remove noisy pixels and to fill the holes in the remaining blobs. In second step, a weakness threshold is used to implement shadow removal in bounded local regions and recovery foreground pixels lost in the first step.

$$M(i_b, j_b) = \begin{cases} 1, E_r(i_b, j_b) < T_2 \; and \; E_g(i_b, j_b) < T_2 \; and \; E_b(i_b, j_b) < T_2 \\ 0, otherwise \end{cases} \tag{3}$$

where T_2 is the threshold for the second level of shadow removal, and i_b and j_b are pixel coordinates in bounded local regions. Then, morphological operations

on this binary mask are performed to remove noisy pixels and to fill the holes in the remaining blobs.

Combined Detector with the Object Appearance Model-Based Detector. This part detects people in an input image using the Histogram of Oriented Gradient (HOG) features and a trained Support Vector Machine (SVM) classifier, and detects unoccluded people in an upright position.

Local object appearance and shape can often be characterized rather well by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions. This is implemented by dividing the image window into small spatial regions, for each region accumulating a local 1-D histogram of gradient directions or edge orientations over the pixels of the region. The representation is formed by the combined histogram entries. For better invariance to illumination, shadowing, etc., contrast-normalizing is also used for the local responses before using them. This can be done by accumulating a measure of local histogram "energy" over somewhat larger spatial regions ("blocks") and using the results to normalize all of the cells in the block. We will refer to the normalized descriptor blocks as Histogram of Oriented Gradient (HOG) descriptors. Human detection chain is given by tiling the detection window with a dense (in fact, overlapping) grid of HOG descriptors and by using the combined feature vector in a SVM based window classifier. One-class SVM has been widely used for outlier detection. Only positive samples are used in training. The basic idea of one-class SVM is to use a hypersphere to describe data in the feature space and put most of the data into the hypersphere. The problem is formulated into an objective function as follows:

$$\min_{\mathcal{R}\in\mathbb{R},\xi\in\mathbb{R}^l,c\in F} R^2 + \frac{1}{vl}\sum_i \xi_i, \tag{4}$$

$$\|\Phi(X_i) - c\|^2 \leq \mathbb{R}^2 + \xi_i, \forall i \in \{1, ..., l\} : \xi_i \geq 0 \tag{5}$$

where $\Phi(X_i)$ is the multi-dimensional feature vector of training sample X_i, l is the number of training samples, R and c are the radius and center of the hypersphere, and $v \in [0, 1]$ is a trade-off parameter. The goal of optimizing the objective function is to keep the hypersphere as small as possible and include most of the training data. The optimization problem can be solved in a dual form by QP optimization methods, and the decision function is:

$$f(X) = R^2 - \|\Phi(X_i) - c\|^2, \tag{6}$$

where $\|\Phi(X_i) - c\|^2 = k(X, X) - 2\sum_i \alpha_i k(X_i, X) + \sum_{i,j} \alpha_i \alpha_j k(X_i, X_j)$, and α_i and α_j are the parameters for each constraint in the dual problem. In our task, we use the radius basis function (RBF) $k(X, Y) = \exp\{-\|X - Y\|^2/2\sigma^2\}$ as kernel in one-class SVM to deal with high-dimensional, non-linear, multi-mode distributions. The decision function of kernel one-class SVM can well capture the density and modality of feature distribution.

The model is specified as either 128×64 or 96×48, whose size is the image size used for training indicated by the pixel dimensions. The images used to train

the models include background pixels around the person. Therefore, the actual size of a detected person is smaller than the training image size.

$$M(i,j) = \begin{cases} 1, if\ person\ is\ detected \\ 0, otherwise \end{cases} \tag{7}$$

where $M(i,j)$ is the binary mask. Then, morphological operations on the resulting binary mask are performed to remove noisy pixels.

