Tracking Persons-of-Interest via Adaptive Discriminative Features

Shun Zhang[1], Yihong Gong[1], Jia-Bin Huang[2], Jongwoo Lim[3], Jinjun Wang[1], Narendra Ahuja[2] and Ming-Hsuan Yang[4]

[1]Xi'an Jiaotong University [2]University of Illinois at Urbana-Champaign
[3] Hanyang University [4]University of California at Merced
http://shunzhang.me.pn/papers/eccv2016/

1 Overview

In this supplementary material, we present extensive experimental evaluation and algorithmic details to complement the manuscript.

1. We present quantitative evaluation in terms of clustering purity and multi-target tracking metrics in Section 2.
2. We show qualitative evaluation by visualizing the face tracking results overlaid on the video sequences in Section 3.
3. We describe algorithmic details for the improved triplet loss in Section 4 and hierachical tracklet linking in Section 5.

2 Quantitative Evaluation

2.1 Datasets

The 8 challenging music videos tested in our experiments are publicly available on YouTube. In Table 1, we list the links of all music videos. The sequences T-ARA, WESTLIFE, and PUSSYCAT DOLLS are live music concert recordings and acquired from multiple cameras with different views. The other sequences BRUNO MARS, APINK, HELLO BUBBLE, DARLING, and GIRLS ALOUD are MTV videos taken in different scenes. All music videos contain large face appearance variations across different shots due to changes in pose, view angle, scale, makeup, illumination, camera motion, and heavy occlusion. In Figures 1–4, we show randomly selected sample faces in temporal order in the videos for each person (using ground truth annotations) to illustrate the intra-class variations and inter-class variations on four challenging sequences (APINK, DARLING, T-ARA and BRUNO MARS). Table 2 summarizes the statistics of these videos, including the duration, frames, and the number of shots, tracklets, detections, and main casts.

2.2 Clustering Evaluation

We compare our method with four recent state-of-the-art face clustering algorithms [1,2,3,4] on the Frontal and BBT01 videos, which all exploit the visual constraints

Table 1. Links to the music video dataset.

Tara	https://www.youtube.com/watch?v=ai1_E5bMsp8
Pussycat Dolls	https://www.youtube.com/watch?v=I4v_22Kk0mM
Bruno Mars	https://www.youtube.com/watch?v=OPf0YbXqDm0
Apink	https://www.youtube.com/watch?v=CDhfIgS4aAo
Hello Bubble	https://www.youtube.com/watch?v=9lSJMKi184c
Darling	https://www.youtube.com/watch?v=QB4dQcxgJPY
Westlife	https://www.youtube.com/watch?v=h4T23UlySTY
Girls Aloud	https://www.youtube.com/watch?v=bBPtP4t2J1k

Table 2. Statistics of the video datasets used in our experiments.

Video	Duration (sec)	Frames	Main casts	Shot changes	Tracklets	Face detections
Frontal	51	1,277	4	0	43	4,267
BBT01	1,373	32,976	7	402	689	51,981
BBT02	1,271	30,481	6	375	793	52,327
BBT03	1,328	31,848	13	406	903	63,659
BBT04	1,246	29,881	5	370	907	59,342
BBT05	1,217	29,185	5	321	850	61,029
BBT06	1,267	30,385	5	353	844	85,054
BBT07	1,273	30,522	10	372	611	52,450
T-ara	152	4,547	6	68	280	12,595
Pussycat Dolls	198	5,937	6	34	272	17,515
Bruno Mars	270	6,483	11	165	507	14,837
Apink	220	5,275	6	162	249	6,294
Hello Bubble	157	3,769	6	116	236	4,731
Darling	197	4,729	8	203	637	11,522
Westlife	229	5,736	4	45	680	27,306
Girls Aloud	221	5,531	5	134	984	22,798

from tracklets for face clustering. Unlike the methods based on hand-crafted features and linear transformations, we apply a deep nonlinear metric learning method by adapting all layers of the CNN to learn discriminative features for faces of specific videos. Table 3 shows the clustering accuracy results over faces and tracklets (using the same datasets and metrics as [1,3])[1]. The results show that our adaptive features achieve higher clustering accuracy than the other three baseline features and competing methods on both videos by a large margin. We attribute the performance improvement to the feature adaptation to the *specific* video for capturing face appearance variations in the video.

Figures 5–6 show the quantitative comparison of different features with clustering purity versus the number of clusters on 7 BBT sequences and 8 music videos. The ideal line (purple dash line) means that all faces are correctly grouped into ideal clusters with weighted purity $W_C = 1$. For more effective features, their weighted purity measures approach to 1 at a faster rate. For each feature, we show the weighted purity at the ideal

[1] The code and data of some other methods (e.g., [5]) are not publicly available.

Table 3. Clustering accuracy on the Frontal and BBT01 videos. We compare our results with three baseline features and four other state-of-the-art face clustering methods [1,2,3,4] based on the same face tracks input and metrics as in [1,3].

Method	Frontal		BBT01	
	faces	tracklets	faces	tracklets
HOG	0.411	0.402	0.495	0.472
AlexNet	0.591	0.435	0.716	0.698
Pre-trained	0.777	0.381	0.747	0.775
Cinbis-ICCV-11 [2]	0.844	0.861	0.581	0.565
Wu-CVPR-13 [3]	0.950	0.907	0.626	0.596
Wu-ICCV-13 [1]	0.950	0.907	0.665	0.668
Xiao-ECCV-14 [4]	0.962	0.938	0.694	0.721
Ours-SymTriplet	**0.998**	**0.998**	**0.939**	**0.978**

number cluster (i.e., the number of people in the video) in the legend. The figures show that identity-preserving features (Pre-trained and VGG-Face) trained on face recognition datasets offline achieve better performance than the generic feature representation (e.g., AlexNet and HOG). Our video-specific features (Ours-SymTriplet) achieve superior performance to all other alternatives, highlighting the importance of learning adaptive features.

2.3 Multi-target Tracking Evaluation

Evaluation metrics. We conduct experimental evaluations and comparisons on multi-face tracking using a comprehensive metric set in [6]. We list these evaluation metrics in Table 4. The up and down arrows indicate whether higher scores or lower scores are sought after for each respective variable.

Experimental results on multi-face tracking. Table 6-7 show quantitative results of the proposed algorithm and the mTLD [7], ADMM [8] and IHTLS [9] on the BBT dataset. Table 8-9 show the quantitative results on the music video dataset. The mTLD method achieves the lowest performance in term of Recall, Precision, F1 and MOTA on both datasets. We attribute the poor performance to its tendency to drift and the use of low-level features (Haar-like features). The ADMM [8] and IHTLS [9] often produce many identity switches and fragments because they fail to re-identify persons when abrupt camera motions or shot changes occur. Using the pre-trained features, our method does not perform well in terms of F1 and MOTA, as the offline features are not effective for linking the tracklets from one person in clustering. Ours-mTLD has more IDS and Frag than Ours-SymTriplet. The main reason is that the shot-level trajectories by mTLD are shorter and noisier than the original trajectories, since TLD trackers sometimes drift or do not output tracking results when there are large appearance changes. With the video specific features (Ours-SymTriplet), the proposed method achieves improved performance in terms of precision, F1, and MOTA metrics, with significantly fewer identity switches and fragments than the ADMM and IHTLS.

Table 5 shows the quantitative results with comparisons to two recent state-of-the-art multi-face trackers [1,3] on the Frontal and BBT01 videos. For fair comparisons, we

Table 4. Evaluation metrics for multi-face tracking. The up and down arrows indicate whether higher scores or lower scores are sought after for each respective variable.

