Multi-person pose tracking with occlusion solving using motion models

Lucas Gamez[1,2], Yusuke Yoshiyasu[1] and Eiichi Yoshida[1]

***Abstract*— We present a method for the multi-person human tracking problem including occlusion solving. To track and associate frame-by-frame human detections obtained using a deep learning approach, we propose to combine motion features extracted by optical flow and Kalman filtering, which allow us to predict the future poses of targets. By taking advantage of the characteristics of both motions features, we are able to handle sharp motions of the target and occlusions. With our simple occlusion handling mechanism, we achieve comparable results with state of the art and are able to keep track of a target identity even when occlusions occur.**

I. Introduction

Multi-person pose tracking is a basic yet challenging problem in computer vision, with a wide range of applications such as surveillance, biomechanics and robotics. This is an extension of the Multi Object Tracking (MOT) problem [1] to the tracking of human joints. Deep learning has proven to be one of the most accurate solutions in this field where tracking-by-detection approaches give us the current state-of-the-art results [2], [3], [4]. These approaches first detect persons in every frame and then match them between frames.

While these methods in general work well on a sequence with multiple people, they often fail when there are occlusions. These cases are very common in long sequences and are very challenging problems. The main reason why previous tracking approaches fail during occlusion is that they rely on short-term visual features. For example, state-of-the-art methods such as [4] uses optical flow to extract short term visual information. However short-term features are not reliable during occlusion due to the absence of visual information and often lead to tracking failures. Common approaches that handle occlusions aim to focus on a special feature to extract from detection and associate them to tracklets [5], usually based on visual appearance of the target extracted using re-identification (Re-ID) techniques [6], [7]. However, there is yet no ultimate solution to handle the occlusion problem in a robust way.

In this paper, we propose a simple method to solve multi-person pose tracking with occlusion olving. To keep the system simple, our framework is based on the simple baseline human pose tracker using optical flow [4] equipped with a Kalman filter based occlusion solver. Unlike [5] that predict the motions of bounding boxes, we apply Kalman filter on pose joints to predict their motions. This way, future poses can be predicted from the past to provide mid term motion information to the tracker, even during occlusions when visual information is absent. Consequently, we are able to keep track of an identity even after an occlusion as in Fig. 1.

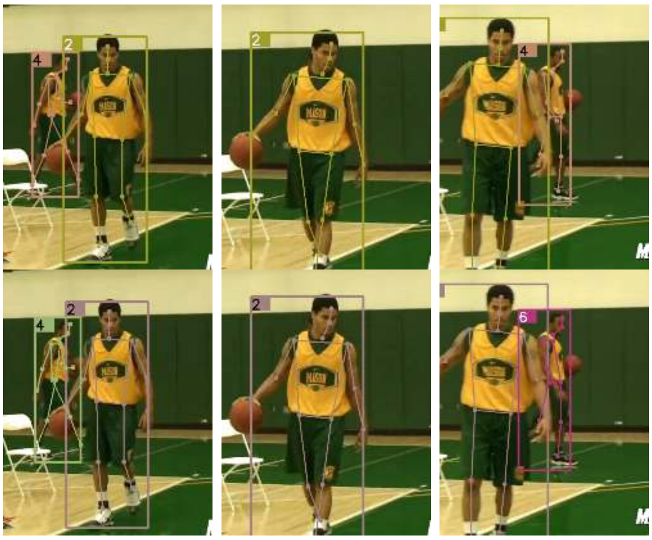

Fig. 1: Our method allows pose tracking while keeping the same unique identity number for a person even when this person completely disappear due to occlusion. **Top row**: Our tracking. **Bottom row**: basic flow tracking.

The contributions of this paper is summarized as follows:

- We propose a new combination of motion features to handle occlusion in multi-person pose tracking: visual motion features, which is the frame-to-frame motion computed using optical flow, and history motion features, which is the prediction of target trajectory computed using Kalman filtering.
- We propose an algorithm for associating detection and tracklet, called double greedy matching which is designed to prioritize visual motion features and can switch to the history motion features when occlusion occur. Leading to less identity loss during the tracking.

