FACE-TLD: TRACKING-LEARNING-DETECTION APPLIED TO FACES

Zdenek Kalal [†], *Krystian Mikolajczyk* [†], *Jiri Matas* [‡]

[†] Centre for Vision, Speech and Signal Processing, University of Surrey, UK
[‡] Center for Machine Perception, Czech Technical University, Czech Republic

ABSTRACT

A novel system for long-term tracking of a human face in unconstrained videos is built on Tracking-Learning-Detection (TLD) approach. The system extends TLD with the concept of a generic detector and a validator which is designed for real-time face tracking resistent to occlusions and appearance changes. The off-line trained detector localizes frontal faces and the online trained validator decides which faces correspond to the tracked subject. Several strategies for building the validator during tracking are quantitatively evaluated. The system is validated on a sitcom episode (23 min.) and a surveillance (8 min.) video. In both cases the system detects-tracks the face and automatically learns a multi-view model from a single frontal example and an unlabeled video.

Index Terms— long-term face tracking, learning, detection, verification, real-time

1. INTRODUCTION

Long-term real-time tracking of human faces in unconstrained environments is a challenging problem: given a single example of a specific face, track the face in a video that may include frame cuts, sudden appearance changes, long-lasting occlusions etc. In such environments, the frame-by-frame tracking meets face detection and verification at one point with a common goal to determine the location of the specific face. This paper proposes a novel solution that is suitable in such situations.

Two approaches are used for modeling an object appearance in tracking: static and adaptive. Static models [1] assume that the object appearance change is limited and known. Unexpected changes of the object appearance can not be tracked. This drawback is addressed by adaptive methods [2] which update the object model during tracking. The underlying assumption is that every update is correct. Every incorrect update brings error to the model that accumulates over time and causes drift. In the context of faces, 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 face was in the field of view. In scenarios where a face moves in and out of the frame, face re-detection is essential. Face detection have been extensively studied [4] and a range of ready-to-use face detectors are available [5] which enable tracking-by-detection. Apart from expensive offline training, the disadvantage of tracking-by-detection is that all faces have the same model and therefore the identities can not be distinguished. To elevate this problem, Li et al. [6] proposed a face tracking algorithm that splits the face model into three parts with different lifespan. 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 offline processing and adaptation for real-time tracking is not straightforward.

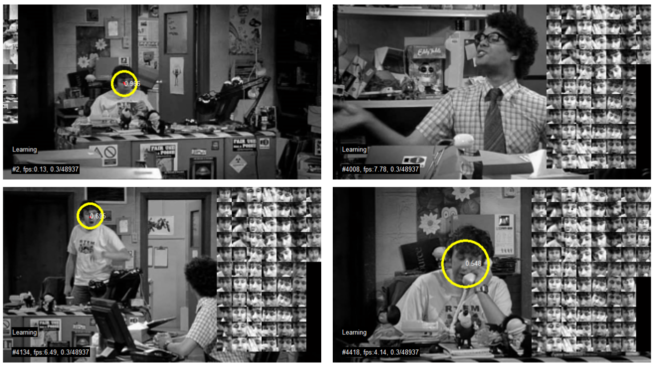
Learning
#2, fps:0.13, 0.3/48937
Learning
#4008, fps:7.78, 0.3/48937
Learning
#4134, fps:6.49, 0.3/48937
Learning
#4418, fps:4.14, 0.3/48937

Fig. 1. Our system tracks, learns and detects a specific face in real-time in unconstrained videos.

In this work, we build on an approach called Tracking-Learning-Detection (TLD) [8], whose learning part was analyzed in [9]. The TLD method was designed for long-term tracking of arbitrary objects in unconstrained environments. 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. This paper has three contributions w.r.t. TLD: (i) Additional source of information (offline detector) is embedded to the TLD framework which simplifies the learning task in cases when the ob-

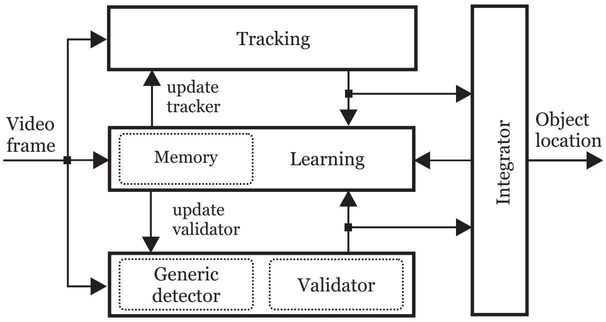

Fig. 2. The proposed face tracking system based on TLD extended with generic detector and online trained validator.