Name	**Definition**
Recall↑	(Frame-based) correctly matched objects / total ground truth objects
Precision ↑	(Frame-based) correctly matched objects / total output objects
F1 ↑	The harmonic mean of precision and recall. $F1 = 2(Precision \cdot Recall)/(Precision + Recall)$
FAF ↓	(Frame-based) No. of false alarms per frame
GT	No. of ground truth trajectories
MT ↑	Mostly tracked: Percentage of GT trajectories which are covered by tracker output for more than 80% in length
PT ↓	Partially tracked: Percentage of GT trajectories which are covered by tracker output for less than 80% in length and more than 20%
Frag ↓	Fragments: The total of No. of times that a ground truth trajectory is interrupted in tracking result
IDS ↓	ID switches: The total of No. of times that a tracked trajectory changes its matched GT identity
MOTA ↑	The Multiple Object Tracking Accuracy takes into account false positives, missed targets and identity switches
MOTP ↑	The Multiple Object Tracking Precision is simply the average distance between true and estimated targets

use the same tracklet inputs as [1,3]. Note that our algorithm performs better in terms of tracking accuracy with fewer ID switches and fragments, which suggests the proposed adaptive discriminative features are effective in identifying faces across multiple shots.

Table 5. Comparison with other state-of-the-art multi-face tracking algorithms. Experimental results of tracklet linking on BBT01 and Frontal videos. We use the same tracklet inputs and metrics as in [1,3]. **GT**: ground-truth tracks pre-defined based on a threshold of the frame gap $t_0 = 150$. **NPT**: the number of predicted tracks. **MT**: mostly tracked tracks. **Frag**: number of fragments. **IDS**: number of ID switch. The **best** value is highlighted with the bold.

Method	BBT01					Frontal				
	GT	**NPT**	**MT↑**	**Frag↓**	**IDS↓**	**GT**	**NPT**	**MT↑**	**Frag↓**	**IDS↓**
Wu-CVPR-13[3]	73	72	68	81	10	5	11	4	24	13
Method1 [1]	73	74	64	82	9	5	11	4	24	13
Method2 [1]	73	79	68	83	4	5	15	**5**	25	5
Ours-SymTriplet	73	73	**72**	**74**	**1**	5	18	**5**	**19**	**1**

(a) Person 1

(b) Person 2

(a) Person 3

(b) Person 4

Person 5

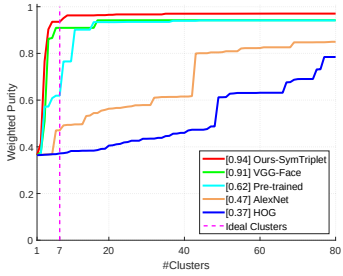
Person 6

Fig. 1. Sampling ground truth faces of 6 people on the APINK sequence to illustrate the intra-person and inter-person appearance variations.

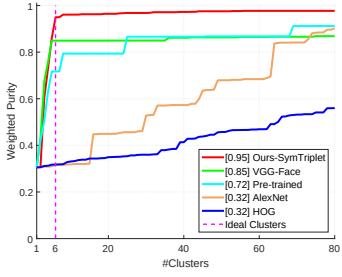
Fig. 2. Sampling ground truth faces of 4 people on the DARLING sequence to illustrate the intra-person and inter-person appearance variations.

Fig. 3. Sampling ground truth faces of 6 people on the T-ARA sequence to illustrate the intra-person and inter-person appearance variations.

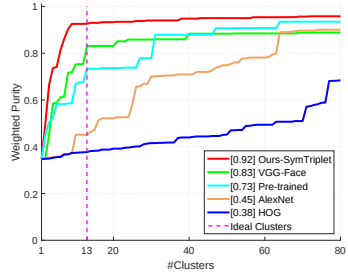
Fig. 4. Sampling ground truth faces of 6 people on the BRUNO MARS sequence to illustrate the intra-person and inter-person appearance variations.



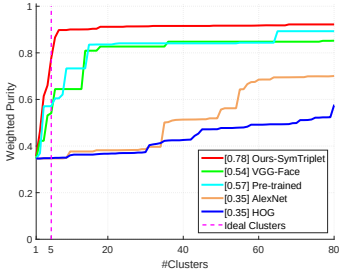
(a) BBT01



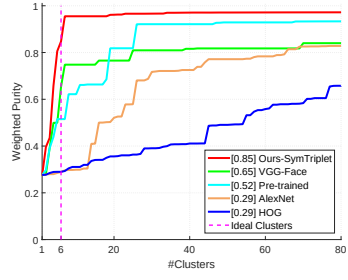
(b) BBT02



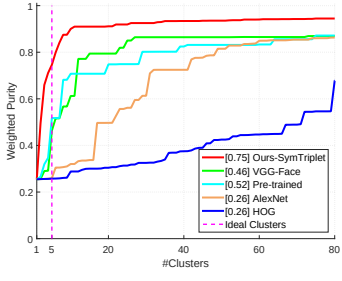
(c) BBT03



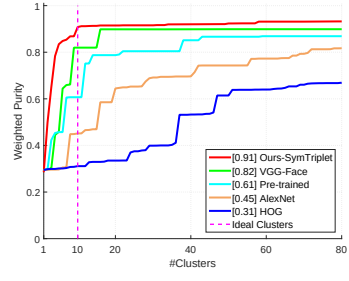
(d) BBT04



(e) BBT05

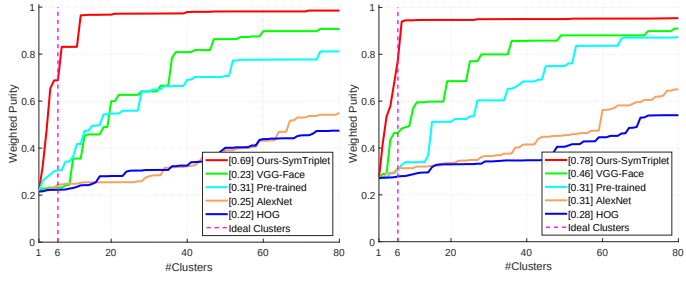


(f) BBT06



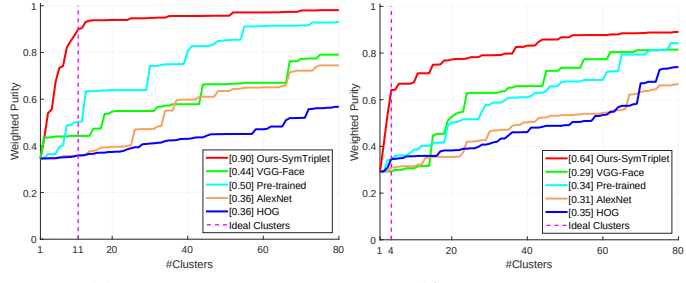
(g) BBT07

Fig. 5. Quantitative comparison of different features with the clustering purity versus the number of clusters on the BBT dataset. The ideal line (dash line) means that all faces are correctly grouped into ideal clusters with weighted purity $W_C = 1$. For more effective features, their weighted purity measures approach to 1 with a faster rate. For each feature, we show the weighted purity at the ideal number cluster in the legend.