II. Related work

Tracking based on deep learning As in other image recognition tasks, deep-learning revolutionized multi-person person tracking and now deep-learning approaches obtains

[1] Lucas Gamez, Yusuke Yoshiyasu and Eiichi Yoshida are with CNRS-AIST JRL (Joint Robotics Laboratory) IRL, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan {lucas.gamez, yusuke-yoshisu, e.yoshida}@aist.go.jp,

[2] Lucas Gamez is with University of Montpellier, Montpellier, France

This work was partly supported by JSPS Kakenhi No. 18H03315 and No. 20K19836.

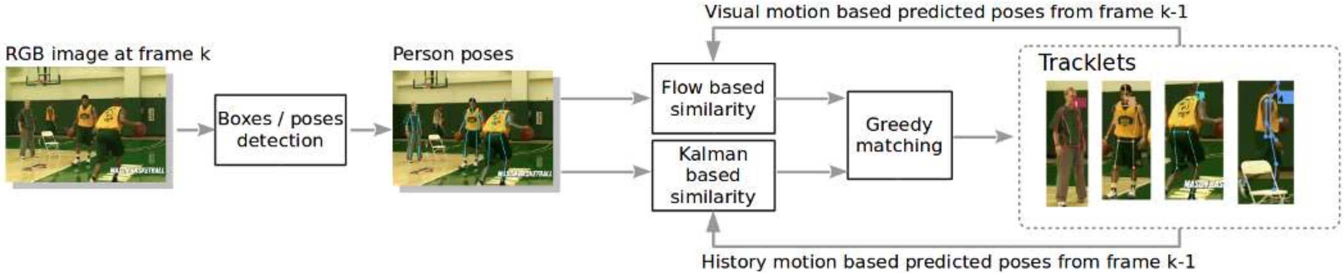

Fig. 2: The proposed pose tracking pipeline consists of pose estimation, pose prediction and data association. We separate the scoring process for the different prediction. The matching algorithm works in two step: first we use the score from visual motion prediction, and then process another greedy algorithm for the remaining poses using score from history motion prediction.

state-of-the-art results. Among them, the methods based on tracking-by-detection are the most widely used and give the best results [4], [2], [3]. Xiao et al. [4] uses a top-down method detection and compute optical flow to propagate the current pose to the next frame. On the other hand, Raaj et al. [8] directly detect heatmaps at joints without using bonding boxes. Connections between adjacent joints are then establish as heat map and affinity field for pose estimation, which is used to construct a spatio-temporal graph to solve the tracking problem. Other methods use bounding box regression to perform the tracking frame by frame [6] [9] [10] manually design occlusion pattern and train a joint detector robust to occlusion. Then the tracker minimizes a continuous energy function over all trajectories.

Tracking based on motion models Motion models have been used in multi-object tracking techniques [11], [12]. Among them, Kalman filter has been widely used to track bounding box of target in order to create more box candidates or to keep a track of a target identity [13], [14]. Optical flow and Kalman filter have been used together in [15] for complementing the smoothness and sharpness of the predictions for target's traveling trajectories. Instead, we are interested in using them for occlusion handling where optical flow is used whenever possible and it is switched to a Kalman filter model when there is no visual information to be extracted due to occlusions.

Tracking with re-identification networks The use of re-identification networks in tracking have also become a popular way to solve not just occlusion but long term tracking identity failure. These methods use a CNN to extract unique features from an individual and use these features to retrieve target identity [6], [7]

III. Tracking method

The overview of our system is depicted in Fig. 2. Our system is based on [4]. We first detect person bounding boxes. Then, we estimate the pose of each detected person one by one. In parallel, we compute predicted poses for the tracklet from the past frames using motion features. Finally, we pair each pose detected with one tracklet by comparing detected poses and predicted poses computed. Here, we use two different methods to find prediction: they are flow-based prediction computed with visual motion features and Kalman based prediction computed with history motion features. At each step, box detection, pose estimation and paring, are done step-by-step.