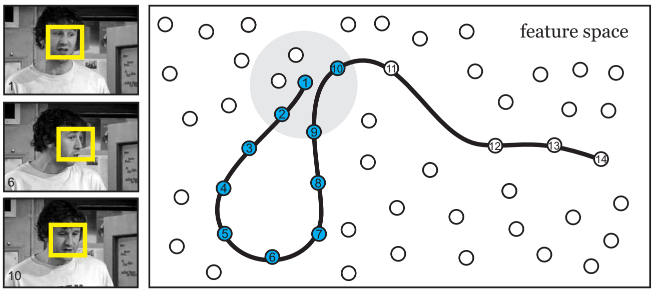

Fig. 3. The face is modeled by a collection of patterns discovered by a learning strategy. The Loop accepts the points 1-10, Absolute: 1,2,9,10 and Change: 1-11.

ject is ”roughly” known a priori. (ii) The system was adapted to perform real-time specific face tracking robust to viewpoint change, which we consider a significant contribution. (iii) Three learning strategies in context of faces are quantitatively evaluated.

2. FACE-TLD

The long-term tracking problem addressed by TLD can be constrained when applied to faces. In the original formulation, the entire detector was learned online, starting from a single frame. An efficient classifier (randomized forest) was used to represent the decision boundary between the object and its background. In the case of face tracking, building the entire detector is not necessary since a range of face detectors is readily available. The learning therefore consists of building a validator that decides wether a face patch corresponds to the target or not. The validation is significantly less time demanding than the face detection since only a fraction of candidates have to be verified. On the other hand its precision has to be high in order to avoid confusing two different identities. Face recognition [10] is in general very challenging and a large number of sophisticated methods have been designed already. Here we show that a very simple validator and a learning method works very well when tracking a face in cluttered background.

We adapted a frontal face detector from [5] which demonstrated state-of-the-art performance and runs at 20 fps. On the top of the detector we incorporated a validator, a module that analyzes a face patch and outputs a confidence that the patch corresponds to the specific face. The validator is realized by a collection of examples. This collection is initialized by a single example in one frame and it is extended during tracking by inserting more examples.

The block diagram of the proposed TLD Face tracker is shown in Fig. 2. Every frame is processed by a tracker and a detector and their outputs are passed to an integrator which estimates the object location. The object location as well as the detections and the trajectory are analyzed by a learning block which generates training data for the detector (update of the tracker is not considered in this implementation). In contrast to the original TLD, the detector is split into two parts: (i) generic detector, and (ii) validator. The generic detector returns all patches that correspond to certain visual class (e.g. face), validator decides if the object represents a specific instance selected for tracking.

We adapted the tracker from [8], i.e. the tracked patch is described by a set of local overlapping sub-patches which displacements are estimated by Lucas-Kanade [11], the global motion is defined as median over the local displacements. This local approach is resistent to partial occlusions. The validator consists of a collection of positive and negative patches where each patch is described by a pattern $\mathbf{x} = [x^1, x^2, \ldots, x^K]$, where x^i is a single 2 bit Binary Pattern [8](2bitBP) feature measuring quantized gradient orientation within its supporting area on the patch. The features are generated at random locations within the patch in advance of tracking. The reason for choosing 2bitBP is that these features can be efficiently evaluated at multiple scales using integral images which are already pre-computed for the face detector. Further more, they are invariant to (local) illumination changes.

The validator stores all positive and negative patterns that have been collected during tracking $X = \{\mathbf{x}_i\}$. The similarity of pattern $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as $S(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{K}\sum_{i=1}^{K}([x_i^k = x_j^k])$ where [.] is a logical operator that outputs 1 or 0. The confidence of a query pattern $\mathbf{x}_q$ is defined as a confidence of a 1-NN classifier. The learning strategies for collecting the patterns will be will be quantitatively analyzed in the next section.

3. MODELING THE FACE APPEARANCE

This section addresses the following problem: given a single example of a face in one frame, discover all its appearances in the video volume. The situation is illustrated in Fig. 3 where the appearance of the face patches are projected to the feature space. The space of all patches extractable from the video is huge and within this space, there is a single point representing the face selected for tracking. A naive approach to selecting related patterns would be to take all patches that are ”similar” to the initial instance in the feature space. The problem is that the appearance variability of a specific face is often larger than the variability between different identities. Therefore, if two instances are far apart from each other it does not necessary mean that they do not belong to the same identity. In order to solve for these cases, some form of additional information is necessary. In our case we have two sources of additional information: tracking and face detection. We first reduce the space of all possible patches to those that have been discovered by tracking. Such constrained problem can be reformulated as follows: given a single example detected by a detector, track it by a tracker and accept all patterns along the trajectory that are likely to correspond to the same identity. Patches surrounding a validated trajectory are considered as negative.

