Dynamic Multi-Loss Weighting for Multiple People Tracking in Video Surveillance Systems

Xuan-Thuy Vo, Tien-Dat Tran, Duy-Linh Nguyen and Kang-Hyun Jo
Department of Electrical, Electronic and Computer Engineering,
University of Ulsan
Ulsan (44610), Korea
Email: {xthuy, tdat}@islab.ulsan.ac.kr; ndlinh301@mail.ulsan.ac.kr; acejo@ulsan.ac.kr

2021 IEEE 19th International Conference on Industrial Informatics (INDIN) | 978-1-7281-4395-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/INDIN45523.2021.9557515

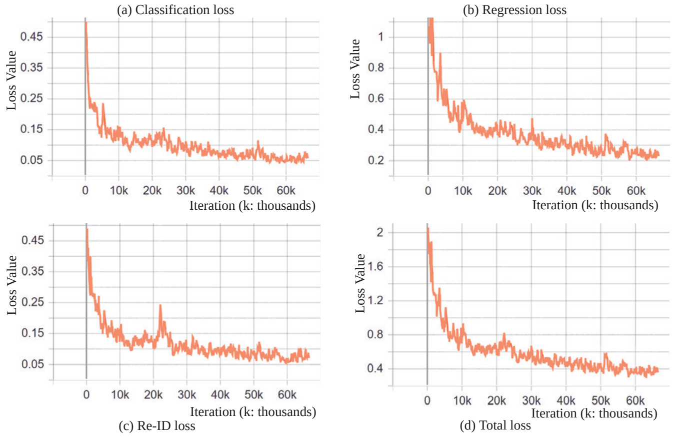
***Abstract*—Multiple people tracking is a fundamental yet challenging task in the computer vision field, which served as a primary process for high-level tasks such as human behaviors, action recognition, pose estimation. Person tracking is decomposed into detection and re-identification (re-ID) sub-tasks. Conventionally, the detection learns classification and regression objectives simultaneously; and the re-ID sub-task is treated as a classification task. Therefore, person tracking is multiple task learning corresponding to multiple loss functions (multiple objectives) with one bounding box regression and two classifications. The difference between various tasks is as follows: the ranges of each objective are inconsistent, the contribution of each task to the overall gradient is altered, and the learning pace of each task is different (level of difficulty). It leads to an objective imbalance in multi-task learning. Previous methods proposed weighting factors as new hyper-parameters to balance the ranges of each task. The dimension of search space for manually tuning these hyper-parameters is high, which depends on the number of tasks. Accordingly, selecting reasonable weighting factors is difficult and complicated. This paper introduces dynamic multi-loss weighting (DMW) with simple but effective in which the weighting factors are dynamically changed during training without introducing any hyper-parameters. The dynamic weights are optimized to balance regression and classification objectives, which depend on the difficulty level of each task and the correlation between each task. Additionally, the general convolution operations are spatially invariant to some degree, which hinders the network's performance. Hence, this work employs the position-sensitive operation improving feature extraction. The proposed method is conducted on the MOT17 challenging benchmark, which outperforms the online multiple people trackers without using additional data.**

***Index Terms*—Dynamic multi-loss weighting, position-sensitive operation, multiple people tracking, surveillance systems**

I. Introduction

Multiple people tracking is the basic task for understanding person in visual data such as images and videos. The input of multiple people tracking is a consecutive video, separated into discrete frames at 30 FPS (frames per second). These frames are considered as images to be forwarded into the network. The output of multiple people tracking is the detection results predicting classification scores and localization (offsets) of each person in all frames, re-ID scores are used to associate person detection over time-domain (predicting trajectories). Multiple people tracking has been widely used in many scene understanding applications such as intelligent surveillance systems, robotics, self-driving vehicles.

Recently, the deep learning technique achieved great performance in the image classification, object detection, and object segmentation, which brought significant improvement in solving multiple people tracking tasks. Many trackers applied tracking-by-detection methods only paying attention to the data association network for predicting trajectories. It means the detection results created are available in all frames and they match detection across frames according to re-ID score or classic methods, e.g., Kalman filter and Hungarian algorithm. However, the data association is directly affected by detection performance. Separating detection and data association into different networks hampers the overall performance. Hence, this paper uses the single network predicting detection and re-ID score according to task-dependent.