The key idea we present is to use occlusion dedicated features only when we need them. Using the optical flow for frame by frame tracking show great result, especially when tracking specifics points in the images, here, keypoints of human pose. It can handle the tracking of very sharp and unpredictable motion that are often present in video sequences. But as we use predictions based on optical flow, we need constant visual information to update accurately a pose. In case of occlusion, this visual information is not available anymore and we can not rely on the optical flow to follow the target. To solve this problem, we choose to use Kalman filter to model the motion of a target and skip the update step when needed, such as during occlusion, to handle inconsistent or missing information about the target.

A. Person detection and human pose detection

We use two networks for bounding box detection and pose estimation. Given a new frame, we use a Faster RCNN network [16] with FPN [17] backbone as the bounding box detector. For pose estimation, we use the network model presented in [4]. This network consist of a ResNet followed by deconvolution layers. The detectors produce the pose detection results for frame k as follows:

$$D^k = [D_i^k]_{i=1::P^k}$$

$$D_i^k = [J_n]_{1::N}$$

Here D^k is the list of the P^k detected person poses at frame I^k. D_i^k is the ith pose in frame I^k, which is composed of N joints, J_n.

B. Predicted Poses

We predict the tracklet poses from the past frames to perform matching with the detected poses from the current frame. Here, we use two different methods to find prediction: flow-based prediction and Kalman based prediction.

Optical flow based prediction These prediction will be used as the default method for tracking people when the target is visible.

To obtain optical flow based prediction $\hat{D}^{\text{flow}}_{(k,i)}$, we first compute the optical flow $F_{k-1\to k}$ between frames I^{k-1} and I^{k} using FlowNet2. Then, the prediction $\hat{D}^{\text{flow}}_{(k,i)}$ in the frame I^{k} is obtained by propagating every joint J_n in D^{k-1}_{i} according to $F_{k-1\to k}$:

$$\hat{J}^{\text{flow}}_{n} = \begin{bmatrix} u_n + \delta u \\ v_n + \delta v \end{bmatrix}$$

Where (u_n, v_n) are the coordinates of the keypoint $\mathbf{J_n}$ and δu and δv are the value of $F_{k-1\to k}$ at the location (u_n, v_n). In the case where the last detection available of one tracklet is not from the very previous frame, i.e. last detection is D^{l}_{i} with $k - l > 1$, the detection's joints will be propagate through all the flow that separate the frame $I^{k} - l$ from the frame I^{k}, that are $[F_{l\to k}, F_{l\to l+1}, F_{l+1\to l+2}, ..., F_{k-1\to k}]$. To avoid propagation to the background, we set a max value L for l.

Kalman filter based prediction Because flow based prediction need constant visual information about the target, it will inevitably fail in case of occlusion. To predict new poses of targets even when visual information is not accessible, we use Kalman filter that is able to give a prediction for a given step even without providing update data. At a frame k, for each tracklet $T_p = ([D^{x}_{y}], id)$, $\hat{D}^{\text{kalman}}_{(k,p)} = [\hat{J}^{\text{Kalman}}_{n}]_{1::N}$ is the Kalman filter based predicted pose of this tracklet. The prediction is obtained by applying a Kalman filter to each keypoint of the detected pose individually. For each keypoint, we assume its speed to be constant and composed of two value: horizontal speed and vertical speed. More formally, we give the state X and the 4×4 transition matrix $\mathbf{A}$ for one keypoint of the pose:

$$\hat{\mathbf{x}}_k = \begin{bmatrix} \hat{u}_k \\ \hat{v}_k \\ \dot{\hat{u}}_k \\ \dot{\hat{v}}_k \end{bmatrix}, \mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $[\hat{u}_k, \hat{v}_k]$ and $[\dot{\hat{u}}_k, \dot{\hat{v}}_k]$ are respectively the coordinates of the keypoint and their speed. And Δt is the time between two frames. The predicted coordinates $\hat{J}$ of the keypoint at frame k is computed as:

$$\hat{J}^{\text{Kalman}}_{n} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} (\mathbf{A}\hat{\mathbf{x}}_{k-1})$$

In case of occlusion during some frames, we omit the update step of the filtering and compute directly the prediction step. Same as optical flow based prediction, we set a maximum number L of update step that we skip before stopping the tracking of an identity. This kind of filtering is simple yet effective for cases where people pass in front each other.

