Particle Filter Tracking With Online Multiple Instance Learning

Zefeng Ni, Santhoshkumar Sunderrajan, Amir Rahimi, B.S. Manjunath
Department of Electrical and Computer Engineering, University of California Santa Barbara
{*zefengni,santhosh,rahimi,manj*}*@ece.ucsb.edu*

Abstract

This paper addresses the problem of object tracking by learning a discriminative classifier to separate the object from its background. The online-learned classifier is used to adaptively model object' appearance and its background. To solve the typical problem of erroneous training examples generated during tracking, an online multiple instance learning (MIL) algorithm is used by allowing false positive examples. In addition, particle filter is applied to make best use of the learned classifier and help to generate a better representative set of training examples for the online MIL learning. The effectiveness of the proposed algorithm is demonstrated in some challenging environments for human tracking.

1. Introduction

Visual tracking remains to be a challenging problem despite many years of research in computer vision. Many existing algorithms fail because of significant variations in object's appearance and/or its background. To adapt to such appearance variations, recently, many researchers [1, 2, 3, 5] have started to apply machine learning algorithms to learn a discriminative classifier to separate the object from its background. These approaches treat tracking as a binary classification problem, and hence often referred as "tracking by detection".

As pointed in [2], one of the main challenges with online learning is the generation of training examples. One common method is to take positive examples around current location of the object and negative examples from the region outside of it. This way of generating training examples could easily cause a significant drift after a period of time because of the noisy nature of the tracked object's location. Instead, a richer set of positive examples can be sampled from a neighborhood around the current location of the object. However, multiple positive examples might confuse many learning algorithms while finding the most discriminative features with some false positive examples. To overcome this issue, an online multiple instance learning (MIL) approach is proposed in [2] to learn a classifier from multiple positive and negative training examples.

MIL approach has been successfully applied for object detection [7, 10]. The idea is to group training examples into bags and assign labels to the bags rather than individual examples. A bag is considered to be positive as long as it contains at least one positive example. The learning algorithm then has the flexibility to define the most likely positive instance(s) while learning the classifier and hence it is robust to a few false positive examples.

Compared with object detection where training examples are often generated manually, training examples in tracking has more ambiguity because they are generated online according to the tracked object's estimated location. MIL approaches are expected to give significant benefits as demonstrated in [2]. To update the classifier at each frame, an online MILBoost learning algorithm is proposed, the first such algorithm utilizing MIL for visual tracking as claimed in [2]. The main contributions of this paper can be summarized as

- Given a learned discriminative classifier, a probabilistic approach is combined with particle filtering to track the object. This approach shows an improvement when comparing with the greedy strategy. The training examples are generated based on particle distribution over the image frame. On the other hand, the learned classifier gives a natural way to re-weight particles for each new frame.

- Compared with [2], at each frame, an more efficient online MILBoost algorithm is proposed to update the classifier with new examples from the current frames while still maintaining information learned from the previous frames (See Section 2.2 for a detailed analysis).

2. Proposed Methodology

Generally, any tracking system consists of three main components: image representation, appearance model and motion model. In this paper, image representation consists of a set of Haar-like features[9], and a special culture color histogram [11]. Each feature m corresponds to a weak classifier $h_m \in \{+1, -1\}$ and contributes to the discriminative classifier $\mathbf{H} = \sum_k \alpha_k h_k$, which forms the appearance model of the object. Given an image patch x, and its binary label $y = 1$ indicating the presence of the tracked object, the instance probability $p(y = 1|x)$ ($p(y|x)$ hereafter) is modeled as

$$p(y|x) = \sigma(\mathbf{H}(x)) = \frac{1}{1 + e^{-\mathbf{H}(x)}}. \quad (1)$$

At frame t, the tracker maintains the object state* $\mathbf{O}_t = [row, col, scale]^{\mathrm{T}}$ with a particle set $\{\mathbf{O}_t^{(l)}, \pi_t^{(l)}\}$ where $[row, col]$ is the object center position on the image plane. Given a particle set from the previous frame $t-1$, a basic sequential importance re-sampling*(SIR)* [4] is used to update the particles. The procedure is as follows:

1. Generate an updated particle set by sampling from the proposal distribution (assumed to be Gaussian here), $\mathbf{O}_t^{(l)} \sim p(\mathbf{O}_t^{(l)}|\mathbf{O}_{0:t-1}^{(l)}) = \mathcal{N}(\mathbf{O}_t^{(l)}; \mathbf{O}_{0:t-1}^{(l)}, \Psi)$, where Ψ is the covariance matrix of the state variables. I.e., state dynamics is modeled using a Brownian motion.

