Tracking Pedestrian Heads in Dense crowd

Ramana Sundararaman Cédric De Almeida Braga Eric Marchand Julien Pettré
Univ Rennes, Inria, CNRS, Irisa, Rennes, France
ramanasubramanyam.sundararaman@polytechnique.edu,
{cedric.de-almeida-braga,julien.pettre}@inria.fr,
eric.marchand@irisa.fr

Abstract

In this supplementary material, we provide more detailed insights into the statistics of our dataset and its annotation procedure. We also report the influence of hyperparameters of trackers, which we have used for performing baseline experiments. Finally, we explore the role of various head detectors in tracking performances and present the sequence-wise result of HeadHunter and HeadHunter-T on CroHD.

1. CroHD Annotation

We annotated heads of pedestrians in this dataset in order to reduce the intra-target occlusions. The annotation work was performed with the help of Crowdsourcing platform, Fiverr [1] using the CVAT Annotation tool [2]. Due to the number of targets to be tracked being plentiful, while the area of tracking is significantly smaller than existing approaches, the margin for errors in this annotation procedure is large. As a result, we employed a three-stage reviewing process for thoroughgoing the annotation. First, we automated the process of spotting identity switches and track fragmentations, which were the most common mistakes made by annotators. Then, the annotations corresponding to a sequence were reviewed by a team of annotators, separate from those who annotated the particular scene, to avoid any bias. Finally, we (the authors of this work) manually inspected the annotation.

Automation of reviewing: A pedestrian head is assigned an ID as soon as it becomes visible and the same ID is maintained until it leaves the field of view (FoV). Using this information, we gathered tracks which have not terminated near the image boundary, with the last few frames being an exception. This helped us in identifying tracks whose annotations have been fragmented. Another common mistake in annotations were identity switches, when the identity of two pedestrian heads end up mutually swapping. In order to spot this, for each target, we analyzed the displacement of respective bounding box centroids. If at a particular frame, the motion of a particular track was two standard deviations away from the mean displacement, such tracks were flagged for a potential identity switch review. Note that both methods mentioned in this section are not complete and do not recognize all fragmentation and identity switches. However, they have significantly helped in minimizing human efforts in spotting such errors.

Visibility: Figure 1 shows an example of various types of occluders across all scenes in our dataset. Occluders in the scene, which are either opaque or translucent, affect the visibility of pedestrians. Heads obscured by Translucent occluders such as tree leaves were annotated with the "ignore label" for tracking but are considered for evaluation of head detectors. Heads obscured by opaque occluders were neither considered for the evaluation of tracking nor detection and are annotated with visibility flag of "0". Assigning a visibility flag for a heads was left to the best discretion of annotators.

Key Frame Annotation: Due to the high frame rates (25 FPS) across videos, we employ keyframe annotation rule, with every 10^{th} frame considered a keyframe. Annotations were performed only on keyframes with a linear interpolation employed to annotate the positions of bounding boxes for the frames in between two successive keyframes. We used every 5^{th} frame to be a keyframe in sequences CroHD-03 and CroHD-13, where the pedestrian density and velocity are significantly higher than the other sequences, and parts of sequences where minor camera motion was incurred. Bounding boxes were adjusted in between keyframes for pedestrians in a particular frame if needed due to perceptible head motion. Once annotations were completed for a particular scene, two separate annotators reviewed the frames in between keyframes to supervise termination, initialization and occlusion handling of tracks.

Statistics: We analyze the detailed statistics of our benchmark in this section as summarized in Table 6. Specifically

[1] http://fiverr.com/
[2] https://github.com/opencv/cvat

we look into the statistics of our track length, pedestrian velocities, bounding box ratio, occlusions and class distribution. Average pedestrian velocity is the mean distance travelled by the tracks between each frame in pixels, averaged over the whole sequence and represented as $px.s^{-1}$. Bounding box ratio (BBR) denotes the ratio of spatial dimensions of frames to that of average bounding box in the respective sequence. Occlusion refers to the average time (in frames) that a target was annotated with a visibility flag of "0".