C. Similarity computation

In order to compute the similarity score between detected poses and tracklets, we compare the poses using Object Keypoint Similarity (OKS) [4]. Comparing a single detected pose to a tracklet will be computing the OKS metric between this detected pose and the two predicted poses of the tracklet. By doing so, we obtain two scores, flow-based similarity and Kalman-based similarity. We store these scores in a $d \times p \times 2$ tensor where d is the number of detection and p is the number of prediction.

TABLE I: Result for the training of the pose estimation network on Posetrack 2018 validation dataset.

AP	AP .5	AP .75	AR	AR .5	AR .75
0.706	0.877	0.780	0.731	0.888	0.792

From this step, we have a batch of detection from the poses detected $D^{k} = [D^{k}_{1}, \ldots D^{k}_{N}]$, which is accurate, more realistic but without correspondence to a track. We also have a batch of prediction pair (from optical flow and Kalman filter) $[(\hat{D}^{\text{flow}}_{(k,1)}, \hat{D}^{\text{kalman}}_{(k,1)}) \ldots (\hat{D}^{\text{flow}}_{(k,P^k)}, \hat{D}^{\text{kalman}}_{(k,P^k)})]$, less accurate but with correspondence to an identity. In addition, we have a scoring matrix containing the similarities between detected poses and predictions.

D. Pairing with double greedy matching

Flow-based predictions and Kalman-based prediction are complementary, so are their associated similarity scores. The optical flow allows us to generate accurate predictions even during jerky motion but will fail to give any usable information during occlusion. On the other hand, the Kalman filter gives useful information to recover targets from occlusion but will have trouble to predict accurately a pose if the target keep having jerky motion. Due to this complementary characteristics, the two similarity scores can show opposite roles depending on the event occurring, i.e. abrupt change of direction or occlusion. For this reason, we chose not to fuse the scores and use them separately instead for the pairing step. Specifically, we use greedy matching algorithm twice on the scoring matrix to solve the paring problem. A first matching using flow based similarity scores is done for the matching. Then, if the algorithm stop because of the threshold limit (meaning there is still detection *and* prediction unpaired), a second matching is done using Kalman based similarity scores. Remaining detection after the matching are considered as new tracklet and a new identity is created.

IV. Experiments

A. Dataset

The whole method has been tested on the Posetrack 2018 dataset [19]. This dataset provides body joint and tracking id annotations for image sequences of multi person video. It contains 593 videos with 30 annotated frames for the training data and 74 annotated videos for the validation data. We also used the MOT17 dataset [20]. It contains a total of 14 videos of 450 to 1500 frames. Each frame is annotated with person bounding boxes and identity numbers. This dataset does not provide pose annotations but is filled with longer video with more occlusion cases.

TABLE II: Result for pose tracking on Posetrack validation dataset.

	MOTA							
Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	**Tot.**
MSRA-50 [4]	72.1	74.0	61.2	53.4	62.4	61.6	50.7	62.9
MSRA-152 [4]	73.9	86.9	63.7	56.1	65.5	65.1	53.5	65.4
Lighttrack [18]	67.7	72.6	67.3	57.8	63.5	63.8	57.7	64.6
Baseline	63.2	68.2	64.1	55.0	61.0	61.2	55.1	61.2
Our	67.2	72.1	67.0	57.4	63.3	63.6	57.5	64.2

TABLE III: Result for Multiple Object Tracking on MOT17 dataset.