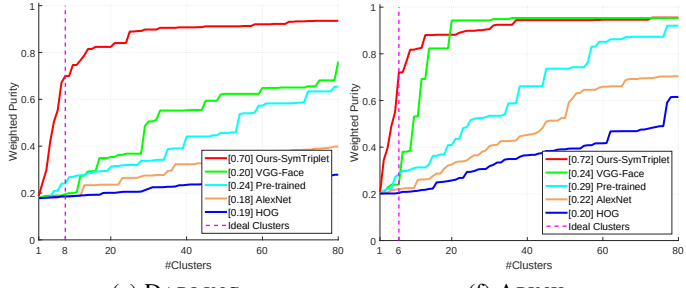
(a) T-ARA

(b) PUSSYCAT DOLLS



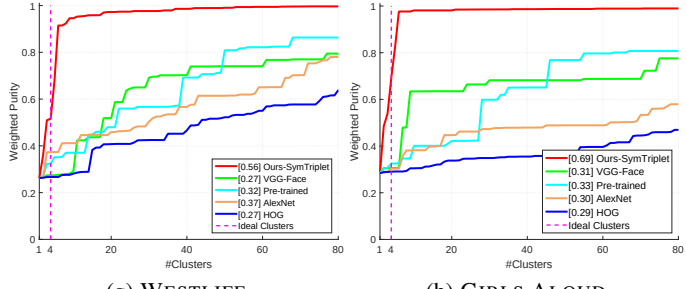
(c) BRUNO MARS

(d) HELLO BUBBLE



(e) DARLING

(f) APINK



(g) WESTLIFE

(h) GIRLS ALOUD

Fig. 6. Quantitative comparison of different features with the clustering purity versus the number of clusters on the music video dataset. The ideal line (dash line) means that all faces are correctly grouped into ideal clusters with weighted purity $W_C = 1$. For more effective features, their weighted purity measures approach to 1 at a faster rate. For each feature, we show the weighted purity at the ideal number cluster in the legend.

Table 6. Quantitative comparison with other state-of-the-art multi-target tracking methods on the BBT01-BBT04 videos. The **best** and second best results are highlighted with the bold and underline, respectively.

BBT01											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	1.0	5.7	1.7	0.25	7	0	**1**	**1**	**5**	-16.3	**74.8**
ADMM [8]	73.5	65.9	69.5	0.40	7	2	5	323	894	42.5	64.0
IHTLS [9]	73.4	71.2	72.3	0.36	7	2	5	312	890	45.7	64.0
Pre-trained	49.0	90.3	63.5	0.1	7	0	5	171	394	41.9	73.3
Ours-mTLD	67.0	91.4	77.3	0.09	7	0	6	223	556	58.4	73.8
Ours-Siamese	75.4	**93.8**	83.6	**0.07**	7	0	7	144	583	69	73.7
Ours-Triplet	77.3	92.5	84.2	0.09	7	1	6	164	610	69.3	73.6
Ours-SymTriplet	**80.8**	92.1	**86.1**	0.1	7	**4**	3	156	651	**72.2**	73.7

BBT02											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	0.2	2.7	0.4	0.11	6	0	**1**	**1**	**1**	-7.6	**82.8**
ADMM [8]	**73.1**	65.5	69.1	0.40	6	**1**	5	395	602	41.3	71.3
IHTLS [9]	72.1	74.1	73.1	0.37	6	0	6	394	587	42.4	71.4
Pre-trained	33.2	88.4	48.3	**0.06**	6	0	4	130	296	27.4	74.5
Ours-mTLD	51.6	89.5	65.5	0.09	6	0	6	174	434	43.6	75.9
Ours-Siamese	66.0	**93.8**	77.5	**0.06**	6	0	6	116	547	60.4	75.8
Ours-Triplet	68.2	91.4	78.1	0.09	6	0	6	143	582	60.2	75.7
Ours-SymTriplet	68.5	92.2	**78.6**	0.08	6	0	6	102	589	**61.6**	75.7

BBT03											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	4.3	40.6	7.8	**0.08**	13	1	**2**	**5**	**52**	-2.1	**69.4**
ADMM [8]	**82.3**	53.2	64.6	0.57	13	**9**	3	370	662	30.8	68.1
IHTLS [9]	81.7	59.2	68.7	0.53	13	**9**	3	376	650	33.5	68.0
Pre-trained	39.6	66.1	49.5	0.25	13	2	4	110	376	17.8	67.5
Ours-mTLD	63.9	72.6	68.0	0.29	13	6	6	142	530	38.0	67.9
Ours-Siamese	76.3	**77.4**	**76.8**	0.27	13	3	8	109	655	**52.6**	67.9
Ours-Triplet	76.9	75.7	76.3	0.3	13	5	7	121	664	50.7	67.8
Ours-SymTriplet	76.8	76.7	76.7	0.28	13	4	7	126	664	51.9	67.8

BBT04											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	0.5	3.1	0.9	**0.18**	5	0	**0**	**0**	**1**	-15.9	**76.8**
ADMM [8]	**73.6**	40.9	52.6	0.63	5	**1**	4	298	621	9.7	65.8
IHTLS [9]	72.3	45.7	56.0	0.58	5	0	5	295	594	13.3	65.8
Pre-trained	27.3	50.8	35.5	0.28	5	0	4	46	217	0.1	66.3
Ours-mTLD	53.7	57	55.3	0.43	5	0	5	103	424	11.6	66.3
Ours-Siamese	68.3	**60.8**	**64.3**	0.47	5	0	5	85	543	**23**	66.4
Ours-Triplet	70.1	58.2	63.6	0.54	5	0	5	103	580	18	66.4
Ours-SymTriplet	70.1	58.7	63.9	0.53	5	0	5	77	580	19.5	66.4

Table 7. Quantitative comparison with other state-of-the-art multi-target tracking methods on the BBT05-BBT07 videos. The **best** and second best results are highlighted with the bold and underline, respectively.

BBT05											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	1.6	8.4	2.7	0.24	6	0	**0**	**1**	**21**	-15.5	**76.9**
ADMM [8]	**84.1**	57.7	68.4	0.59	6	**5**	1	380	488	37.4	68.2
IHTLS [9]	83.8	64.8	73.1	0.64	6	4	2	360	474	33.8	68.2
Pre-trained	49.9	75.3	60.0	**0.23**	6	1	5	98	302	32.3	75
Ours-mTLD	66.8	78.4	72.1	0.26	6	0	6	169	401	46.4	74.9
Ours-Siamese	79.1	**82.5**	80.8	**0.23**	6	3	3	128	477	60.7	75
Ours-Triplet	80.6	81.2	**80.9**	0.26	6	4	2	118	499	60.5	74.9
Ours-SymTriplet	80.6	81.2	**80.9**	0.26	6	4	2	90	497	**60.9**	74.9

BBT06											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	0	1.2	0	**0.07**	5	0	**0**	**0**	**0**	-3.9	89.3
ADMM [8]	**85.8**	51.9	64.7	0.58	5	**4**	1	527	556	47.5	97.6
IHTLS [9]	85.1	60.4	70.7	0.64	5	**4**	1	515	515	43.2	97.7
Pre-trained	51.6	70.1	59.4	0.38	5	0	5	191	405	27.8	**98.2**
Ours-mTLD	67.3	70.8	69.0	0.48	5	0	5	192	591	37.7	97.8
Ours-Siamese	77.9	72	74.8	0.53	5	2	3	156	672	46.2	97.9
Ours-Triplet	77.2	72	74.5	0.52	5	1	4	185	661	45.4	98.0
Ours-SymTriplet	75.8	**74.2**	**75.0**	0.46	5	1	4	196	646	**47.6**	98.0

BBT07											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	0.5	2.4	0.8	0.30	10	0	**1**	**0**	**3**	-17.9	**78.5**
ADMM [8]	74.6	67.3	70.8	0.27	10	1	9	416	800	54.2	64.3
IHTLS [9]	74.4	71.7	73.0	0.32	10	1	9	396	786	51.0	64.4
Pre-trained	56.2	91.6	69.7	**0.08**	10	1	6	162	445	49.4	75.3
Ours-mTLD	71.3	93.3	80.8	**0.08**	10	0	10	221	551	64.0	76.0
Ours-Siamese	76.4	**94.3**	84.4	**0.08**	10	1	8	146	574	70.3	75.9
Ours-Triplet	80.8	93.3	86.6	0.1	10	**4**	6	110	627	73.9	75.9
Ours-SymTriplet	**81.2**	93	**86.7**	0.1	10	**4**	6	99	634	**74.1**	75.9

Table 8. Quantitative comparison with other state-of-the-art multi-target tracking methods on the HELLO BUBBLE, T-ARA, PUSSYCAT DOLLS and BRUNO MARS videos. The **best** and second best results are highlighted with the bold and underline, respectively.