We compare CroHD with multiple pedestrian tracking benchmarks based on number of pedestrian annotations, pedestrian densities and tracks annotated as depicted in Table 1. The density in the table refers to the average number of pedestrian annotations per frame. CroHD has the largest pedestrian annotation, pedestrian density and number of tracks.

Dataset	Videos	Frames	Boxes	Density	Tracks
MOTChallenge-15 [8]	**22**	11,283	101,345	8.95	1221
MOTChallenge-16 [10]	14	11,235	292,733	25.8	1342
MOTChallenge-19[3]	9	**13,410**	2,259,143	171.0	3882
MOTS[14]	8	5,906	59,163	10.0	578
CroHD	9	11,463	**2,276,838**	**178.0**	**5230**

Table 1. Comparison between CroHD and existing multiple-pedestrian tracking benchmarks. Barring density, all the other columns refer to total figures for respective benchmarks.

2. Hyperparameter Tuning

In this section, we discuss the influence of hyperparameters for trackers which we used for baseline experiments on CroHD - IoU Tracker [2] and SORT [1]. For the two experiments, we used the detection provided by HeadHunter, to ensure fairness in evaluation.

2.1. IoU Tracker

We mainly study the influence of parameter $\sigma_{iou}, \sigma_h, ttl$ and t_{min}. The minimum IoU between two detection overlaps to be considered a track is denoted by σ_{iou}. Tracks are filtered if they do not contain at least one detection with an IoU $\geq \sigma_h$ for at least t_{min} frames. *ttl* denotes the number of frames through which visual tracking is performed backwards, with the Kernelized Correlation Filters (KCF) [5] applied for visual tracking. We observe no noticeable change with modification of parameters σ_h and *ttl*. We further attempted MedianFlow [6], TLD [7] as choices for visual tracking and no significant changes were observed with these modifications either. We hypothesize the size of objects being tracked as a reason for the observed invariance in performances. The results are summarized in Table 2. First row shows the performance of this tracker with all hyperparameters set to their default value. Better performance with respect to the identity metric are observed in the case of default t_{min} value while a lower t_{min} and higher σ_{iou} signifies a better MOTA score.

σ_{iou}	t_{min}	MOTA	IDEucl	IDF1
0.3	5	51.0	31.9	33.7
0.2	5	51.4	**32.6**	**34.1**
0.4	5	50.1	28.8	32.2
0.5	5	48.0	23.6	29.0
0.8	5	42.5	17.1	23.6
0.3	4	51.6	30.9	33.6
0.3	3	52.1	30.2	33.4
0.3	2	**52.4**	29.1	33.2

Table 2. Results of tuning V_IOU[2] tracker's hyper-parameters on the training set of CroHD.

2.2. SORT

We analyze three parameters corresponding to SORT [1], namely, max_age, min_hits and min_IoU. The maximum age a track will be kept alive without being associated to a detection is denoted by max_age. Without an associated detection, the position of tracks are updated through a Kalman Filter framework following Constant Velocity Assumption (CVA) for max_age frames. The minimum IoU required between subsequent detection of a particular track is denoted by min_IoU and min_hits denotes the number of minimum subsequent detection required to be associated to initialize a track. Table 3 summarizes the performance of SORT with varying hyperparameters. The first row corresponds to the default configuration while the last row denotes the best amongst the configurations we have varied. A straightforward observation is improvement with increasing max_age, more notably in-terms of IDEucl metrics. This is in contrast with what Bewely *et al.* [1] remark in their original paper. Furthermore, a significant improvement is also observed by reducing the min_IoU. These two occurrences can be explained due to significantly reduced overlaps between bounding boxes in tracking by head detection paradigm compared to tracking by full-body detection.