Method	IDF1	IDP	IDR	IDs	MOTA
Baseline	43.3%	64.5%	32.6%	2794	46.6
Our	48.4%	72.1%	36.4%	1044	48.2

B. Implementation

As said previously, we use Faster R-CNN to detect bounding box from an image. We use the pretrained weights provided from [16] to process the Posetrack dataset images. For the MOT17 dataset, we use the Faster R-CNN public detection as bounding boxes. We trained the pose estimation network on the Posetrack 2018 train dataset. We report in Table I add table the performance of the network on the Posetrack 2018 validate dataset. The flow between two frames is computed using Flownet2 [21] trained on the flying chair dataset [22]. For all of our implementation, we set the value of L at 20.

C. Baselines

For additional studies on our method, we developed two versions of our method. We will refer to the first one as the baseline, it is a basic tracking using only the optical flow to predict future poses. The second one is the complete method explain in this paper, using both flow based prediction and Kalman filter based prediction. We also compare our method with great performing multi-person trackers. MSRA tracker [4] is the winner of PoseTrack ECCV 2018 Challenge for Multi-Person Pose Tracking. It propagates poses using optical flow and uses this propagation to generate additional bounding box candidates. Lighttrack [18] is another tracker that uses a re-identification network on person poses to retrieve identity of lost targets.

D. Metrics

MOTA metric is a metric widely used for tracking evaluation and focuses on three kinds of error: misses, false positives and identification switch. It computed as follows:

$$MOTA = \frac{FP + M + MM}{N}$$

Where FP, M and MM are respectively the number of false positive, misses and mismatches over a sequence. N is the number of ground truth identities appearances. In MOTA, the proportion of the errors in the final score is not even and the contribution of the identification switch is small compared to the two others. For this reason, we also use IDF1 metric [23] to evaluate our method. IDF1 metric aims to minimize the number of false positive and false negative errors during the matching between detection and ground truth. In other word, the MOTA metric evaluates how often the tracker fails to identify the target, and the IDF1 metric evaluates for how long the tracker is able to track the target. IDP and IDR are intermediate value obtained during IDF1 computation that designate respectively the identification precision and the identification recall. We also report for MOT dataset the number of identity switch over all the dataset sequences.

E. Results

Qualitative results are shown in Fig. 3, each sequence contain 3 frames representing the 3 phases of an occlusion case present in the dataset: before the occlusion, during the occlusion when the target is not visible and after the occlusion when the id is reassigned to the target. For each sequence, the top row is the result obtained with the baseline and the bottom row is obtained by our method. These results show how our tracker is able to handle occlusion when the baseline systematically change the identification of a target after an occlusion. These cases are not rare in the dataset. The average identification number given by the baseline in Fig. 3 is higher than the one from our method. Which means that many identities have been falsely generated by the baseline, whereas our tracker was able to keep the same identification for the targets.

We report the result of our method on the dataset in Table II. While our method does not outperform state of the art, having a dedicated part for occlusion improve the original performance of the tracker. But in person pose tracking, the pose estimation take an important place in the accuracy.

We report in Table III the result from MOT17 dataset [20] which is only focused on the Multiple Object Tracking task. While MOTA is slightly improved, results in IDF1 and identification switch (IDs) show clear improvement compare to the baseline. These numbers show that, despite not generating significantly less error (in term of MOTA metric), our method can keep a track of an identity for a longer period of time.

Limitations While we design our tracker in order to have the limitations of optical flow balanced by the Kalman filter,