(a) Classification loss

(b) Regression loss

(c) Re-ID loss

(d) Total loss

Fig. 1. The learning paces of regression, classification, and re-ID tasks with the proposed network. The weighting factors are assigned equally.

Multiple people tracking is multi-task learning (multi-loss functions) including detection and re-ID task. The detection minimizes classification and regression objectives simultaneously. The re-ID task is treated as a classification task. Accordingly, person tracking solves one bounding box regression and two classification tasks. However, various tasks have different properties: (1) The range of each objective are inconsistent since the regression task takes input in logarithmic transformation with range $[0, \infty)$ while classification task uses softmax or sigmoid function to normalize scores between intervals $[0, 1]$, (2) The contribution of each task to the overall gradient is different because the gradient norm of each task is different, (3) The level of difficulty of each task is different because the regression task focuses on accurate bounding box localization of foreground samples, the classification learns semantic features to classify person and background, and the re-ID task learn appearance features to distinguish two people or person versus distractor. Therefore, it leads to the objective imbalance in multi-task learning. For instance, we plotted the learning paces of the tracking network during training on the MOT17 [1] dataset in Fig. 1. The classification loss values are always smaller than the regression loss values. Additionally, classification loss converges faster than the regression loss. Usually, multiple losses are accumulated into the single total loss by using a linear weighted sum of losses. Previous trackers such as CTracker [2], CenterTrack [3] and detectors that are RetinaNet [4], FCOS [5], Faster R-CNN [6] proposed weighting factors as extra hyper-parameters to multi-loss learning. These weights tuning by hand are difficult and complicated because the dimension of search space is high for multiple tasks. Moreover, balancing losses by linear weighting factors as task-independent will hinder the overall performance since regression and classification tasks have a positive correlation. During training, propagating gradient signals to the network of easy samples and hard samples are different at each epochs. Hence, weighting factors are fixed during training in a straightforward way. This paper proposes dynamic multi-loss weighting (DMW) without introducing any hyper-parameter by investigating the correlation between regression and classification task (task-dependent), and the difficulty level of each task. These weighting factors are dynamically changed during training.

The normal convolution operations are spatially invariant, which are suitable for several tasks such as image classification, object segmentation. The regression task leverages the spatially variant property to predict person coordinate precisely. Inspired by CoordConv [7], this paper applies position-sensitive operation for embedding pixel coordinates into feature map.

The proposed method is implemented on the MOT17 challenging benchmark to evaluate the effectiveness of the DMW approach and position-sensitive operation in the one-shot tracker, which surpasses the online state-of-the-art tracker without using extra data.

II. Related Works

Multiple people tracking. Multiple people tracking is classified into the online method and offline method. The online tracking utilizes the last frames and current frames as input, thus avoiding high model complexity. While the offline tracking takes the last frames, current frames, and future frames as input of the network. Although offline method achieves high performance due to leveraging motion extraction and optical flow, it is still high computational cost. The online tracking in [8], [9], includes detection and data association step. Since, MOT challenging [1] provided detection results generated by detectors such as DPM, Faster R-CNN [6], the most of tracking method focuses on data association. JDE [10], CTracker [2], CenterTrack [3], and FairMOT [11] joined detection and re-ID into single end-to-end network, which employs re-detection to improve data association task. This paper uses joined network as the baseline.

Person Detection. Many popular detectors such as RetinaNet [4], FCOS [5], and Faster R-CNN [6] is the generic detection for detecting many categories in the scene, which is potential source to specific category such person, car, etc. This work applies RetinaNet [4] for detection step due to simplicity of it.

Multi-loss weighting. Multi-task weighting [12] introduced uncertainty weighting for scene geometry and semantics, which investigates homoscedastic of the regression task and classification task. JDE [10] and FairMOT [11] applied the uncertainty perspective for balancing multiple losses of tracking task. SWN [13] proposed sample weighting networks embedded into object detection network to predict the weight for each sample according to the difficulty level of each sample. Specifically, SWN used several fully connected layers with the input of classification loss, regression loss, IoU score, and classification score to learn sample weight (samples' uncertainty).

