CLEANING UP AFTER A FACE TRACKER: FALSE POSITIVE REMOVAL

Makarand Tapaswi[1], *Cemal Çağrı Çörez*[2], *Martin Bäuml*[1], *Hazım Kemal Ekenel*[2], *Rainer Stiefelhagen*[1]

[1]Karlsruhe Institute of Technology, Karlsruhe, Germany
[2]Istanbul Technical University, Istanbul, Turkey

ABSTRACT

Automatic person identification in TV series has gained popularity over the years. While most of the works rely on using face-based recognition, errors during tracking such as false positive face tracks are typically ignored. We propose a variety of methods to remove false positive face tracks and categorize the methods into *confidence-* and *context-based*. We evaluate our methods on a large TV series data set and show that up to 75% of the false positive face tracks are removed at the cost of 3.6% true positive tracks. We further show that the proposed method is general and applicable to other detectors or trackers.

Index Terms— false positive removal, face tracking, video processing, TV series

1. INTRODUCTION

Person identification in TV series is a popular task in video analysis and computer vision [1, 2, 3, 4, 5, 6]. A majority of the research in this area deals with face-based person identification, involving face detection [7, 8], tracking [9, 10] and subsequent recognition via supervised [1, 2] or semi-supervised [4, 6] learning schemes. To localize all faces in a video, the face detection/tracking methods are typically tuned for a high recall. This, however, leads to a decrease in precision, or an increased number of false positive tracks.

One possible solution to deal with the lowered precision is to manually discard *false positive face tracks* (FPFT). Although the FPFT (see Fig. 1) are typically less than 10-15% of all tracks, this step hinders complete automation. Towards the goal of an automatic and improved workflow, we propose multiple post-processing strategies to detect FPFT. We desire to improve precision by removing false positive tracks, while at the same time minimizing the reduction in recall by not discarding true positive tracks.

The major contribution of this paper is to propose a set of cues both generic and domain-specific to tackle the problem of false positive face track detection. We evaluate our methods on 11400 tracks from two diverse TV series and show promising results. We also present a short evaluation on face tracks obtained from different detection and tracking schemes.

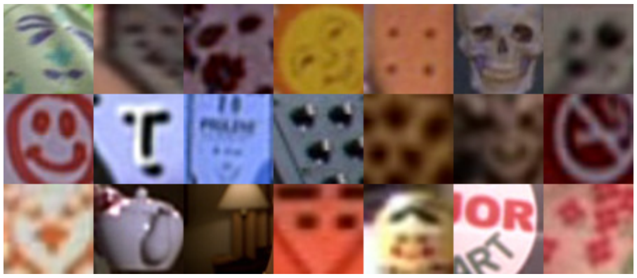


Fig. 1: Examples of false positive face tracks.

The paper is organized as follows. We first present some related work in Sec. 1.1. Sec. 2 details the five types of cues we use to detect FPFTs. Sec. 3 presents the evaluation and finally Sec. 4 concludes the paper.

1.1. Related Work

Object detection, in general, suffers from the problem of false positives. For the special case of person detection, spatial context is quite popular and ground plane estimates are used extensively to either speed up detection or remove false positives [11, 12]. We use spatial context to determine the expected location of a face without requiring knowledge about the ground plane in Sec. 2.2.

In some previous work, a simple time-based measure such as a minimum track length has been used to detect false positives and to adapt person detectors on-the-fly [13, 14]. There is also some work on removing false positive faces from images. Arandjelovic *et al.* [15] use skin color classification based on RGB Gaussian models [16] to find false positive faces. More recently Li and Chen [17] train classifiers based on Canny edge detection to remove false positive detections. Atanasoaei *et al.* [18] use the idea that a detector fires multiple times in the vicinity of a true detection, but a false positive does not exhibit this phenomenon. They generate subwindows around the target detection and run the face detector [8] to obtain scores which form a part of their feature.

In comparison to above methods, we propose context-based cues that analyze multiple tracks at once. While object detection in images might bring in many false positives, tracking in video data is a first step to reduce them [19]. Our face tracks are already obtained through a first-level filtering, which makes the data much harder to begin with.