2.2 Non-linear Multi-mode Tracker with Combined Detector

Many trackers were build upon the model of object appearance. This means that, given a moving object, these detectors discard the information from the motion of object, i.e. the variation of foreground. Therefore, this may result in tracking an object far away from the location with the real variation of foreground. Our detector combines the information from the gray-scale distribution with the variation of foreground. This makes the tracker to search an object at the place consistent with the location where exists the real variation of foreground. This approach compares the result of PF(Particle Filter) with one of background subtraction, after PF implemented. That is, after PF implemented, this re-checks if object detected by background subtraction exists in the place as object region estimated by PF or how much they are overlapping. If existed or overlapped, the result of PF will be corrected to one of background subtraction, and if not, there will be two proposals. If the confidence of PF result is not sufficient, then it will be canceled completely, and if not, a new object region will be added to the result of background subtraction to compensate error of background subtraction. In fact, when object is overlapped seriously, real object region may be considered as background region from incompleteness of person detector. These regions will be just recovered by object appearance model-based detector.

$$M(i,j) = \begin{cases} 1, if\ the\ large\ confidence,\ existed\ or\ overlapped \\ 0, otherwise \end{cases} \tag{8}$$

where $M(i,j)$ is the binary mask.

Finally, PF is implemented around extended regions of each bounding boxes obtained by background subtraction, i.e. size of each extended region is lager than ones bounded by background subtraction. All coordinates for PF are calculated as relative coordinates for each extended regions.

3 Experimental Results

We evaluate the performance of our proposed tracking framework. Our experiment should be aimed at the relative evaluation for our proposed approaches themselves in relation with the previous approach. The sophisticated feature descriptor, of course, may be used to this tracking scenario to obtain high

accuracy of tracking. However, our experiment should be aimed at the relative evaluation for our proposed tracking framework in relation with previous approaches. Therefore, these experiments do not use other feature descriptors. We test our proposed approach and the previous approach in several challenging image sequences.

The performance evaluation includes two parts. The first part contains the evaluation for the performance of our proposed tracking framework in image sequences given a single example of a specific object. The second part compares the performance of our proposed tracking framework with one of the previous framework in the context of tracking multiple object.

All experiments are conducted on an Intel® Core(TM) i5 2.40 GHz PC with 4 GB memory(1.32 GHz). The real image sequences are available at http://www.ces.clemson.edu/~stb/research/headtracker and under MATLAB(2013a) directory.

The first set of experiments is implemented in a popularly used real video sequence containing 500 frames with resolution of 128×96 pixels, respectively. Figure 1 show the comparison of tracking results in real color video, the absolute error for every frame and the error histograms for two approaches, respectively. Red box indicates the result of our approach, and blue for the TLD. The experiment results show our proposed approach has also a robust performance for real video sequences. As a result, the tracker based on proposed approaches shows the most satisfactory performance among two trackers.

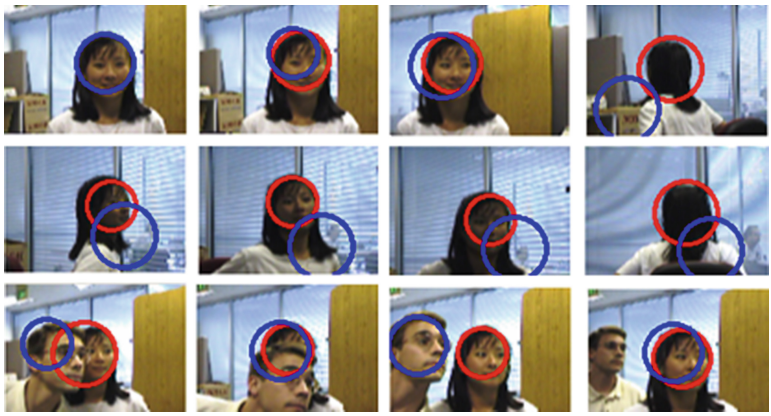
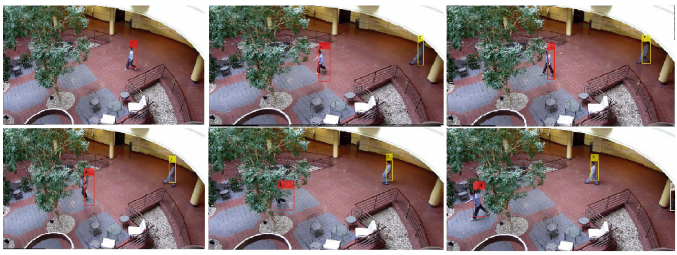


Fig. 1. Comparison of tracking result on real color video; Red box indicates the result of our approach, and blue for the TLD; Shown are frames 0, 3, 22, 98(top); 117, 126, 135, 188(middle); and 427, 457, 471, 500(bottom) (Color figure online).