Coordinate Convolution. CoordConv [7] proposed coordinate convolution with simple operation concatenated to feature map in which convolutional kernel learns the coordinates (x, y) of the input data. SOLO [14] inherited the CoordConv operation to improve localized feature map learning for instance segmentation task. This paper also inspires this operator for detection and re-ID task in multiple people tracking systems.

III. The Proposed Method

The online single end-to-end network is described in Fig. 2. The input of the network is the adjacent frames $\{\mathrm{F}_{t-2}, \mathrm{F}_{t-1}, \mathrm{F}_t\}$ as the tracklet. The backbone network is used to extract informative features. In this paper, ResNet-50 pre-trained on ImageNet is employed as feature extraction. This backbone consists of five stages {S1, S2, S3, S4, S5} corresponding to five down-scaled times to reduce the spatial resolution. Following common methods such as RetinaNet [4], FCOS [5], BNLNet [15], and EFPN [16], we only selects feature maps {S3, S4, S5} to construct the feature pyramid {P3, P4, P5, P6, P7} treating scale imbalance in detection task. Each tracking head includes classification, regression, and re-ID branch. Each branch consists of four convolutional layers in which each layer contains 3×3 convolution + Group Normalization + ReLU activation function, and one 3×3 convolution operation that the output channel dimension is suitable to the specific task, e.g., channel dimension is 2A for the classification branch, 8A for the regression branch, and A for re-ID branch (A is the number of anchor boxes tiled on each location of the input frame). The DMW performs a weighted sum of losses as a linear operation, which will discuss in section III-A. Note that position-sensitive operation works as normalized pixel coordinates, will describe in subsection III-B.

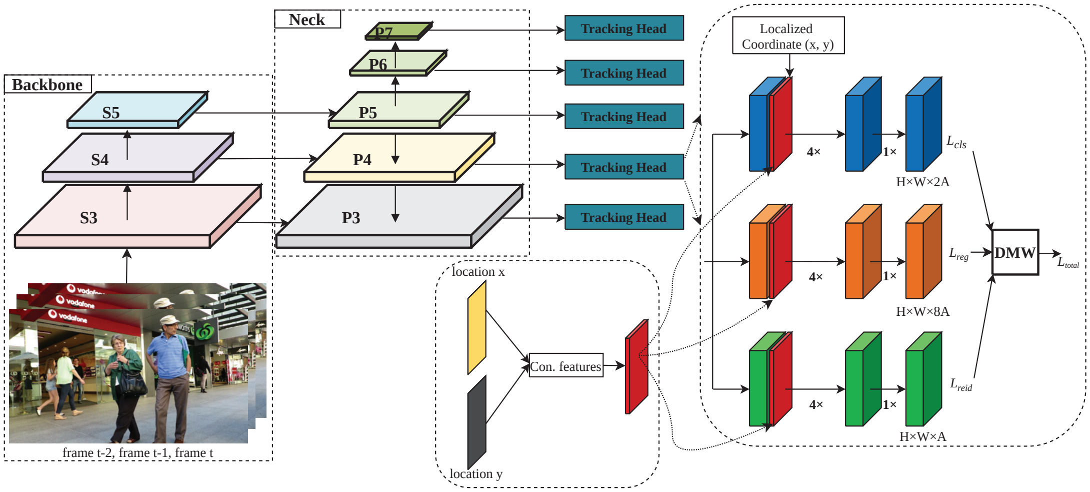


Fig. 2. The overall architecture consists of three parts: backbone network, neck, and tracking head. The backbone network served as feature extraction takes the current frame and past frame as input. The neck part forming a pyramid feature {P3, P4, P5, P6, P7} takes three feature maps {S3, S4, S5} from stage 3, stage 4, and stage 5 of the backbone network to gather low-level and high-level feature. Each tracking head includes three branches: classification branch, regression branch, and re-ID branch. 4× denotes four convolutional layers in which each layer contains one 3×3 convolution operation following by Group Normalization and ReLU activation function. 1× denotes one convolution operation without normalization and activation function. H, W is the height and width of the feature map. A indicates the number of anchor boxes per location. L_{cls}, L_{reg}, L_{reid}, L_{total} are classification, regression, re-ID loss, and total loss, respectively. DMW is the dynamic multi-loss weighting function.