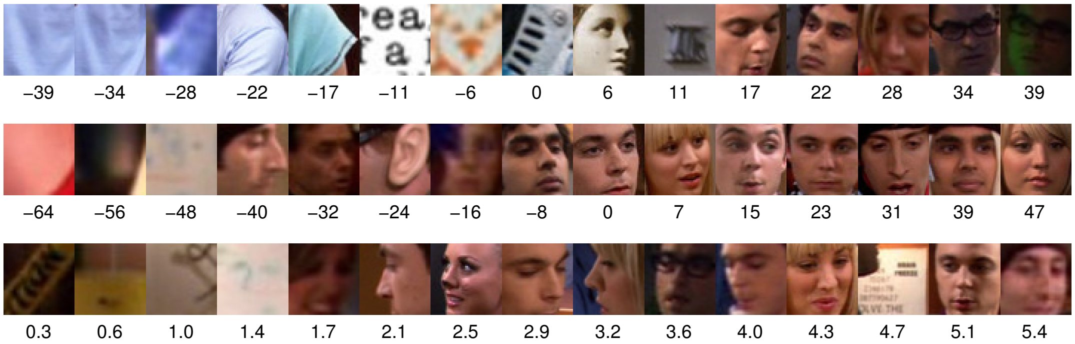


Fig. 2: Examples of tracks from the BBT dataset and their scores for three different cues. Top: Skin color log-likelihood. Mid: Facial feature point localization confidence. Bottom: Animation score.

2. FALSE POSITIVE CUES

We use a variety of cues to determine whether a track is a false positive. The cues are categorized into two major types. (i) *confidence based*: the classification of a track as a false positive depends solely on itself and its inherent properties. The application areas for these cues are thus more widespread. (ii) *context based*: a model typically learned by a combination of multiple tracks and with domain-specific information (in our case, cinematography) is used to determine whether a track is a false positive.

2.1. Confidence based cues

Skin color Detecting skin color is among the simplest methods for classifying whether a detected face is a true positive [15]. In fact, early face detection schemes use skin detection and blob processing to find faces [20, 21]. We use Gaussian RGB models μ_S and μ_{NS} for skin and non-skin respectively by Jones and Rehg [16] and analyze $T = 5$ faces for each track, computing their average log-likelihood of skin vs. non-skin.

$$\alpha_{skin} = \frac{1}{T \cdot MN} \sum_{f=1}^{T} \sum_{x \in \mathcal{P}_f} \ln \frac{p_x(x|\mu_S)}{p_x(x|\mu_{NS})}, \qquad (1)$$

where $\mathcal{P}_f$ is the set of all pixels for the face f of size $M \times N$. Fig. 2 (top) shows sample faces and false positives with their corresponding skin color log-likelihood scores.

Facial feature point localization A popular topic in computer vision, facial features improve face recognition by providing part-based features [1] or better alignment [22]. While methods which use regression [23] do not provide an easy way to determine localization confidence, others which use discriminative models such as [1] can provide one.

We use the nine point detector [1] and extract the points and corresponding confidence scores for each face. The track score α_{fp} is the average of confidences for all faces in the track. Fig. 2 (mid) shows the variation in confidence as the images span false positives to profile faces and finally to frontal faces on which localization works best.

Animation Faces are a highly animate set of objects, while false positives, especially on the background, do not change. We measure the amount of animation, *i.e.* internal visual movements or deformations in the face track as a cue. To improve speed, we analyze 5 face candidates (equally spaced in time) from every track. For each face candidate image f_t at time t, we find the best matching correspondence f^*_{t+1} in the next frame $t + 1$. A scanning window search in the vicinity of the face detection is used to find the region with highest similarity to the face f_t. The animation score for the face is

$$\alpha^t_{anim} = \frac{1}{MN} \min_{f_{t+1}} \|f_t - f^*_{t+1}\|, \qquad (2)$$

where f_t has a size $M \times N$. The final score for each track is obtained by averaging the scores for the 5 frames. Fig. 2 (bottom) shows that false positives tend to be less animate and thus have a lower animation score.