2.3. HeadHunter-T

We mainly analyze the impact of minimum confidence(or particle weights), λ_{reg}, required to keep a track alive. Table 4 shows the corresponding result. Surprisingly, lowering the λ_{reg} performs the best amongst the other values. We believe thresholding detection to 0.6 to be a possible reason behind this observation. Hence, we also analyze the effect of λ^{det}, the minimum confidence score to initialize a track with $\lambda^{det} = 0.8$ and $\lambda^{det} = 0.3$. A reduction in λ^{det} implied a mild deterioration in the identity preserving metrics, IDF1 and IDEucl. However, increas-

max_age	min_hits	min_IoU	MOTA	IDEucl	IDF1
1	3	0.3	41.1	28.4	30.3
1	3	0.2	41.2	28.4	30.3
1	3	0.4	41.0	28.2	30.0
15	3	0.3	43.2	54.1	44.9
30	3	0.3	43.3	57.8	46.6
15	1	0.3	50.6	52.7	48.3
30	1	0.3	50.8	56.5	50.5
1	1	0.3	46.8	27.3	30.5

Table 3. Results depicting fine-tuning hyperparameter of SORT[1] on the training set of CroHD.

ing λ^{det} showed a noticeable decline in performance. An increment in the either initialization threshold (λ^{det}) or regression threshold (λ_{reg}) produces monotonically decreasing performance results.

λ_{reg}	$\lambda^{det} = 0.3$			$\lambda^{det} = 0.6$			$\lambda^{det} = 0.8$		
	MOTA	IDEucl	IDF1	MOTA	IDEucl	IDF1	MOTA	IDEucl	IDF1
0.1	**64.9**	59.3	56.6	64.0	**61.5**	**58.5**	54.8	57.0	52.2
0.2	63.2	51.4	50.6	60.7	54.5	52.7	51.0	51.9	47.4
0.3	61.2	43.4	41.9	56.9	47.7	50.2	48.3	48.7	44.3
0.4	58.0	35.7	33.5	55.7	45.1	43.5	45.7	44.9	40.7
0.5	53.7	32.7	28.3	53.0	38.8	37.3	43.1	40.1	36.7
0.6	48.3	33.0	25.7	49.7	32.4	29.0	40.1	35.1	30.9

Table 4. Hyperparameter Fine-Tuning results of HeadHunter-T on the training set of CroHD.

2.4. Detection and Tracking

In this section, we analyze the tracking performances of various object detectors that were used for baseline experiments on head detection task of CroHD. Table 5 shows the object detectors upon whose output, the initialization of tracks in HeadHunter-T depends on. The tracking performances were evaluated on the training set of CroHD. These experiments were preformed analogous to Public Detection experiments on the standard MOTChallenge Benchmarks [3, 10]. Since the task of Face Detection is cognate to Head Detection, we used RetinaFace [15], a recent face detector which is the state-of-the-art method on WIDER FACE dataset. We used the implementation and model weights provided by the author. HeadHunter without Fine-Tuning on CroHD and without the Context Module are denoted as HeadHunter W/O FT and HeadHunter W/O Ctx respectively. For Headhunter W/O FT, we trained only on the training sets of CrowdHuman [13] and SCUT-HEAD dataset [11] . Barring RetinaFace and HeadHunter W/O FT, the remaining head detectors have been trained on CroHD.

Method	MOTA ↑	IDEucl ↑	IDF1 ↑	MT ↑	ML ↓	ID Sw. ↓
FRCNN[12]	46.0	37.8	36.1	140	111	12,178
FPN[9]	49.1	37.0	35.5	202	95	10,424
HeadHunter W/O Ctx	49.7	44.0	42.3	115	193	2,579
HeadHunter W/O FT	54.5	40.0	38.4	142	116.0	7,621
RetinaFace[4]	27.7	41.1	29.0	34.5	455	2,304
HeadHunter-T	**58.2**	**52.5**	**49.9**	**157**	**122**	**1941**

Table 5. Tracking performance comparison of HeadHunter-T on training set of CroHD with tracked initialized from various detectors.