HELLO BUBBLE											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	3.8	34.7	6.8	**0.10**	4	0	**0**	**7**	**20**	-3.5	66.5
ADMM [8]	66.1	80.2	72.5	0.23	4	0	4	115	191	47.6	69.9
IHTLS [9]	65.9	84.8	74.2	0.16	4	0	4	109	190	52.0	69.9
Pre-trained	47.1	83.8	60.3	**0.10**	4	0	4	71	187	36.6	68.5
Ours-mTLD	67.4	84.8	75.1	0.17	4	0	4	139	255	52.6	70.5
Ours-Siamese	67.6	**88**	**76.5**	0.13	4	0	4	105	249	56.3	**70.6**
Ours-Triplet	**68.6**	86.4	**76.5**	0.15	4	0	4	82	256	56.2	70.5
Ours-SymTriplet	**68.6**	86.5	**76.5**	0.15	4	0	4	69	256	**56.5**	70.5

T-ARA											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	24.7	52.4	33.6	0.72	6	0	**3**	130	**148**	1.4	67.9
ADMM [8]	58.0	68.3	62.8	0.86	6	0	6	251	641	29.4	63.8
IHTLS [9]	58.0	73.2	64.7	0.68	6	0	6	218	632	35.3	63.8
Pre-trained	60.9	**95.9**	74.5	0.10	6	0	6	143	232	57.3	72.4
Ours-mTLD	62.1	93.5	74.6	0.14	6	0	6	251	241	56	**72.6**
Ours-Siamese	62.1	95.5	75.3	**0.09**	6	0	6	106	213	58.4	72.5
Ours-Triplet	**63.5**	94.2	**75.9**	0.12	6	0	6	94	233	59.0	72.5
Ours-SymTriplet	62.8	95.4	75.7	0.10	6	0	6	**75**	235	**59.2**	72.4

PUSSYCAT DOLLS											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	13.4	56.9	21.7	0.24	6	0	**1**	**24**	**78**	3.1	**71.3**
ADMM [8]	89.3	74.2	81.0	0.58	6	**4**	2	287	412	63.2	63.5
IHTLS [9]	**89.5**	78.6	83.7	0.42	6	**4**	2	248	413	**70.3**	63.5
Pre-trained	76.4	88.0	81.8	0.3	6	2	4	128	405	65.1	64.9
Ours-mTLD	79.7	**89.5**	84.3	**0.22**	6	2	4	296	444	68.3	64.9
Ours-Siamese	81.2	88.9	**84.9**	0.24	6	2	4	107	430	**70.3**	64.9
Ours-Triplet	81.4	88.3	84.7	0.26	6	2	4	99	435	69.9	64.9
Ours-SymTriplet	81.6	88.2	84.8	0.26	6	2	4	82	439	70.2	64.9

BRUNO MARS											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	4.7	26.2	7.9	0.34	11	0	**2**	**35**	**77**	-8.7	65.3
ADMM [8]	**68.9**	76.0	72.3	0.40	11	**3**	8	428	503	50.6	85.7
IHTLS [9]	68.5	83.5	**75.2**	0.35	11	**3**	8	375	491	52.7	85.8
Pre-trained	53.7	92.3	67.9	**0.10**	11	0	9	151	453	48.3	**88**
Ours-mTLD	58.0	**94.0**	71.7	**0.10**	11	2	9	278	551	52.6	87.9
Ours-Siamese	62.3	92.8	74.6	0.12	11	2	8	126	540	56.7	87.8
Ours-Triplet	62.4	92.6	74.6	0.13	11	2	9	126	543	56.6	87.8
Ours-SymTriplet	62.9	91.9	74.7	0.14	11	2	9	105	551	**56.8**	87.8

Table 9. Quantitative comparison with other state-of-the-art multi-target tracking methods on the APINK, WESTLIFE, DARLING and GIRLS ALOUD videos. The **best** and second best results are highlighted with the bold and underline, respectively.

APINK											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	16.4	47.5	24.4	0.25	6	0	**2**	**31**	**70**	-2.2	71.2
ADMM [8]	81.2	92.8	86.6	0.09	6	**4**	**2**	179	158	72.4	76.1
IHTLS [9]	81.2	95.4	87.7	0.05	6	**4**	**2**	173	157	74.9	76.1
Pre-trained	56.4	98.3	71.7	**0.01**	6	0	6	100	170	54.0	75.5
Ours-mTLD	81.5	98.0	89.0	0.02	6	3	3	173	240	77.4	**76.3**
Ours-Siamese	81.6	**98.9**	89.4	**0.01**	6	3	3	124	238	79.0	**76.3**
Ours-Triplet	82.1	98.5	89.6	0.02	6	**4**	**2**	140	244	78.9	**76.3**
Ours-SymTriplet	**82.4**	98.3	**89.7**	0.02	6	**4**	**2**	78	246	**80.0**	**76.3**

WESTLIFE											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	4.9	11.1	6.9	0.78	4	0	**0**	**20**	**113**	-34.7	56.9
ADMM [8]	89.1	36.0	51.3	0.60	4	**4**	**0**	223	184	62.4	87.5
IHTLS [9]	**89.4**	39.9	55.2	0.65	4	**4**	**0**	113	177	60.9	87.5
Pre-trained	77.8	79.5	78.6	0.40	4	1	3	85	128	57	**88.2**
Ours-mTLD	86.0	76.5	81.0	0.52	4	3	1	177	169	58.1	88.1
Ours-Siamese	86.8	79.7	83.1	0.44	4	3	1	74	142	64.1	88
Ours-Triplet	86.8	80.1	83.3	0.43	4	3	1	89	140	64.5	88
Ours-SymTriplet	85.6	**83.9**	**84.7**	**0.33**	4	3	1	57	136	**68.6**	88.1

DARLING											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	6.3	18.4	9.4	0.57	8	0	**0**	**24**	**83**	-22.0	69.9
ADMM [8]	88.3	74.0	80.6	0.62	8	**7**	1	412	342	53.0	88.4
IHTLS [9]	**88.5**	80.2	84.2	0.44	8	**7**	1	381	338	62.7	88.4
Pre-trained	53.1	85.2	65.4	**0.20**	8	2	6	115	233	42.7	88.5
Ours-mTLD	79.9	82.3	81.1	0.35	8	4	4	278	461	59.8	**89.3**
Ours-Siamese	85.2	**86.3**	85.7	0.27	8	**7**	1	214	310	69.5	88.9
Ours-Triplet	85.9	85.3	85.6	0.30	8	**7**	1	187	317	69.2	88.9
Ours-SymTriplet	86.7	85.7	**86.2**	0.29	8	**7**	1	169	323	**70.5**	88.9