2.2. Context based cues

In this section, we discuss supervised methods, which model contextual cues from multiple tracks.

Facial location map In the making of videos, cinematography rules heavily influence the placement of faces in the frame. For example, shots with a single person (a 1 shot) typically center the person in the frame; or two people (a 2 shot) are often placed at the rule of thirds. In this cue, we model the probability of seeing a face at any given location in the frame. We build different models depending on the number of visible people n. For scenarios with $n \geq 4$, we lump them together into a *group shot*.

The models are encoded as heat maps indicating likelihood of seeing a face across the frame. These maps $\mathcal{M}_n$ are computed using the locations of set of true-positive tracks $\mathcal{S}_n$. For example, tracks that appear alone in the frame constitute set $\mathcal{S}_1$, and two tracks that appear at the same time in the frame form $\mathcal{S}_2$.

$$\mathcal{M}_n = \sum_{t \in \mathcal{S}_n} \mathbf{I}(t_x, t_y, t_w, t_h), \qquad (3)$$

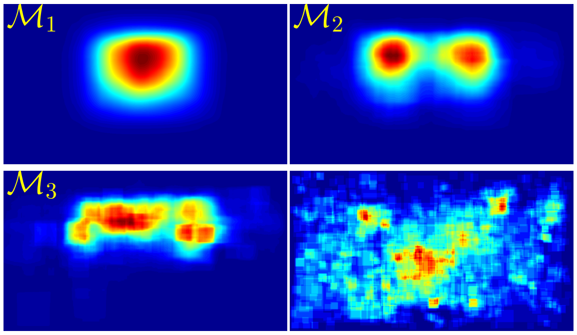


Fig. 3: Heat maps for single face in the frame $\mathcal{M}_1$ (top-left), two faces $\mathcal{M}_2$ (top-right), three faces $\mathcal{M}_3$ (bottom-left) and typical locations of false positive tracks (bottom-right). (Best viewed in color.)

where $\mathbf{I}(x, y, w, h)$ is an all-zero matrix of the size of the frame with a 1 in the areas corresponding to the bounding box specified by $[x, y, w, h]$.

Fig. 3 shows examples of such maps for different n. Note how the faces tend to appear slightly above the vertical center as shots are dominated by head and torso. Also note the locations of the false positive tracks Fig. 3 (bottom-right) which are spread all around the frame.

At test time, if a given frame contains n tracks, we do not know how many of them are true positive. In order to choose the correct model, we evaluate all possible options, *i.e.* all n tracks are FPFT, $n-1$ tracks are FPFT, and so on, up to all tracks are true positive. We score each frame and track based on the amount of overlap with the maps and pick the best fitting (highest scoring) map for each frame. Finally, the track score is

$$\alpha_{map} = \frac{1}{T} \sum_{f=1}^{T} \max_{n=[1..4]} \left[\mathcal{M}_n \cap \mathbf{I}(f_x, f_y, f_w, f_h) \right], \qquad (4)$$

where T is the number of frames in the track and $\mathbf{I}(.)$ is the location mask for face f.

Relative size We observe that most characters (at least the important ones) appear at similar depths in the frame. Thus, if we see a face of 150px along with another of 25px, we can infer that the small face is most likely a false positive.

For frames containing more than one track, we compute α_{size}, a ratio of the face width with respect to other faces in the frame. We simplify the scenario when 2 or more tracks appear simultaneously by grouping faces which appear at similar depth. Fig. 4(left) shows an example where a false positive track (red box) occurs with a true positive with a large difference in relative size. On the other hand, Fig. 4(right) shows a case when we misclassify a true positive track as the character just enters the room thus exhibiting a skewed ratio.

2.3. Fusion

We first normalize the individual confidences or scores via min-max normalization to the $[0-1]$ range. For each track, we form a feature vector $x = [\alpha_{skin}, \alpha_{fp}, \alpha_{anim}, \alpha_{map}, \alpha_{size}]$ and train logistic regression (LR) models to obtain a final classification score. The learned weights indicate that skin color and face location cues are dominant while the other methods follow closely.