2. Re-weight each particle l according to the discriminative classifier: $\pi_t^{(l)} \propto \pi_{t-1}^{(l)} p(y|x(\mathbf{O}_t^{(l)}))$, and normalize so that $\sum_l \pi_t^{(l)} = 1$.

3. Re-sample "P" particles from current particle set according to probabilities π_t. Set $\pi_t^{(l)} = 1/P$ for $l = 1, \ldots, P$.

When updating $\mathbf{H}$ with the current particle set, N_p image patches are sampled from the current frame t, and put into a positive bag X_i. For negative instances, N_n patches are randomly sampled from the region outside of the particle set. Each negative example is put into its own negative bag since typically there is no ambiguity within the negative examples.

2.1 Online Multiple Instance Boosting

Similar to Viola et al. [7, 10], the "AnyBoost" framework [8] is used to train the strong classifier $\mathbf{H}$ to maximize the log-likelihood of bags,

$$\begin{aligned} \mathcal{L}(\mathbf{H}) &= \log(\Pi_i p_i^{y_i}(1-p_i)^{1-y_i}) \\ &= \sum_i [y_i \log(p_i) + (1-y_i)\log(1-p_i)] \end{aligned} \quad (2)$$

where $p_i = p(y_i = 1|X_i)$ is the probability that the bag X_i is positive. Here we adopt the following model

$$p_i = 1 - \Pi_{j=1}^{N_i}(1 - p_{ij})^{1/N_i}, \quad (3)$$

where the instance probability $p_{ij} = p(y_{ij}|x_{ij})$ is given by (1). Compared with the Noisy-OR (NOR) model in [10, 2], $p_i = 1 - \Pi_{j=1}^{N_i}(1 - p_{ij})$, the geometric mean in (3) avoids numerical issues when the number of instances in the bag X_i, N_i, gets too large. "Anyboost" framework [8] is considered to be a gradient descent/ascent problem in the functional space where the weight of each instance is given as the partial derivative of the likelihood function, i.e.,

$$w_{ij} = \frac{\partial \mathcal{L}(\mathbf{H})}{\partial \mathbf{H}} \approx \frac{1}{N_i} \frac{y_i - p_i}{p_i} p_{ij}. \quad (4)$$

The weak classifiers are selected during the boosting stage by maximizing the following energy function $\phi(h)$. It turns out that maximizing the energy function is equivalent to taking a direction that has the steepest ascent in the functional space.

$$\phi(h_k) = \sum_{ij} h_k(x_{ij}) w_{ij}. \quad (5)$$

2.2 Contributions on the Learning Algorithm

Algorithm 1 illustrates the proposed online MILBoost method. The proposed algorithm is novel in the following aspects,

- At each step of boosting, the energy function (5) is maximized and the weight α_k is not absorbed into the h_k. α_k is helpful in discarding the non-contributing weak classifiers. Since maximizing (5) does not require the log-likelihood (2) to be evaluated, the proposed boosting process is more efficient than a brute force search proposed in [2].

- Since (5) is computed over current examples, information from the previous frames is captured by making use of the "T" best weak classifiers from the previous frame's strong classifier. Another way of doing this would be to model the haar features with Gaussian distribution, and update the distribution parameters with the current examples [2]. However, with this type of distribution based weak classifiers, it is hard to balance information from current frame with previous frames that might eventually cause tracker's failure.

*Depends on the application, additional parameters, such as rotation angle, can also be easily added without much changes to the proposed method.