GIRLSALOUD											
Method	**Recall(%)↑**	**Precision(%)↑**	**F1(%)↑**	**FAF↓**	**GT**	**MT↑**	**PT↓**	**IDS↓**	**Frag↓**	**MOTA(%)↑**	**MOTP(%)↑**
mTLD [7]	2.2	40.7	4.2	**0.10**	5	0	**0**	**9**	**32**	-1.1	71.0
ADMM [8]	**70.0**	50.3	58.5	0.61	5	**1**	4	487	528	46.6	87.1
IHTLS [9]	69.8	60.2	64.7	0.46	5	**1**	4	396	482	**51.8**	87.2
Pre-trained	49.3	89.6	63.6	0.20	5	0	5	138	332	42.7	87.7
Ours-mTLD	54.3	90.5	67.9	0.17	5	0	5	322	425	46.7	**88.2**
Ours-Siamese	58.1	90.8	**70.9**	0.17	5	**1**	4	112	376	51.6	87.8
Ours-Triplet	57.2	**92.0**	70.5	0.15	5	**1**	4	80	367	51.7	87.8
Ours-SymTriplet	58.2	90.3	70.8	0.19	5	**1**	4	64	377	51.6	87.8

3 Qualitative Evaluation

3.1 Multi-face Tracking Visualization on the Music Video Dataset

We show the tracking results of the proposed method with Our-SymTriplet features on the 8 music videos. Figures 7-9 show the tracking results of the proposed method on three unconstrained videos (T-ARA, PUSSYCAT DOLLS and WESTLIFE) taken in live music concerts. They contain large face variations including changes of pose, scale, expression, illumination, etc. For the T-ARA sequence, the six singers have similar looks, which makes the face tracking across shots significantly difficult, e.g., Person 6 and Person 7 in Figure 7. The proposed algorithm is able to distinguish similar faces of different people and track them reliably with few id switches.

Figures 10–14 show the tracking results of the proposed method on 5 MTV videos (GIRLS ALOUD, HELLO BUBBLE, APINK, DARLING and BRUNO MARS). These videos contain not only the large face variations including changes of pose, scale, expression and illumination, but also changes of makeup (e.g., Person 2 in Figure 11, and Person 2 in Figure 13), visual style (e.g., Figure 12). People in the HELLO BUBBLE, APINK and DARLING sequences have similar looks, e.g., Person 3, Person 5 and Person 6 in Figure 12. The proposed algorithm is able to track most of the faces correctly.

3.2 Multi-face Tracking Visualization on BBT Dataset

We show the tracking results of the proposed method with the Our-SymTriplet features on 7 BBT videos, shown in Figures 15-21. The BBT videos are taken mostly indoors, and contain frequent changes of camera views and scenes, where faces have large appearance variations in viewing angle, pose, scale, and illumination. The proposed algorithm is able to track multiple faces correctly.

Fig. 7. Sample tracking results of the proposed algorithm on the T-ARA sequence. The faces of the different people are color coded.

Fig. 8. Sample tracking results of the proposed algorithm on the PUSSYCAT DOLLS sequence. The faces of the different people are color coded.

Fig. 9. Sample tracking results of the proposed algorithm on the WESTLIFE sequence. The faces of the different people are color coded.

Fig. 10. Sample tracking results of the proposed algorithm on the GIRLS ALOUD sequence. The faces of the different people are color coded.

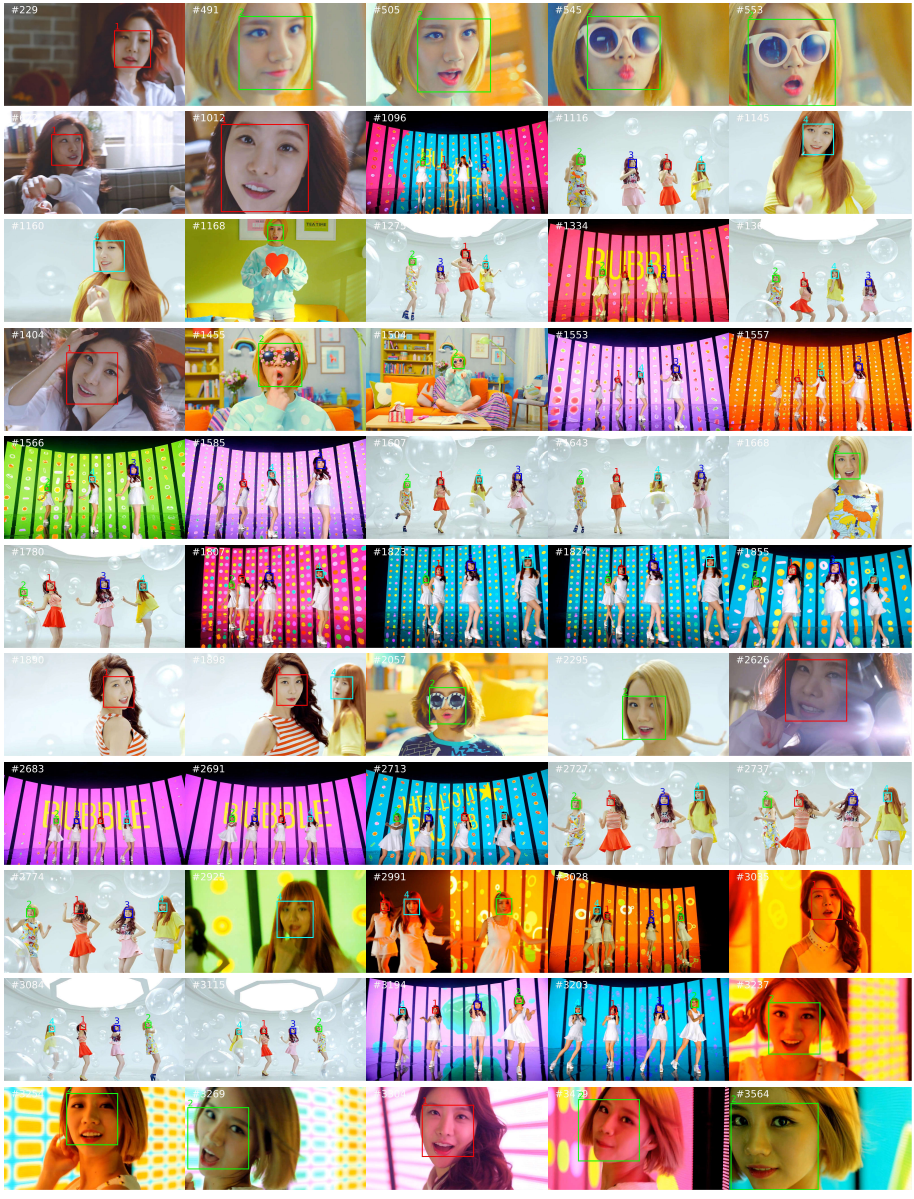


Fig. 11. Sample tracking results of the proposed algorithm on the HELLO BUBBLE sequence. The faces of the different people are color coded.

Fig. 12. Sample tracking results of the proposed algorithm on the APINK sequence. The faces of the different people are color coded.