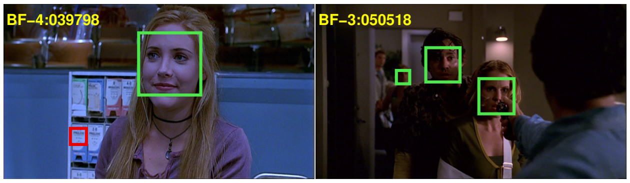


Fig. 4: Relative size of faces in a frame as a false positive detection cue. Left: correctly recognized FPFT (ratio = 0.26); Right: wrongly classified true positive track (ratio = 0.38). (Best viewed in color.)

3. EVALUATION

3.1. Evaluation Setup

Data set We use the face tracks – obtained by a particle filter used in conjunction with multi-pose MCT-based [8] detectors – made publicly available by [6]. The data set includes two diverse TV series: (i) BBT: The Big Bang Theory (season 1, episodes 1–6, ∼20min) is a sitcom with a small cast. The scenes are mostly indoors and take place in well lit conditions. (ii) BF: Buffy the Vampire Slayer (season 5, episodes 1–6, ∼40min) is a fantasy TV series and has a cast size of 15-20 people. The scenes are an equal mix of indoor and outdoor action and have widely varying illumination.

Evaluation criteria As motivated in the introduction, our goal is to detect and remove as many false positives as possible while keeping the removal of true positive tracks to a minimum. Thus, in Table 1 and 2 we use them as our evaluation criteria: number of correctly classified false positive tracks (higher is better) / number of true positive tracks misclassified as false positive (lower is better), *e.g.* $30/6$ would mean that 30 FPFT are correctly detected and 6 true positive tracks are misclassified as false positive.

Cross validation In situations which require training models (facial location map, logistic regression) or learning thresholds, we perform leave-one-out cross-validation, *i.e.* use the five other episodes from the same series as training data.

3.2. False Positive Removal Accuracy

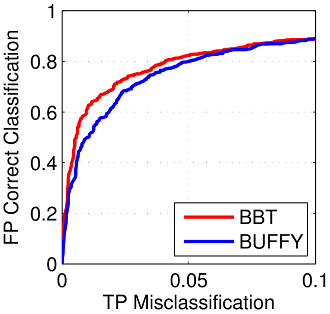
Table 1 presents false positive removal results. In the first two rows, we present the number of face tracks (#Tracks) and number of false positive face tracks (#FPFT). The following rows (3 to 7) evaluate performance of individual cues. We classify the track as a false positive if $\alpha_{method} < \theta_{method}$ for all methods. Note that while some methods perform worse than others (most notably animation), they are useful as they contribute towards the final fusion step.

	BBT-1	BBT-2	BBT-3	BBT-4	BBT-5	BBT-6	BF-1	BF-2	BF-3	BF-4	BF-5	BF-6	TOTAL
#Tracks	704	712	815	685	728	1040	920	1216	1369	1010	963	1281	11443
#FPFT	79	143	202	101	167	211	119	202	226	128	150	172	1900
Skin confidence	45 / 0	69 / 5	106 / 3	36 / 6	43 / 7	62 / 2	30 / 6	76 / 33	55 / 4	57 / 14	34 / 7	32 / 1	645 / 88
Facial features	5 / 1	17 / 3	25 / 5	22 / 0	17 / 2	41 / 0	23 / 8	46 / 13	22 / 9	27 / 13	18 / 10	34 / 18	297 / 82
Animation	0 / 0	12 / 0	41 / 18	8 / 0	23 / 7	21 / 4	21 / 19	22 / 9	13 / 9	16 / 10	38 / 20	24 / 40	239 / 136
Expected location	31 / 5	70 / 7	100 / 21	52 / 18	59 / 22	96 / 32	35 / 2	34 / 4	27 / 3	43 / 3	31 / 4	59 / 6	637 / 127
Relative size	23 / 0	41 / 0	54 / 1	24 / 3	17 / 2	59 / 0	36 / 5	63 / 16	43 / 5	39 / 2	50 / 9	44 / 5	493 / 48
Combined	59 / 3	114 / 22	170 / 19	74 / 19	137 / 42	136 / 24	95 / 19	171 / 62	131 / 22	112 / 32	113 / 40	132 / 43	1444 / 347
MOTA before	78.98	65.19	62.62	78.02	58.64	65.95	74.56	68.46	64.52	68.65	69.45	68.37	68.62
MOTA after	83.18	69.95	73.59	79.93	72.26	71.98	77.87	72.85	68.33	71.85	73.52	72.52	73.98