In the following, three strategies for selecting reliable patterns along the trajectory will be introduced. All of them are based on a similarity between patterns. Patterns $\mathbf{x}_i$ and $\mathbf{x}_j$ will be called *similar* if $S(\mathbf{x}_i, \mathbf{x}_j) > \theta$, where θ is a similarity threshold. ABSOLUTE: accepts all patterns on the trajectory that are similar to the initial pattern $\mathbf{x}_1$. This approach defines a sphere around the initial example in the feature space and accepts all patterns on the trajectory that pass through it (points 1,2,9,10 in Fig. 3). CHANGE: accepts the trajectory up to the pattern that is similar to the previous pattern (points 1-11). LOOP: accepts the trajectory that ends with patterns similar to the initial pattern $\mathbf{x}_1$ (points 1-10).

4. EXPERIMENTS

4.1. Comparison of learning strategies

The first experiment quantitatively evaluates the performance of the learning strategies in terms of the ability to identify patch depicting the given face. The quality is assessed using the precision and recall measures. Precision is the number of correctly accepted patches divided by the total number of patches accepted. Recall is the number of correctly accepted patches divided by the number of patches that should be accepted according to the ground truth.

The evaluation was performed on sitcom IT Crowd (first series, first episode). The episode lasts for 23 minutes which corresponds to 35 471 frames. First, the ground truth trajectories of four main characters in the footage were manually annotated. The trajectories start when a face becomes visible and end when it disappears (multi-view pose was considered). Second, the tracker produced another set of trajectories. At the beginning of each ground truth trajectory the tracker was initialized and tracked the face up to the end of the shot. This resulted in 1,180 tracks of average length 32 frames. Not all trajectories thus obtained were entirely correct. By comparing the tracked trajectories with ground truth in terms of bounding box overlap the correct parts of the trajectories were identified (overlap $> 70\%$). In an ideal case, these correct parts should be identified by the learning strategies.

Character	ABSOLUTE		CHANGE		LOOP	
	Precision	Recall	Precision	Recall	Precision	Recall
Roy	0.98	0.23	0.96	0.61	0.99	0.41
Moss	0.98	0.24	0.89	0.62	0.98	0.40
Jen	0.98	0.35	0.97	0.69	0.99	0.60
Denholm	0.91	0.24	0.93	0.56	0.99	0.38
Mean	0.96	0.26	0.93	0.62	0.99	0.45

Table 1. Precision and recall of different learning strategies.

All the strategies with identical threshold ($\theta = 0.8$) on the distance were applied to the trajectories. Each trajectory was analyzed independently, the distance was measured with respect to the initial pattern on the particular trajectory. Table 5 shows the resulting precision/recall for the main characters separately as well as the averaged performance. The ABSOLUTE strategy accepts only the patterns within a sphere of diameter $1 - \theta$, and therefore its average recall is the lowest of all (26%). The CHANGE strategy enables growing outside the ball which increases the recall up to 62% but this is at the cost of decreased precision to 93%. Typical errors happen when the person rotates out-of-plane or slowly becomes occluded. The LOOP strategy identifies such situations assuming that drifted tracker is unlikely to recover. If the similarity drops and then increases again, it is likely that tracker has been following the target correctly and the drop in similarity was caused by changed appearance. This strategy has thus the best precision of 99% and recall of 45%, which is significantly improved w.r.t.ABSOLUTE. The recall is not as high as for the CHANGE strategy, since not all correctly tracked faces return back to the vicinity of the initial pattern. For the purpose of correct model update, the strategies with high precision are preferred.

4.2. Sitcom episode

The experiment compares the standard TLD [9] with Face-TLD on the same episode as in previous experiment. Both systems were initialized on a face of one character (Roy) at his first appearance. The subject appears in 12 222 frames, the entire episode contains 35 471 frames. The TLD correctly tracked/detected at the beginning of the episode, but failed to detect the character in the second half. The overall recall was of 37% and precision of 70%. The Face-TLD was able to re-detect the target throughout the entire episode leading to recall of 54% and precision of 75%. The introduction of face detector increased the recall by 17%. Both approaches processed the episode at frame-rate on a laptop.

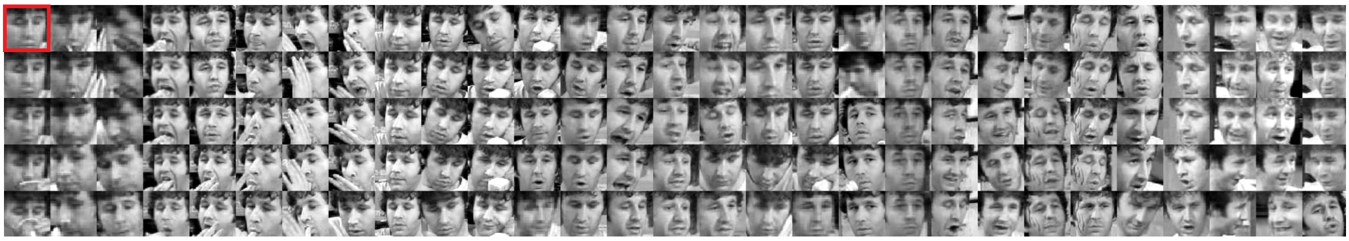

Fig. 4. Using a single example (RED) the proposed system learned a multi-view appearance by a single pass through a video.