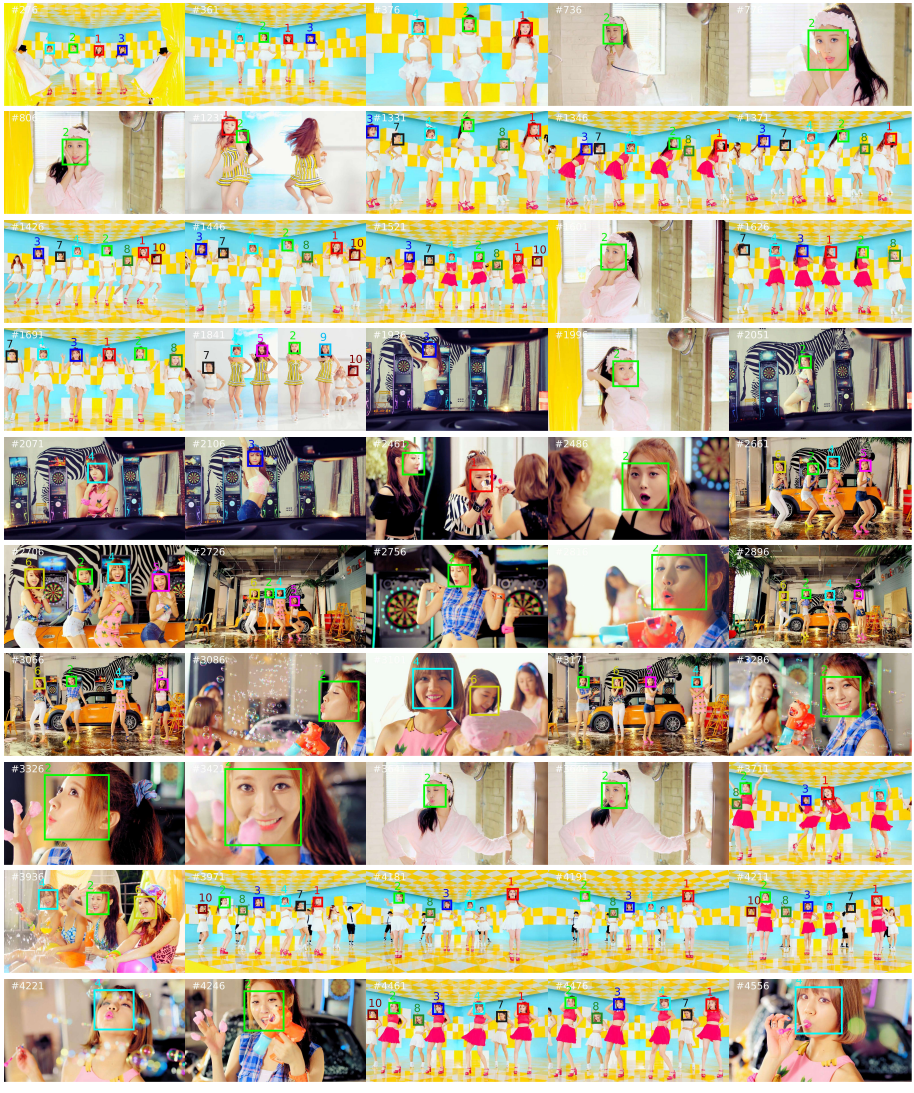


Fig. 13. Sample tracking results of the proposed algorithm on the DARLING sequence. The faces of the different people are color coded.

Fig. 14. Sample tracking results of the proposed algorithm on the BRUNO MARS sequence. The faces of the different people are color coded.

Fig. 15. Sample tracking results of the proposed algorithm on the BBT01 sequence. The faces of the different people are color coded.

Fig. 16. Sample tracking results of the proposed algorithm on the BBT02 sequence. The faces of the different people are color coded.

Fig. 17. Sample tracking results of the proposed algorithm on the BBT03 sequence. The faces of the different people are color coded.

Fig. 18. Sample tracking results of the proposed algorithm on the BBT04 sequence. The faces of the different people are color coded.

Fig. 19. Sample tracking results of the proposed algorithm on the BBT05 sequence. The faces of the different people are color coded.

Fig. 20. Sample tracking results of the proposed algorithm on the BBT06 sequence. The faces of the different people are color coded.

Fig. 21. Sample tracking results of the proposed algorithm on the BBT07 sequence. The faces of the different people are color coded.

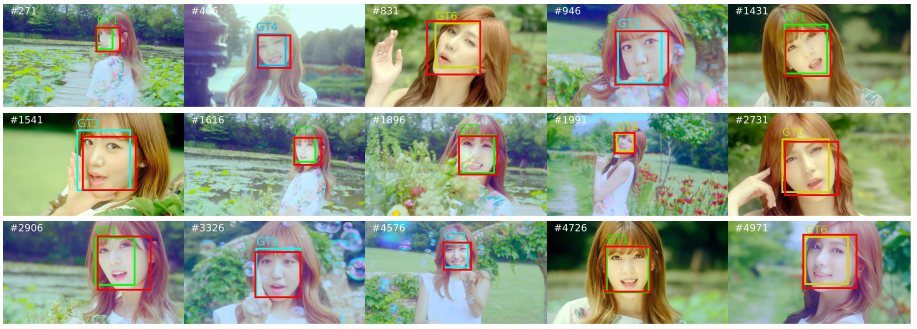


Fig. 22. Failure cases on the APINK sequence. Our method incorrectly identifies Persons 1, 3, 4 and 6 as the same person across shots. Numbers and colors of the rectangles indicate the ground truth identities of persons. The red rectangles show the tracking results of one trajectory (*i.e.* the same person) by our method.

Fig. 23. Failure cases on the DARLING sequence. Our method incorrectly identifies Persons 1 and 4 as the same one across shots. Numbers and colors of rectangles indicate the ground truth identities of persons. The red rectangles show the tracking results of one trajectory (*i.e.* the same person) by our method.

3.3 Failure Modes

In the case where one person has significant appearance variations in different shots, our method has difficulty. For example, in Figure 13, ID5 and ID3 are identified with different trajectories, despite being the same person. We observe the same issue in ID1 and ID6. Similarly, in Figure 12, the ID1, ID8 and ID11 are also incorrectly labeled as different identities.

Figure 22 shows some failure cases on the APINK sequence. Since the video consists of many shots with one single persons, our method cannot generate sufficient negative face pairs to train the Siamese/Triplet network for distinguishing similar faces. Persons 1, 3, 4 and 6 are incorrectly identified as the same person across shots. Figure 23 shows some failure cases on the DARLING sequence. Persons 1 and 4 are incorrectly tracked as one person.

4 Improved Triplet Loss

4.1 Training Algorithm

We train the Triplet network model with the SymTriplet loss function by the stochastic gradient decent with momentum. We compute the derivatives of Eqn. (4) in the manuscript as follows:

$$\frac{\partial L_s}{\partial \mathbf{W}} = \begin{cases} \frac{\partial \widetilde{L_s}}{\partial \mathbf{W}} & L_s > 0, \\ 0 & L_s = 0, \end{cases} \tag{1}$$

where

$$\begin{aligned} \frac{\partial \widetilde{L_s}}{\partial \mathbf{W}} =& 2(\mathbf{f}(\mathbf{x}_k^i) - \mathbf{f}(\mathbf{x}_l^i)) \frac{\partial \mathbf{f}(\mathbf{x}_k^i) - \partial \mathbf{f}(\mathbf{x}_l^i)}{\partial \mathbf{W}} - (\mathbf{f}(\mathbf{x}_k^i) - \mathbf{f}(\mathbf{x}_m^j)) \frac{\partial \mathbf{f}(\mathbf{x}_k^i) - \partial \mathbf{f}(\mathbf{x}_m^j)}{\partial \mathbf{W}} \\ & - (\mathbf{f}(\mathbf{x}_l^i) - \mathbf{f}(\mathbf{x}_m^j)) \frac{\partial \mathbf{f}(\mathbf{x}_l^i) - \partial \mathbf{f}(\mathbf{x}_m^j)}{\partial \mathbf{W}}, \end{aligned} \tag{2}$$

For the above derivations, we can compute the gradients from each input triplet examples given the values of $\mathbf{f}(\mathbf{x}_k^i)$, $\mathbf{f}(\mathbf{x}_l^i)$, $\mathbf{f}(\mathbf{x}_m^j)$ and $\frac{\partial \mathbf{f}(\mathbf{x}_k^i)}{\partial \mathbf{W}}$, $\frac{\partial \mathbf{f}(\mathbf{x}_l^i)}{\partial \mathbf{W}}$, $\frac{\partial \mathbf{f}(\mathbf{x}_m^j)}{\partial \mathbf{W}}$, which can be obtained by running the standard forward and backward propagations separately for each image in the triplet examples. The algorithm needs to go through all the triplets in each batch to accumulate the gradients for each iteration. Algorithm 1 shows the main steps of the training algorithm.