Table 1: Statistics and false positive track classification results across different methods and after fusion on the multimedia dataset. The thresholds used for individual methods are: $\theta_{skin} = -45$, $\theta_{fp} = -40$, $\theta_{anim} = 1.1$, $\theta_{eloc} = 5.5$ and $\theta_{relsz} = 0.4$. Rows 3-8 present the number of correctly removed FPFT / number of misclassified true positive tracks.

		BBT-1	BF-2	BF-5
VGG [2]	#Tracks	–	962	760
	#FPFT		151	94
	Combined		81 / 51	51 / 22
ABT [25]	#Tracks	584	802	–
	#FPFT	53	62	
	Combined	41 / 4	48 / 32	
PF [10]	#Tracks	704	1216	963
	#FPFT	79	202	150
	Combined	59 / 3	171 / 62	113 / 40

Table 2: False positive removal on face tracks obtained from different detection and tracking methods.



Please note:
We show the
true positive
misclassification
axis only up to
10% as we already
discard ~94% FPFTs.

Fig. 5: ROC curve for FPFT detection for BBT and BUFFY.

Row 8 (Combined) presents the results for the logistic regression based fusion of all the methods. We see that about 75% of the FPFT can be removed, while misclassifying ∼3.6% of true positive tracks.

Finally, the last 2 rows of Table 1 present the multiple object tracking accuracy (MOTA) [24] – a tracking evaluation measure – before and after removing false positive tracks. The MOTA score incorporates false positives, missed detections and track switches, and thus an average improvement of 5.4% is substantial.

We present the FPFT detection ROC in Fig. 5. Note that depending on the application, ∼60% of the FPFT can be removed at the cost of only 1% of true positive tracks.

Different detection / tracking schemes We analyze the effect of false positive removal on different detection / tracking methods. We compare (i) VGG [2], which uses Haar-cascade based face detection [7] and tracking using KLT [9]. Tracks which have more than 3 frames are selected, pre-filtering hundreds of false positive and small tracks. (ii) ABT [25], an offline tracker which uses MCT detections [8] and a multi-stage association based tracking scheme; and (iii) PF [10], an online particle filter based tracker which also uses MCT detectors (this is our primary tracker and is also used in Table 1).

Table 2 shows a similar reduction in FPFTs for ABT [25], thus suggesting that the cues work across online and offline tracking schemes. Further, we see that we are able to detect roughly 54% of the FPFT from VGG tracks [2] which use a different detector [7], at the cost of less than 5% of true positive tracks.

The above experiments demonstrate the effectiveness of our method over different detection and tracking procedures. For a fair comparison, the results are obtained with the same parameters across all scenarios.

4. CONCLUSION

Automatic person identification primarily relies on good face detection and tracking. We present a method to automatically remove false positive face tracks using both confidence and context based cues. We propose to use a logistic regression model for the final fusion of individual methods and evaluate our methods on a data set consisting of 11400 tracks. We are able to remove 75% of the false positives at the cost of less than 4% of true positive tracks, improving MOTA by over 5%. We also show promising results on face tracks obtained from different detection and tracking schemes.

Acknowledgments This work was supported by the Quaero Program, funded by OSEO, French State agency for innovation; TUBITAK within the CHIST-ERA project titled CAMOMILE, project no. 112E176; and a Marie Curie FP7 Integration Grant within the 7th EU Framework Programme.

5. REFERENCES

[1] M. Everingham, J. Sivic, and A. Zisserman, ""Hello! My name is... Buffy" – Automatic naming of characters in TV video," in *British Machine Vision Conference (BMVC)*, 2006.