- In addition to Haar-like feature, the proposed culture color histogram based weak classifier provides an efficient way to tackle color drifts under different lighting conditions (see Section 2.3).

Algorithm 1: Proposed Online MILBoost

Input: Training bags $\{X_i, y_i\}$. Each bag X_i contains a set of training instances $\{x_{i1}, \ldots, x_{iN_i}\}$ and, bag labels $\{y_i\} \in \{0, 1\}$.

Output: Updated discriminative classifier $\mathbf{H}$

1: Train a set of Haar-like feature based weak classifiers $\{h_m\}_{m=1}^{M}$, and culture color histogram based weak classifier h_{M+1} with the current set of training samples.
2: Set $\mathbf{H}_0 = 0$.
3: Select T best performing weak classifiers from the previous frame's strong classifier.
4: **for** $k = T + 1$ to K **do**
5: With $\mathbf{H}_{k-1}$, update p_{ij}, p_i from (1).
6: Update weights $w_{ij} = \frac{1}{N_i} \frac{p_i - y_i}{p_i} p_{ij}$.
7: Find h_k that maximizes the energy ϕ in (5).
8: Find α_k with line search by maximizing the log-likelihood $\mathcal{L}(\mathbf{H}_{k-1} + \alpha_k h_k)$.
9: $\mathbf{H}_k = \mathbf{H}_{k-1} + \alpha_k h_k$.
10: Terminate if the contribution of weak classifier h_k falls below a threshold δ i.e. $\alpha_k < \delta$
11: **end for**

Note: The T best weak classifiers can be selected using the above algorithm with "for-loop" index k running from 1 to T.

2.3 Weak Classifiers and Image Features

For each image patch, two kinds of features are computed: a vector of M dimensional Haar-like features f^H [9], and an 11-dimensional culture color histogram f^C which is a coarse quantization of the color space into 11 bins [11].

Haar-like features: For each f_m^H, a weak classifier h_m, which is a linear perceptron with a simple threshold [9], is used with a polarity $\mathcal{P} \in \{1, -1\}$, i.e.

$$h_m(x) = \begin{cases} +1 & \text{if } \mathcal{P} f_m^H(x) < \mathcal{P}\theta \\ -1 & \text{otherwise} f \end{cases} \quad (6)$$

This binary weak classifier is much simpler compared to the log odds ratio method in [2], which fixes scalar weight α_m.

Culture Color Histogram: For an image patch x, $\mathbf{f}_x^C$ is the culture color histogram based feature vector [11].

A weak classifier h_{M+1} is defined as

$$h_{M+1}(x) = \begin{cases} +1 & \text{if } EMD(\mathbf{f}_{mean}^C, \mathbf{f}_x^C) < 0.5 * D_{mean}^- \\ -1 & \text{otherwise} \end{cases} \quad (7)$$

where EMD is the Earth mover's distance [6], $\mathbf{f}_{mean}^C$ is the mean culture color histogram of the positive samples and D_{mean}^- is the mean of the Earth mover's distances of culture color histograms of all the negative samples to $\mathbf{f}_{mean}^C$.

3 Experimental Results

To demonstrate the effectiveness of the proposed method, it is tested with our own dataset[†] with some complex scenarios. In particular, we show how the proposed algorithm deal with the complex shapes and appearance changes in humans, and the unexpected illumination variations (e.g. shiny floor surface, shadows, sudden lighting changes caused by door-opening and closing etc.).

Figure 1. Sample tracked results with the proposed algorithm. Left column: positive examples generated based on particle distribution. Right column: corresponding negative examples generated.