A. Dynamic Multi-Loss Weighting (DMW)

As shown in Fig. 2, three losses corresponding to three tasks are accumulated by Dynamic Multi-Loss Weighting (DMW) operation. In normal way, the classification loss L_{cls} is Focal loss [4], defined as:

$$L_{cls} = \frac{1}{N_{pos}} \sum_{i=1}^{N_{pos}} FL(p_i, \hat{p}_i), \tag{1}$$

$$FL(p_i, \hat{p}_i) = -a(1-p_i)^b \log(\hat{p}_i), \tag{2}$$

where N_{pos} is number of positive sample. p_i, $\hat{p}_i$ is classification score, class label respectively. $FL(p_i, \hat{p}_i)$ is the Focal loss in which a, b are balanced variant, modulating factor, respectively. The range of the classification score is constrained by intervals $[0, 1]$ due to that:

$$p_i = \delta(s_i) = \frac{1}{1+e^{-s_i}}, \tag{3}$$

where δ is sigmoid function normalizing digit score s_i to get probability of each class.

In this paper, the regression loss L_{reg} is $smooth_{L1}$ loss [6], computed as:

$$L_{reg} = \frac{1}{N_{pos}} \sum_{i=1}^{N_{pos}} \sum_{j}^{\Delta} smooth_{L1}(o_{i,j} - \hat{o}_{i,j}), \tag{4}$$

where N_{pos} is number of positive samples. $o_{i,j}$, $\hat{o}_{i,j}$ is the offset prediction and transformed ground truth bounding box. $\Delta = \{\Delta x_i, \Delta y_i, \Delta w_i, \Delta h_i\}$ is the transformed coordinates (center (x, y), width, and height of the bounding box) by logarithmic algorithm, computed as:

$$\begin{aligned} \Delta x_i &= (x_i - x_{i,a})/w_{i,a}, & \Delta y_i &= (y_i - y_{i,a})/h_{i,a}, \\ \Delta w_i &= \log(w_i/w_{i,a}), & \Delta h_i &= \log(h_i/h_{i,a}), \end{aligned} \tag{5}$$

where $\{x_i, y_i, w_i, h_i\}$ is the offset prediction (center, width, height of bounding box i). $\{x_{i,a}, y_{i,a}, w_{i,a}, h_{i,a}\}$ is the coordinates of positive anchor box i. As shown in Equation 5, the center $(\Delta x_i, \Delta y_i)$ of the bounding box is still in real number and $(\Delta w_i, \Delta w_i)$ is transformed by $\log$ function. It means the range of bounding box prediction belongs to $[0, \infty)$.

Similar with CTracker [2], the re-ID loss L_{reid} is Focal loss [4], defined as:

$$L_{reid} = \frac{1}{N_{pos}} \sum_{i=1}^{N_{pos}} FL(id_i, \hat{id_i}), \tag{6}$$

where id_i, $\hat{id_i}$ is identification score, truth label according to IoU score (Intersection of Union). Since re-ID loss is the classification loss, the range of it is $[0, 1]$.

Finally, the total loss L_{total} is the weighted sum of losses:

$$L_{total} = \alpha L_{cls} + \beta L_{reg} + \gamma L_{reid}, \tag{7}$$

where α, β, and γ are the weighting factors to balance the range of losses.

In the conventional approach, the weighting factors are tuned by many experiments to select optimal values. It takes many days for each implementation. For instance, Table I shows the various selection of weighting factors. Note that the search space with three tasks is large. During training, the weighting factors are fixed, thus the network can not determine the difficulty level of each sample to learn the model in an efficient way.

TABLE I
SEVERAL EXPERIMENTS WITH THE FIXED WEIGHTING FACTORS ON THE MOT17 VALIDATION SET.