[2] J. Sivic, M. Everingham, and A. Zisserman, ""Who are you?" – Learning person specific classifiers from video," in *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[3] T. Cour, B. Sapp, C. Jordan, and B. Taskar, "Learning from ambiguously labeled images," in *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[4] T. Cour, B. Sapp, A. Nagle, and B. Taskar, "Talking Pictures: Temporal Grouping and Dialog-Supervised Person Recognition," in *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[5] C. Liang, C. Xu, J. Cheng, and H. Lu, "TVParser: An Automatic TV Video Parsing Method," in *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[6] M. Bäuml, M. Tapaswi, and R. Stiefelhagen, "Semi-supervised Learning with Constraints for Person Identification in Multimedia Data," in *Computer Vision and Pattern Recognition (CVPR)*, 2013.

[7] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *International Journal on Computer Vision (IJCV)*, vol. 57, no. 2, pp. 137–154, 2004.

[8] B. Fröba and A. Ernst, "Face Detection with the Modified Census Transform," in *Automatic Face and Gesture Recognition (FG)*, 2004.

[9] J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition (CVPR)*, 1994.

[10] M. Bäuml, K. Bernardin, M. Fischer, H. K. Ekenel, and R. Stiefelhagen, "Multi-Pose Face Recognition for Person Retrieval in Camera Networks," in *Advanced Video and Signal-Based Surveillance (AVSS)*, 2010.

[11] P. Sudowe and B. Leibe, "Efficient Use of Geometric Constraints for Sliding-Window Object Detection in Video," in *Intl. Conference on Computer Vision Systems*, 2011.

[12] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool, "Pedestrian detection at 100 frames per second," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[13] X. Wang, G. Hua, and T. Han, "Detection by Detections: Nonparametric Detector Adaptation for a Video," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[14] P. Sharma and R. Nevatia, "Efficient Detector Adaptation for Object Detection in a Video," in *Computer Vision and Pattern Recognition (CVPR)*, 2013.

[15] O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell, "Face Recognition with Image Sets Using Manifold Density Divergence," in *Computer Vision and Pattern Recognition (CVPR)*, 2005.

[16] M. J. Jones and J. M. Rehg, "Statistical Color Models with Application to Skin Detection," *International Journal on Computer Vision (IJCV)*, vol. 46, no. 1, pp. 81–96, 2002.

[17] H. Li and L. Chen, "Removal of False Positive in Object Detection with Contour-based Classifiers," in *International Conference on Image Processing (ICIP)*, 2010.

[18] C. Atanasoaei, C. McCool, and S. Marcel, "A principled approach to remove false alarams by modelling the context of a face detector," in *British Machine Vision Conference (BMVC)*, 2010.

[19] C. Huang, B. Wu, and R. Nevatia, "Robust Object Tracking by Hierarchical Association of Detection Responses," in *European Conference on Computer Vision (ECCV)*, 2008.

[20] Q. Chen, H. Wu, and M. Yachida, "Face Detection by Fuzzy Pattern Matching," in *International Conference on Computer Vision (ICCV)*, 1995.

[21] D. Li, G. Wei, I. K. Sethi, and N. Dimitrova, "Person identification in TV programs," *Journal of Electronic Imaging*, vol. 10, no. 4, pp. 930–938, 2001.

[22] T. Berg and P. Belhumeur, "Tom-vs-Pete Classifiers and Identity-Preserving Alignment for Face Verification," in *British Machine Vision Conference (BMVC)*, 2012.

[23] M. Dantone, J. Gall, G. Fanelli, and L. Van Gool, "Real-time Facial Feature Detection using Conditional Regression Forests," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[24] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," *EURASIP Journal on Image and Video Processing*, pp. 1–10, 2008.

[25] M. Roth, M. Bäuml, R. Nevatia, and R. Stiefelhagen, "Robust Multi-Pose Face Tracking by Multi-Stage Tracklet Association," in *International Conference on Pattern Recognition (ICPR)*, 2012.