For all the experiments, $K = 50$ weak classifiers are chosen for boosting and Haar-like feature dimension M is set to 250. The maximum number of positive training examples, N_p, is set to 45. The maximum number

[†]Sample datasets are available at http://nanonet.ece.ucsb.edu/HFH

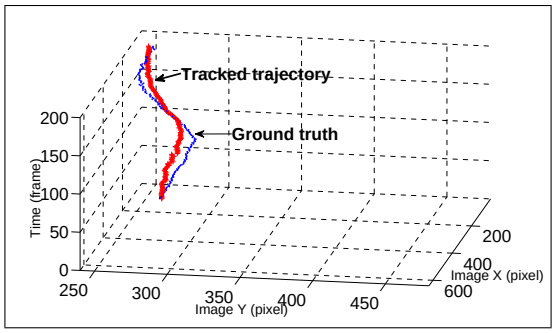

Figure 2. Comparison of tracked trajectory with ground truth.

of negative image patches, N_n, is set to 50. Figure 1 illustrates how the particle filters behaves and helps to generate training examples for MIL learning. Figure 2 compares the tracked trajectory (the mean of the particle distribution) with the manually marked ground truth.

Under different illuminations, one culture color could appear differently on the image (e.g., a red color might appear black on the image under low lighting conditions). In this scenario, simple classifier in (7) might cause problem. However, with the proposed algorithm, the different color appearance at different time can be learned and incorporated into the classifier $\mathbf{H}$. This is demonstrated in Figure 3. At frame 200, $\mathbf{H}$ contains three color-based weak classifier (7) corresponds to the h_{M+1} trained at frame 1, 61, and 118 respectively. This is because at these three instances, the object's culture color appears different. This particular color drifting pattern matches the results presented in [11].

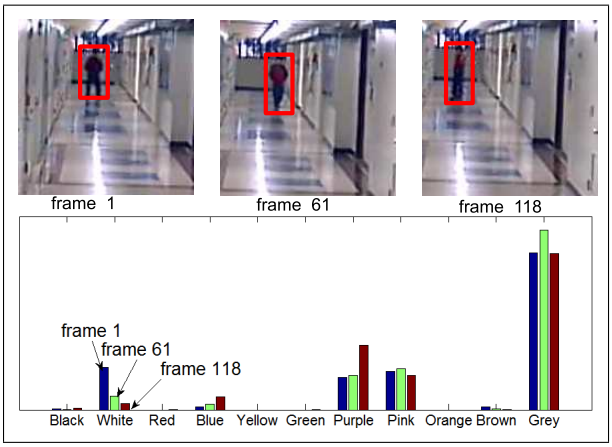

Figure 3. Culture color drifts detected by the proposed learning algorithm (best viewed in color).

4 Conclusion

In this paper, we have presented an efficient online multiple instance boosting algorithm to adaptively model object appearance for object tracking. Online multiple instance learning is effectively coupled with particle filtering, i.e., training samples are generated from the particle distribution and particle weights are updated based on learned appearance model. It gives the flexibility to define the most likely positive instance(s) while learning the classifier. Experiment results has demonstrated the robustness of the proposed algorithm for challenging scenarios. For future work, it would be interesting to try the proposed algorithm for multiple objects tracking where we need to associate different objects using the learned appearance models and particle filters.

References

[1] S. Avidan. Ensemble tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 494–501, 2005.

[2] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 983–990, 2009.

[3] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Robust tracking-by-detection using a detector confidence particle filter. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1515–1522, 2009.

[4] A. Doucet, N. D. Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[5] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via online boosting. In *British Machine Vision Conference (BMVC)*, pages 47–56, 2006.

[6] E. Levina and P. Bickel. The earth movers distance is the mallows distance: Some insights from statistics. In *IEEE International Conference on Computer Vision (ICCV)*, pages 251–256, 2001.

[7] Z. Lin, G. Hua, and L. S. Davis. Multiple instance feature for robust part-based object detection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 405–412, 2009.

[8] L. Mason, J. Baxter, P. Bartlett, and M. Frean. Boosting algorithms as gradient descent. In *Advances in Neural Information Processing Systems (NIPS)*, pages 512–518, 2000.

[9] P. Viola and M. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[10] P. Viola, J. Platt, and C. Zhang. Multiple instance boosting for object detection. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1417–1426, 2005.

[11] G. Wu, A. Rahimi, E. Y. Chang, K. Goh, T. Tsai, A. Jain, and Y.-F. Wang. Identifying color in motion in video sensors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 561–569, 2006.