Algorithm 1 Triplet-based training with stochastic gradient descent

Input
  Training samples $\{(\mathbf{x}_k^i, \mathbf{x}_l^i, \mathbf{x}_m^j)\}$.
Output
  The network parameters $\mathbf{W}$,
for $t = 1 \rightarrow$ Max number of iterations **do**
  $\frac{\partial L_s}{\partial \mathbf{W}} = 0$
  for all training triplet samples $(\mathbf{x}_k^i, \mathbf{x}_l^i, \mathbf{x}_m^j)$ **do**
    Calculate $\mathbf{f}(\mathbf{x}_k^i)$, $\mathbf{f}(\mathbf{x}_l^i)$ and $\mathbf{f}(\mathbf{x}_m^j)$ by forward propagation;
    Calculate $\frac{\partial \mathbf{f}(\mathbf{x}_k^i)}{\partial \mathbf{W}}$, $\frac{\partial \mathbf{f}(\mathbf{x}_l^i)}{\partial \mathbf{W}}$ and $\frac{\partial \mathbf{f}(\mathbf{x}_m^j)}{\partial \mathbf{W}}$ by back propagation;
    Calculate $\frac{\partial L_s}{\partial \mathbf{W}}$ according to (1) and (2).
  end for
  Udapte the parameters $\mathbf{W}^t = \mathbf{W}^{t-1} - \lambda_t \frac{\partial L_s}{\partial \mathbf{W}}$
end for

5 Multi-face Tracking via Hierarchical Tracklet Linking

5.1 Linking Tracklets Within Each Shot

We use conventional multi-target tracking algorithms to perform data association of face tracklets within each shot. In this paper, we use the Hungarian algorithm proposed

in [10,11]. We measure the linking probabilities between two tracklets based on temporal, kinematic and appearance information. Here, we present the algorithmic details.

Supposing there are n tracklets in one shot, the goal of tracklet association within each shot is to find the maximum weighted matchings $\mathbf{M} = [m_{ij}]_{n\times n}$, where $m_{ij} \in \{0,1\}$ indicates whether there is a match between $\mathbf{T}^i$ and $\mathbf{T}^j$ ($m_{ij} = 1$ as a match). We can then transfer the data association problem into a standard assignment problem by applying the Hungarian algorithm to the similarity matrix $\mathbf{C}$. The similarity matrix $\mathbf{C}$ can be divided into four block matrices:

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}^a & \mathbf{C}^t \\ \mathbf{C}^s & \mathbf{C}^0 \end{bmatrix}_{2n\times 2n}, \tag{3}$$

where the first block matrix $\mathbf{C}^a = \{c^a(\mathbf{T}^i, \mathbf{T}^j)\}_{n\times n}$ models the association score between tracklets $\mathbf{T}^i$ and $\mathbf{T}^j$; the second block matrix $\mathbf{C}^t = diag\{c_1^t, \cdots, c_n^t\}$ models the likelihood of the tracklet $\mathbf{T}^i$ being a terminal object trajectory; the third block matrix $\mathbf{C}^s = diag\{c_1^s, \cdots, c_n^s\}$ models the likelihood of the tracklet $\mathbf{T}^j$ being an initial object trajectory; $\mathbf{C}^0 = \{0\}_{n\times n}$ serves as a place holder.

Similarity matrix $\mathbf{C}^a$. The value $c^a(\mathbf{T}^i, \mathbf{T}^j)$ is the similarity score between $\mathbf{T}^i$ and $\mathbf{T}^j$. In this paper, we adopt the bounding box representation. Hence, the k^{th} detection response in the trajectory $\mathbf{T}^i = \{\mathbf{x}_1^i, \cdots, \mathbf{x}_{n_i}^i\}$ is represented as $\mathbf{x}_k^i = \{\mathbf{p}_k^i, \mathbf{q}_k^i, \mathbf{a}_k^i, t_k^i\}$, where $\mathbf{p}_k^i$ and $\mathbf{q}_k^i$ are the central position and the size (width, height) of the bounding box, respectively. The vector $\mathbf{a}_k^i$ is the 64-D feature descriptor that is extracted from our learned Siamese/Triplet network. The parameter t_k^i is the frame index.

We define the similarity score $c^a(\mathbf{T}^i, \mathbf{T}^j)$ as follows:

$$c^a(\mathbf{T}^i, \mathbf{T}^j) = w_m\phi_m(\mathbf{T}^i, \mathbf{T}^j) + w_s\phi_s(\mathbf{T}^i, \mathbf{T}^j) + w_a\phi_a(\mathbf{T}^i, \mathbf{T}^j), \text{ s.t. } i \neq j, \tag{4}$$

where $\phi_m(\mathbf{T}^i, \mathbf{T}^j)$, $\phi_s(\mathbf{T}^i, \mathbf{T}^j)$ and $\phi_a(\mathbf{T}^i, \mathbf{T}^j)$ are the similarity scores between $\mathbf{T}^i$ and $\mathbf{T}^j$ in terms of motion trend, size, and appearance descriptors, respectively. The parameters w_m, w_s and w_a are the weights of $\phi_m(\mathbf{T}^i, \mathbf{T}^j)$, $\phi_s(\mathbf{T}^i, \mathbf{T}^j)$ and $\phi_a(\mathbf{T}^i, \mathbf{T}^j)$, respectively ($w_a = w_s = 0.3$, $w_m = 0.4$ in our implementation). We describe $\phi_m(\mathbf{T}^i, \mathbf{T}^j)$, $\phi_s(\mathbf{T}^i, \mathbf{T}^j)$ and $\phi_a(\mathbf{T}^i, \mathbf{T}^j)$ as follows.

For the motion trend cue, we compute the probability that the tracklet $\mathbf{T}^j$ is linked to $\mathbf{T}^i$:

$$\phi_m(\mathbf{T}^i, \mathbf{T}^j) = \frac{1}{1 + e^{d_m(\mathbf{T}^i, \mathbf{T}^j)}}, \tag{5}$$

where $d_m(\mathbf{T}^i, \mathbf{T}^j)$ denotes the difference between the predicted positions and the positions of the true positions. We fit the two tracklets through the polynomial curve fitting. We use the fitted curve $\hat{\mathbf{p}}^i(\cdot)$ to predict the positions of the tracklet $\mathbf{T}^i$:

$$d_m(\mathbf{T}^i, \mathbf{T}^j) = \sum_{k\in\{1,2,3\}} \|\hat{\mathbf{p}}^i(t_k^j) - \mathbf{p}_k^j\|_2^2 + \sum_{k\in\{n_i-2, n_i-1, n_i\}} \|\hat{\mathbf{p}}^j(t_k^i) - \mathbf{p}_k^i\|_2^2, \tag{6}$$

where $\hat{\mathbf{p}}^i(t_k^j)$ denotes the predicted position of $\mathbf{T}^i$ at the frame t_k^j.