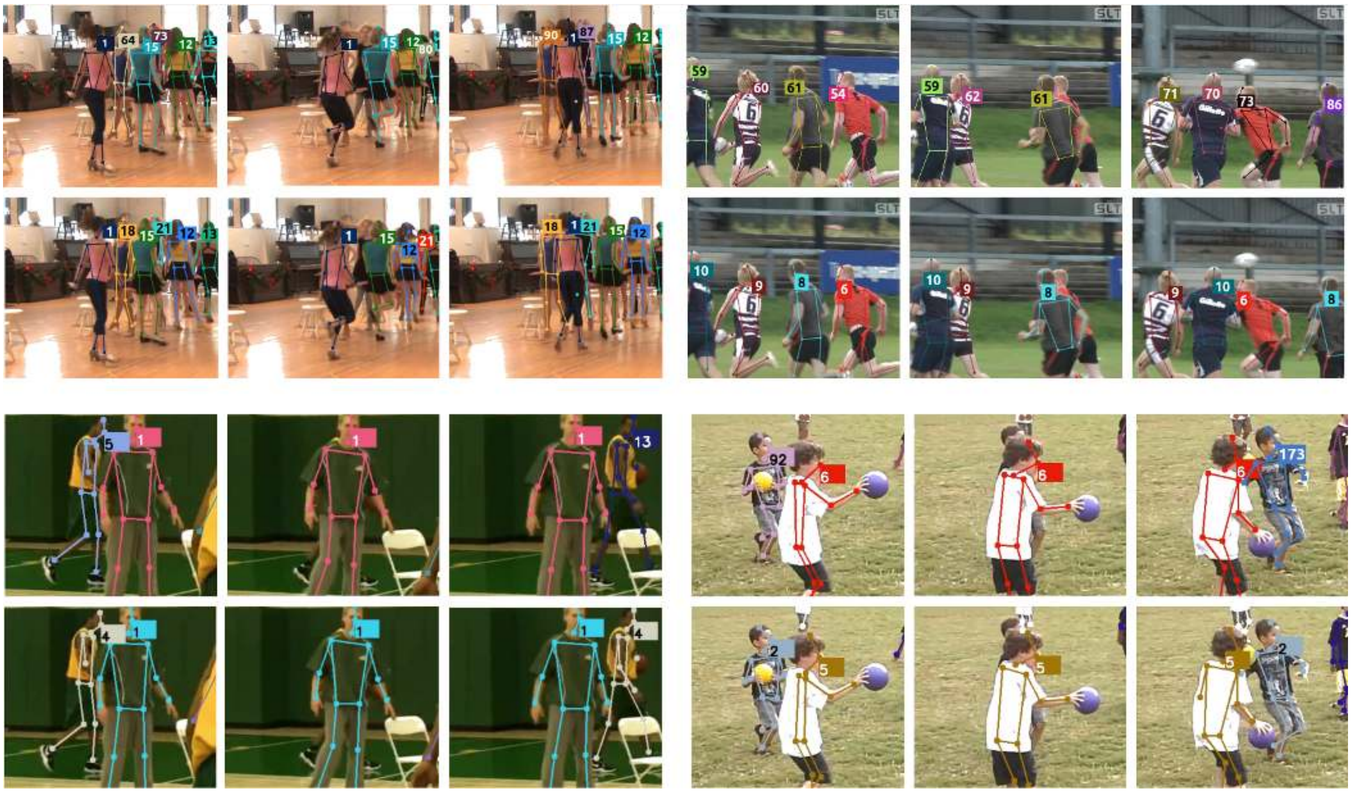

Fig. 3: Qualitative results of our method compare to our baseline on occlusion cases present on the validation set of PoseTrack dataset. For each of the 4 sequences, the top row is the result from the baseline and the bottom row is the result from our method.

and the ones of Kalman filter balanced by the optical flow, our tracker still have some limits. These limits are inherent to the nature of our motion models. We chose a linear model to track people during occlusion, thus preventing us to retrieve targets when the target change direction during an occlusion. For the same reason, long occlusions are also challenging to solve. Even if the real motion of the target is linear, motion models tend to drift if no information updates are provided.

V. Conclusion

We presented a multiple person pose tracking method that uses two different temporal information to predict targets pose and match poses through frames. By using the two motion models to predict poses, we are able to take advantage of their strengths and complement each other: optical flow can deal with changes in speed of the target but fails during occlusion cases; Kalman filter can handle occlusions but struggles to follow sharp motions. This idea allows our tracker to be more robust to identity loss for short term tracking, especially during occlusion when people cross each other. In future work, we would like to tackle the limitations this work by using it as a baseline to address longer-term tracking challenges using such as re-identification.