α	β	γ	MOTA↑	IDF1↑	MOTP↑	MT↑	FP↓
0.5	1.0	1.5	74.8	65.8	85.2	269	1972
0.5	1.5	1.0	74.8	65.2	85.6	260	2045
0.5	2.0	1.0	74.3	65.2	85.5	261	2057
1.0	1.0	1.0	73.1	63.3	85.3	245	2229
1.0	0.5	1.5	74.9	65.3	84.9	273	1922

To avoid sub-optimal weighting factors, this paper introduces dynamic multi-loss weighting (DMW) operation. The weighting factors in Equation 7 is computed as:

$$\begin{aligned}\alpha &= \frac{L_{reg} + L_{reid}}{L_{cls} + L_{reg} + L_{reid}},\\ \beta &= \frac{L_{cls} + L_{reid}}{L_{cls} + L_{reg} + L_{reid}},\\ \gamma &= \frac{L_{cls} + L_{reg}}{L_{cls} + L_{reg} + L_{reid}}.\end{aligned} \tag{8}$$

Note that the L_{cls}, L_{reg}, and L_{reid} are the loss computed for each sample. Therefore, the weighting factors α, β, and γ is free hyper-parameter because it is dynamically adapted during training. In other aspects, the weighting factors measure the difficulty level of each sample. For example, the hard sample has high classification loss but regression loss and re-ID loss are arbitrary, thus the DMW will control the regression loss and re-ID loss to uniform values. Hence, the DMW makes the range of three tasks consistent.

α, β, and γ are mutual factors due to that they can adapt the parameter of the network according to task-dependent. Conventionally, the regression and classification task have a positive correlation, thus the network will normalize the learning pace based on dynamic weighting factors.

B. Position-sensitive Operation

The normal convolution operation is spatially invariant to some degree, which will hinder network performance to learn general cases. The spatial invariance is suitable for the image classification, segmentation task. Followed by CoordConv [7], the position-sensitive operation is used to be well spatial variant, illustrated in Fig. 2. First, we generate two tensors with the same spatial resolution of the input feature map. Two tensors corresponding to tensor x, and tensor y, consist of x-y pixel coordinates (location x and location y) normalized to the intervals $[-1, 1]$. It means the convolution operation accesses localized coordinates of input data. The two tensors are concatenated into the input feature map with channel dimension $(C + 2)$. This operation is easy to implement and integrate into CNNs architecture.

IV. EXPERIMENTS

A. Dataset and Evaluation Metrics

The challenging benchmark MOT17 [1] is used to measure the effectiveness of the proposed method. This dataset contains 7 videos for training and 7 videos for testing, which captured by single-camera. Since MOT17 did not provide ground truth for evaluation, the results are submitted to the evaluation system.

All results are measured by three prime metrics such as Multiple Object Tracking Accuracy (MOTA), ID switch (IDF1) proposed by CLEAR MOT [17], and Higher Order Tracking Accuracy (HOTA) proposed by [18]. Note that MOTA and IDF1 are standard metrics for evaluating detection and re-ID tasks without measuring localization perspective. HOTA measures whole aspects of tracking performance such as detection, localization, and trajectories problem.

B. Implementation Details

All implementations are utilized the deep learning Pytorch framework. The backbone ResNet-50 is pre-trained on ImageNet. The weight initialization of the added convolutional layers in feature pyramid FPN and five consecutive convolutional layers on each tracking head is filled from a normal distribution. The tracker is trained on GPU Tesla V100 SXM2 (Cuda 10.2, CuDNN 7.6.5) for 100 epochs with a batch size of 8. The initial learning rate is $3{\times}1e^5$ and reduced 10 times at epoch 50, and epoch 75. The Adam optimizer is used to minimize the objective function. Following common settings RetinaNet [4], and FCOS [5], the hyper-parameters in Equation 2 are set as $a = 0.25, b = 2.0$. Note that the number of anchor boxes tiled on each location is one for reducing model complexity.

V. RESULTS

This section describes the main results evaluated on the testing set and ablation study for measuring the importance of each component on the sub-training set.