Sequence Name	Avg Track Length (pixels)	Avg Track Duration (frames)	Avg Velocity ($px.s^{-1}$)	BBRR		Avg Occlusions (frames)	Instances per class			
				width	height		1	2	3	4
CroHD-01	593	244.3	61.7	1:41.7	1:82.2	11.8	79	4	2	0
CroHD-02	889	533.4	41.7	1:43.2	1:75.00	12.2	1,249	22	2	3
CroHD-03	1,322	318.1	103.9	1:33.1	1:63.4	25.7	809	0	0	2
CroHD-04	625	294.1	53.2	1:32.4	1:58.0	24.2	573	7	0	0
CroHD-11	613	270.0	56.8	1:36.6	1:79.7	16.9	120	9	2	2
CroHD-12	1,043	454.7	57.3	1:30.9	1:59.9	11.9	708	28	0	1
CroHD-13	922	351.7	65.5	1:32.7	1:68.0	53.3	731	2	1	0
CroHD-14	523	381.1	34.3	1:43.6	1:82.9	27.3	527	35	478	0
CroHD-15	919	389.6	59.0	1:32.9	1:84.6	25.8	256	61	1	3

Table 6. Detailed statistics of each sequence composing our dataset, CroHD. BBRR indicates bounding box to image ratio (in pixels). Classes correspond to 1:Pedestrian, 2:Static, 3:Ignore and 4:Person on Vehicle.

Sequence Name	Head Detection						Head Tracking							
	AP ↑	R ↑	F1 ↑	MODA ↑	MODP ↑	mAP_COCO ↑	MOTA ↑	IDF1 ↑	IDEucl ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDs ↓
CroHD-01	79.3	83.4	86.5	76.4	64.0	37.3	84.5	76.4	79.1	55	4	237	2,550	59
CroHD-02	40.4	52.9	61.1	50.0	38.6	9.1	66.7	66.4	60.0	548	127	46,479	168,299	2,049
CroHD-03	58.9	60.4	73.3	61.6	45.5	17.2	51.3	45.4	42.9	160	133	9,481	103,562	2,243
CroHD-04	64.6	70.0	76.9	65.7	51.5	20.3	53.6	52.7	47.9	135	98	9,438	61,238	975
CroHD-11	83.1	86.4	88.3	79.5	64.9	37.4	81.5	76.1	75.2	84	7	1,428	4,056	101
CroHD-12	34.8	51.0	58.6	42.1	37.2	10.2	60.6	64.3	57.1	264	64	21,851	100,484	1,173
CroHD-13	41.7	45.6	58.8	47.0	32.6	11.1	32.5	29.5	28.1	29	296	11,499	133,789	2,034
CroHD-14	45.8	62.3	67.5	43.1	46.7	16.0	67.3	61.2	59.4	215	60	11,506	48,580	817
CroHD-15	57.5	71.8	68.5	38.7	54.9	24.2	75.9	70.4	65.9	140	76	5,540	16,710	334

Table 7. Sequence-wise performances of HeadHunter and HeadHunter-T on CroHD.

Figure 1. An overview of annotated frames from our dataset, CroHD. In both train (left column) and test (right column) sets, bounding boxes of heads are either active (dark blue), static (orange), occluded (pink) or non-human (light blue). Occluders are present in many scenes, either opaque (green) or translucent (yellow).