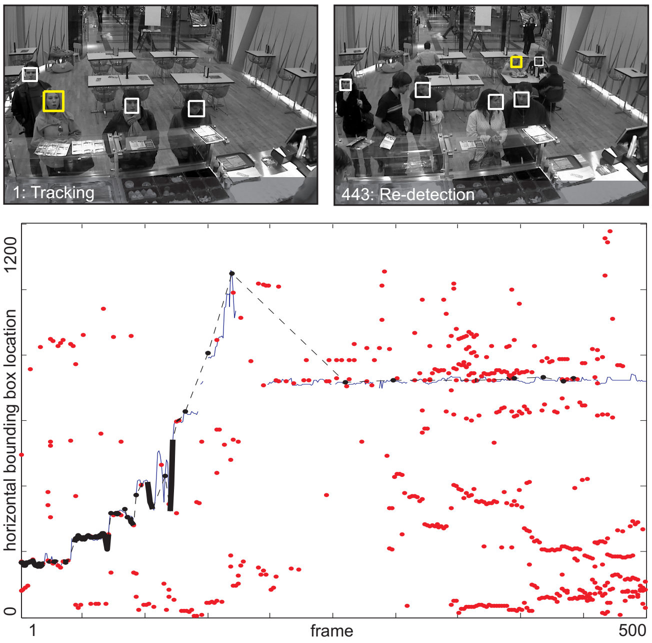

Fig. 5. TOP: The surveillance scenario with multiple similar objects. BOTTOM: Detections through which the Face-TLD tracker finds a path. Detections (red), validated detections (black), ground truth (blue).

4.3. Surveillance footage

We applied Face-TLD to images from CCTV monitoring a shop. The images are accessible in real time at 1 fps (Fig. 5). The objective is to count the number of customers during the day. Simple counting of detections is not appropriate since the same person could be counted many times. Tracking techniques are difficult to apply because of the low frame rate and the fact that the subject may leave the camera view and come back later. The testing sequence contains 500 frames in which we manually annotated a track of one subject. The Face-TLD was again compared to standard TLD. The TLD achieved recall of 12% and precision of 57%, the Face-TLD achieved recall of 35% and precision of 79%. The introduction of face detector increased the recall by 23%. Fig. 5 (bottom) illustrates the recovered path of the subject through the cloud of detections. This experiment demonstrates, that Face-TLD is robust to low-frame rate and does not confuse different faces.

5. CONCLUSIONS

Face-TLD, a system for tracking of human faces based on Tracking-Learning-Detection (TLD), is proposed. Face-TLD combines a priori information about the object class with the information from the video. Namely, a generic face detector and a validator were integrated into the TLD framework. Several learning strategies were quantitatively evaluated in the context of face tracking and identification. The Face-TLD system was tested on two sequences: a sitcom episode (23 min.) and a surveillance video (8 min.). A particular face model was automatically learned from a single example and the video. The system is suitable for real-time tracking/detection (e.g. on mobile devices). A demo application and the test data with ground truth are available online[1].

Acknowledgment: This research was supported by Czech Science Foundation Project P103/10/1585, EC project ICT-215078 DIPLECS, UK EPSRC EP/F003420/1 grant and the BBC R&D.

6. REFERENCES

[1] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," *IJCV*, vol. 29, no. 1, pp. 5–28, 1998.

[2] J. Lim, D. Ross, R.S. Lin, and M.H. Yang, "Incremental learning for visual tracking," *NIPS*, 2005.

[3] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley, "Face tracking and recognition with visual constraints in real-world videos," *CVPR*, 2008.

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *CVPR*, 2001.

[5] Z. Kalal, J. Matas, and K. Mikolajczyk, "Weighted sampling for large-scale boosting," *BMVC*, 2008.

[6] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low frame rate video: A cascade particle filter with discriminative observers of different lifespans," *CVPR*, 2007.

[7] M. Everingham, J. Sivic, and A. Zisserman, "Taking the bite out of automated naming of characters in TV video," *IVC*, vol. 27, pp. 545–559, 2009.

[8] Z. Kalal, J. Matas, and K. Mikolajczyk, "Online learning of robust object detectors during unstable tracking," *OLCV*, 2009.

[9] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints," *CVPR*, 2010.

[10] W. Zhao, R. Chellappa, PJ Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *CSUR*, vol. 35, no. 4, pp. 399–458, 2003.

[11] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *IJCAI*, vol. 81, pp. 674–679, 1981.

[1] http://cmp.felk.cvut.cz/tld/