References

[1] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera, "Deep learning in video multi-object tracking: A survey," *Neurocomputing*, vol. 381, pp. 61–88, Mar 2020.

[2] Dongdong Yu, Kai Su, Jia Sun, and Changhu Wang, *Multi-person Pose Estimation for Pose Tracking with Enhanced Cascaded Pyramid Network: Subvolume B*, pp. 221–226, 01 2019.

[3] Guanghan Ning, Ping Liu, Xiaochuan Fan, and Chi Zhang, "A top-down approach to articulated human pose estimation and tracking," .

[4] Bin Xiao, Haiping Wu, and Yichen Wei, "Simple baselines for human pose estimation and tracking," in *European Conference on Computer Vision (ECCV)*, 2018.

[5] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, USA, Sept. 2016, pp. 3464–3468, IEEE.

[6] Weitao Feng, Zhihao Hu, Wei Wu, Junjie Yan, and Wanli Ouyang, "Multi-object tracking with multiple cues and switcher-aware classification," .

[7] Liqian Ma, Siyu Tang, Michael J Black, and Luc Van Gool, "Customized multi-person tracker," p. 16.

[8] Yaadhav Raaj, Haroon Idrees, Gines Hidalgo, and Yaser Sheikh, "Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields," 11 2018.

[9] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixé, "Tracking without bells and whistles," *CoRR*, vol. abs/1903.05625, 2019.

[10] Siyu Tang, Mykhaylo Andriluka, Anton Milan, Konrad Schindler, Stefan Roth, and Bernt Schiele, "Learning people detectors for tracking in crowded scenes," 12 2013, pp. 1049–1056.

[11] Ju Hong Yoon, Ming-Hsuan Yang, Jongwoo Lim, and Kuk-Jin Yoon, "Bayesian Multi-object Tracking Using Motion Context from Multiple Objects," in *2015 IEEE Winter Conference on Applications of Computer Vision*, Waikoloa, HI, USA, Jan. 2015, pp. 33–40, IEEE.

[12] Caglayan Dicle, Octavia I. Camps, and Mario Sznaier, "The Way They Move: Tracking Multiple Targets with Similar Appearance," in

2013 IEEE International Conference on Computer Vision, Sydney, Australia, Dec. 2013, pp. 2304–2311, IEEE.

[13] Long Chen, Haizhou Ai, Zijie Zhuang, and Chong Shang, "Real-time multiple people tracking with deeply learned candidate selection and person re-identification," .

[14] Nicolai Wojke, Alex Bewley, and Dietrich Paulus, "Simple online and realtime tracking with a deep association metric," *CoRR*, vol. abs/1703.07402, 2017.

[15] Yuichi Motai, Sumit Kumar Jha, and Daniel Kruse, "Human tracking from a mobile agent: Optical flow and kalman filter arbitration," *Sig. Proc.: Image Comm.*, vol. 27, pp. 83–95, 01 2012.

[16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *arXiv:1506.01497 [cs]*, June 2015, arXiv: 1506.01497.

[17] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, "Feature Pyramid Networks for Object Detection," *arXiv:1612.03144 [cs]*, Apr. 2017, arXiv: 1612.03144.

[18] Guanghan Ning and Heng Huang, "Lighttrack: A generic framework for online top-down human pose tracking," *CoRR*, vol. abs/1905.02822, 2019.

[19] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele, "PoseTrack: A benchmark for human pose estimation and tracking," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5167–5176, IEEE.

[20] Anton Milan, Laura Leal-Taixé, Ian D. Reid, Stefan Roth, and Konrad Schindler, "MOT16: A benchmark for multi-object tracking," *CoRR*, vol. abs/1603.00831, 2016.

[21] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.

[22] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[23] Ergys Ristani, Francesco Solera, Roger S. Zou, Rita Cucchiara, and Carlo Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," *CoRR*, vol. abs/1609.01775, 2016.