Algorithm 1 HeadHunter-T

Require: Video $\mathcal{I}$ containing T frames $\{\mathcal{I}_\prime, \cdots, \mathcal{I}_{\mathcal{T}-\infty}\}$
Ensure: Trajectories $\mathcal{T} = \{\mathcal{T}_1, \cdots, \mathcal{T}_k\}$

$\mathcal{L}, \mathcal{T}, \mathcal{D} \leftarrow \phi$
for $t = 1, \cdots, T-1$ **do**
  $\mathbf{F}_t \leftarrow$ ExtractFeature$(\mathcal{I}_t)$
  for $l \in \mathcal{L}$ **do**
    if $l.\lambda^t > \lambda^{age}$ **then**
      $\mathcal{L}_t \leftarrow \mathcal{L}_t \setminus l$
    end if
    l.predict_cva()
  end for
  for $a \in \mathcal{T}$ **do**
    $\overline{\mathbf{p}}_t^a, \overline{\mathbf{w}}_t^a \leftarrow$ RoIPool$(\mathbf{F}_t, \overline{\mathbf{p}}_{t-1}^a$.predict()$)$
    if mean$(\overline{\mathbf{w}}_t^a) < \lambda^{\text{reg}}$ **then**
      $\mathcal{T} \leftarrow \mathcal{T} \setminus a$
      $\mathcal{L} \leftarrow \mathcal{L} \cup a$
    else
      $\mathcal{T} \cup a$
    end if
    if $\hat{\mathbf{N}}_{\text{eff}}^k > \hat{\mathbf{N}}_{\text{thresh}}$ **then**
      a.resample$(\hat{\mathbf{p}}_t^a)$
    end if
  end for
  $\mathcal{D}_t \leftarrow$ filter(RoIPool(RPN$(F_t)), \lambda^{\text{new}})$
  $\mathcal{D}_t \leftarrow \mathcal{D}_t \setminus$ filter(IoU$(\mathcal{D}_t, \mathcal{T}_t), \lambda^{\text{init}})$
  for $d \in \mathcal{D}_t$ **do**
    for $l \in \mathcal{L}$ **do**
      if cost_match$(l, d, \alpha, \beta) > \mathcal{C}$ **then**
        $\mathcal{L}_t \leftarrow \mathcal{L}_t \setminus l$
        $\mathcal{D}_t \leftarrow \mathcal{D}_t \setminus l$
        $\mathcal{T} \leftarrow \mathcal{T} \cup l$
        init_particles(l)
      end if
    end for
  end for
  for $d \in \mathcal{D}_t$ **do**
    $\mathcal{N} \leftarrow$ init_particles(init_track(d))
  end for
  $\mathcal{T} \leftarrow \mathcal{T} \cup \mathcal{N}$ & $\mathcal{N} \leftarrow \phi$
end for
return $\mathcal{T}$

References

[1] Alex Bewley, ZongYuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. *CoRR*, abs/1602.00763, 2016.

[2] E. Bochinski, T. Senst, and T. Sikora. Extending iou based multi-object tracking by visual information. In *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6, 2018.

[3] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taixé. CVPR19 tracking and detection challenge: How crowded can it get? *arXiv:1906.04567 [cs]*, June 2019. arXiv: 1906.04567.

[4] J. Deng, J. Guo, and S. Zafeiriou. Single-stage joint face detection and alignment. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1836–1839, 2019.

[5] João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *CoRR*, abs/1404.7584, 2014.

[6] Z. Kalal, K. Mikolajczyk, and J. Matas. Forward-backward error: Automatic detection of tracking failures. In *2010 20th International Conference on Pattern Recognition*, pages 2756–2759, 2010.

[7] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1409–1422, 2012.

[8] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv:1504.01942 [cs]*, Apr. 2015. arXiv: 1504.01942.

[9] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *CoRR*, abs/1612.03144, 2016.

[10] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. MOT16: A benchmark for multi-object tracking. *arXiv:1603.00831 [cs]*, Mar. 2016. arXiv: 1603.00831.

[11] Dezhi Peng, Zikai Sun, Zirong Chen, Zirui Cai, Lele Xie, and Lianwen Jin. Detecting heads using feature refine net and cascaded multi-scale architecture. *CoRR*, abs/1803.09256, 2018.

[12] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017.

[13] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018.

[14] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. MOTS: multi-object tracking and segmentation. *CoRR*, abs/1902.03604, 2019.

[15] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.