Similarly, for the size similarity, we compute the probability

$$\phi_s(\mathbf{T}^i, \mathbf{T}^j) = \frac{1}{1 + e^{d_s(\mathbf{T}^i, \mathbf{T}^j)}}, \tag{7}$$

where $d_s(\mathbf{T}^i, \mathbf{T}^j)$ denotes the difference between the predicted positions and the positions of the true positions. We also fit the two tracklets through the polynomial curve fitting:

$$d_s(\mathbf{T}^i, \mathbf{T}^j) = \sum_{k \in \{1,2,3\}} \|\hat{\mathbf{q}}^i(t_k^j) - \mathbf{q}_k^j\|_2^2 + \sum_{k \in \{n_i-2, n_i-1, n_i\}} \|\hat{\mathbf{q}}^j(t_k^i) - \mathbf{q}_k^i\|_2^2, \tag{8}$$

where $\hat{\mathbf{q}}^i(t_k^j)$ denotes the predicted size of $\mathbf{T}^i$ at the frame t_k^j.

For the appearance cue, we simply check the similarity of the two tracklets as follows:

$$\phi_a(\mathbf{T}^i, \mathbf{T}^j) = \frac{1}{1 + e^{d_a(\mathbf{T}^i, \mathbf{T}^j)}}, \tag{9}$$

where $d_a(\mathbf{T}^i, \mathbf{T}^j)$ is the Euclidean distance between the two tracklets $\mathbf{T}^i$ and $\mathbf{T}^j$. It is defined as:

$$d_a(\mathbf{T}^i, \mathbf{T}^j) = \frac{1}{n_i} \frac{1}{n_j} \sum_{k=1}^{n_i} \sum_{l=1}^{n_j} \|\mathbf{a}_k^i - \mathbf{a}_l^j\|_2^2. \tag{10}$$

For $i = j$, the self-association of the tracklet $\mathbf{T}^i$ is equivalent to treating it as a false alarm because it cannot be an initial trajectory or a terminated trajectory, or be associated with any other tracklets. We define the likelihood that $\mathbf{T}^i$ is a false alarm as follows:

$$c^a(\mathbf{T}^i, \mathbf{T}^i) = Z_t(1 - \varphi)^{n_i}, \tag{11}$$

where Z_t is a normalization factor, and $\varphi \in (0,1)$ is the precision of the detector (φ is set to 0.8 in our experiments).

Similarity matrix $\mathbf{C}^t$. The matrix $\mathbf{C}^t = diag\{c_1^t, \cdots, c_n^t\}$ is a diagonal matrix of size $n \times n$ defining if $\mathbf{T}^i$ is a terminal object trajectory. Here, we use fixed scores, indicating that each trajectory has a uniform priori probability to be temporally invisible: $\mathbf{C}^t = diag\{0.25, \cdots, 0.25\}$.

Similarity matrix $\mathbf{C}^s$. The matrix $\mathbf{C}^s = diag\{c_1^s, \cdots, c_n^s\}$ models the likelihood of the tracklet $\mathbf{T}^j$ being an initial object trajectory. We empirically set the initialization probabilities of each tracklet as: $\mathbf{C}^t = diag\{0.25, \cdots, 0.25\}$.

We can then apply the Hungarian algorithm on the similarity matrix $\mathbf{C}$ to find the optimal assignment matrix $\mathbf{M}$. For each $m_{ij} = 1$, do as follows:

1. If $i = j \leq n$, $\mathbf{T}^i$ is considered as a false alarm;
2. If $i \leq n$, $j \leq n$ and $i \neq j$, associate $\mathbf{T}^i$ and $\mathbf{T}^j$;
3. If $i \leq n$ and $j > n$, $\mathbf{T}^i$ is considered as a terminated tracklet;
4. If $i > n$ and $j \leq n$, $\mathbf{T}^j$ is considered as a new track.

5.2 Linking Tracklets Across Shots

For linking tracklets across multiple shots, we apply the bottom-up Hierarchical Agglomerative Clustering (HAC) algorithm with a stopping threshold with the learned appearance features as follows:

(a) Supposing there are N tracklets in all shots: $\Gamma = \{\mathbf{T}^1, \mathbf{T}^2, \ldots, \mathbf{T}^N\}$. We start with treating each tracklet as a singleton cluster.

(b) We evaluate all pair-wise distances between tracklets using the mean distance metric: given $\mathbf{T}^i = \{\mathbf{x}^i_1, \ldots, \mathbf{x}^i_{n_i}\}$ and $\mathbf{T}^j = \{\mathbf{x}^j_1, \ldots, \mathbf{x}^j_{n_j}\}$, the distance D^{ij} is defined as:

$$D^{ij} = \frac{1}{n_i}\frac{1}{n_j}\sum_{k=1}^{n_i}\sum_{l=1}^{n_j} \|\mathbf{f}(\mathbf{x}^i_k) - \mathbf{f}(\mathbf{x}^j_l)\|_2^2, \tag{12}$$

where $\mathbf{x}^i_k$ denotes the k^{th} face detection in the i^{th} tracklet, and $\mathbf{f}(\mathbf{x}^i_k)$ denotes the feature extracted from the embedding layer in the Siamese/Triplet network.

(c) For pairs of tracklets which have overlapped frames, we set their distances as infinity.

(d) Find the pair of clusters that has the shortest distance.

(e) Merge the pair into a new cluster, and update all distances from the new cluster to all other clusters. For those clusters which have overlapped frames with the new cluster, the corresponding distances to the new cluster are set to infinity.

(f) Repeat (d)-(e) until the shortest distance is larger than a threshold θ.

The clusters containing less than 4 tracklets and less than 50 frames are removed. The tracklets in each cluster are labeled with the same identity to form the final trajectories.

References

1. Wu, B., Lyu, S., Hu, B.G., Ji, Q.: Simultaneous clustering and tracklet linking for multi-face tracking in videos. In: ICCV. (2013) 1, 2, 3, 4
2. Cinbis, R.G., Verbeek, J., Schmid, C.: Unsupervised metric learning for face identification in tv video. In: ICCV. (2011) 1, 3
3. Wu, B., Zhang, Y., Hu, B.G., Ji, Q.: Constrained clustering and its application to face clustering in videos. In: CVPR. (2013) 1, 2, 3, 4
4. Xiao, S., Tan, M., Xu, D.: Weighted block-sparse low rank representation for face clustering in videos. In: ECCV. (2014) 123–138 1, 3
5. Tapaswi, M., Parkhi, O.M., Rahtu, E., Sommerlade, E., Stiefelhagen, R., Zisserman, A.: Total cluster: A person agnostic clustering method for broadcast videos. In: ICVGIP. (2014) 2
6. Zhang, S., Wang, J., Wang, Z., Gong, Y., Liu, Y.: Multi-target tracking by learning local-to-global trajectory models. PR **48**(2) (2015) 580–590 3
7. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. TPAMI **34**(7) (2012) 1409–1422 3, 11, 12, 13, 14
8. Ayazoglu, M., Sznaier, M., Camps, O.I.: Fast algorithms for structured robust principal component analysis. In: CVPR. (2012) 3, 11, 12, 13, 14
9. Dicle, C., Camps, O.I., Sznaier, M.: The way they move: Tracking multiple targets with similar appearance. In: ICCV. (2013) 3, 11, 12, 13, 14
10. Huang, C., Wu, B., Nevatia, R.: Robust object tracking by hierarchical association of detection responses. In: ECCV. (2008) 33
11. Perera, A.A., Srinivas, C., Hoogs, A., Brooksby, G., Hu, W.: Multi-object tracking through simultaneous long occlusions and split-merge conditions. In: CVPR. (2006) 33