Next, the second set of experiments is implemented in a color image sequence containing motion of multiple object. This video sequence contains 480×360-pixel color images. This video contain such variations as model distortion, occlusion, appearance of multi-objects and noise etc. The purpose of this experiment

(a)



(b)

Fig. 2. Comparison of tracking result on color video with motion of multiple object; TLD(a) and our proposed approach(b); Shown are frames 30, 43, 45(top); and 48, 54, 65(bottom) (Color figure online).

is to evaluate the robustness of our proposed approach under the configuration of multiple object tracking. The result of this experiment is shown in Fig. 2. It can be seen clear that our tracking framework has better performance for occlusion and multiple object tracking than TLD.

All the above experimental results prove that our proposed tracking framework is more robust and accurate compared with the other approaches. The other important thing we want to emphasize here is that our approach may also obtain competitive tracking results on other image datasets.

4 Conclusion

In this paper, we first proposed the combined detector containing the background subtraction and the object appearance model-based detector. This was used to solve such problems as linking, overlapping, false object detecting etc. Then, the non-linear multi-mode tracker with the combined detector was proposed to solve such problems as sudden appearance changes and long-lasting occlusions, etc. Finally, we tested our proposed person tracking framework in single-object and multi-object tracking scenario. Future work should be aimed at extending our proposed tracking approaches to connecting with more sophisticated feature descriptors and similarity measures such as the geogram [13], SOG [14] and AIBS [15].

Acknowledgments. This research is founded by the grant of The Key Technology Research of Multi-Camera Cooperative Awareness and Data Fusion for Smart Community in Wuhan City No. 2014010202010110.

References

1. Isard, M., Blake, A.: Condensation - conditional density propagation for visual tracking. IJCV **29**(1), 5–28 (1998)
2. Lim, J., Ross, D., Lin, R.S., Yang, M.H.: Incremental learning for visual tracking. In: NIPS (2005)
3. Kim, M., Kumar, S., Pavlovic, V., Rowley, H.: Face tracking and recognition with visual constraints in real-world videos. In: CVPR (2008)
4. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR (2001)
5. Kalal, Z., Matas, J., Mikolajczyk, K.: Weighted sampling for large-scale boosting. In: BMVC (2008)
6. Li, Y., Ai, H., Yamashita, T., Lao, S., Kawade, M.: Tracking in low frame rate video: a cascade particle filter with discriminative observers of different lifespans. In: CVPR (2007)
7. Everingham, M., Sivic, J., Zisserman, A.: Taking the bite out of automated naming of characters in TV video. IVC **27**, 545–559 (2009)
8. Kalal, Z., Matas, J., Mikolajczyk, K.: Online learning of robust object detectors during unstable tracking. In: OLCV (2009)
9. Kalal, Z., Matas, J., Mikolajczyk, K.: P-N Learning: bootstrapping binary classifiers by structural constraints. In: CVPR (2010)
10. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face recognition: a literature survey. CSUR **35**(4), 399–458 (2003)
11. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. IJCAI **81**, 674–679 (1981)
12. Choe, G., et al.: Visual tracking based on particle filter with spline resampling. Multimedia Tools Appl. (2014). doi:10.1007/s11042-014-1960-z
13. Choe, G., et al.: Moving object tracking based on geogram. Multimedia Tools Appl. (2014). doi:10.1007/s11042-014-2150-8
14. Gong, L., Wang, T., Liu, F., Chen, G.: A Lie group based spatiogram similarity measure. In: ICME (2009)
15. Choe, G., et al.: An advanced association of particle filtering and kernel based object tracking. Multimedia Tools Appl. (2014). doi:10.1007/s11042-014-1993-3
16. Choe, G., et al.: Particle filter with spline resampling and global transition model. IET Comput. Vision **9**(2), 184–197 (2015)