A. Ablation Study

The importance of individual component. The experiment is implemented to investigate the effect of each component on the overall performance, shown in Table III.
The baseline is the simplest network of the single end-to-end tracker, which achieves 73.1% of the MOTA score. In this version, the weighting factors are set as uniform ($\alpha = 1$, $\beta = 1$, and $\gamma = 1$) followed by CTracker [2]. PSO is the positive-sensitive operation concatenated to the input feature map of each tracking branch, which boosts the baseline performance by 1.7% of the MOTA score. DMW is dynamic multi-loss weighting to balance objectives, gets 75.1% of the MOTA score. Remarkably, the full version of the proposed method achieves 75.7% of MOTA score, gains the baseline performance by a large margin. Note that MOTP, FP is the multiple object tracking precision, the number of false positives.

TABLE II
THE COMPARATIVE PERFORMANCE ON MOT17 TESTING SET WITH SOTA METHODS

Method	MOTA↑	IDF1↑	MOTP↑	MT↑	ML↓	FP↓	FN↓	IDS↓
DMAN [19]	48.2	55.7	75.9	19.3	38.3	26218	263608	2194
MHT DAM [20]	50.7	47.2	76.6	20.8	36.9	22875	252889	2314
FWT [21]	51.3	47.6	77.0	21.4	35.2	24101	247921	2648
SST [22]	52.4	49.5	76.9	21.4	30.7	25423	234592	8431
MOTDT [23]	50.9	52.7	76.6	17.5	35.7	24069	250768	2474
Tracktor [24]	53.5	52.3	78.0	19.5	36.6	**12201**	248047	2072
Tracktor+CTDet [24]	54.4	56.1	78.1	25.7	29.8	44109	210774	2574
DeepSORT [9]	60.3	**61.2**	**79.1**	31.5	**20.3**	36111	185301	2442
CenterTrack [3]	61.5	59.6	-	26.4	31.9	14076	200672	2583
TrackFormer [25]	61.8	59.8	-	-	-	35226	177270	2982
TransTrack [26]	65.8	56.9	-	32.2	21.8	24000	163683	5355
MOTR [27]	66.5	67.0	-	**33.5**	26.2	31302	**155715**	**1884**
CTracker [2]	66.6	57.4	78.2	32.2	24.2	22284	160491	5529
Ours	**67.4**	55.0	78.2	32.7	23.1	21033	156654	6330

TABLE III
THE IMPORTANCE OF EACH COMPONENT

Baseline	PSO	DMW	MOTA↑	IDF1↑	MOTP↑	FP↓
✓			73.1	63.3	85.3	2229
✓	✓		74.8	65.5	85.6	2137
✓		✓	75.1	65.4	84.8	2039
✓	✓	✓	75.7	66.0	85.3	1959

Dynamic Multi-Loss Weighting. This section analyzes how the DMW is introduced for correlating multi-task learning. The result is illustrated in Table IV.

TABLE IV
DYNAMIC MULTI-LOSS WEIGHTING WITH CORRELATION LEARNING

Method	MOTA↑	IDF1↑	MOTP↑	MT↑	FP↓
DMW w/correlation	75.1	65.4	84.8	281	2039
DMW w.o/correlation	73.8	64.7	85.6	256	1829

As shown in Equation 8, the weighting factors is dynamically changed during training, which depends on the correlation learning between each task to balance loss values. If each weighting factor is independently changed, the performance drops by 1.3% of MOTA score compared with correlation learning. Specifically, DMW without correlation learning has $\alpha = (L_{cls})/(L_{cls} + L_{reg} + L_{reid})$, $\beta = (L_{reg})/(L_{cls} + L_{reg} + L_{reid})$, and $\gamma = (L_{reid})/(L_{cls} + L_{reg} + L_{reid})$. The denominator is kept as Equation 8 to normalize weighting factors.

Error Analysis. The tracking error includes detection errors, localization errors, and association errors, which are shown in Fig. 3. The proposed method achieves the average HOTA score of 0.59 from loc_alpha=0.05 to loc_alpha=0.95 (localization thresholds) with a step size of 0.05. DetA got a score of 0.65 indicates detection accuracy, which decomposed into DetRe (detection recall) and DepPr (detection precision). The association accuracy (AssA) measures the overlap between predicted trajectories and ground truth, which consists of AssRe (association recall) and AssPr (association precision). Finally, LocA (localization accuracy) achieves a score of 0.87. The qualitative results of the proposed method are shown in Fig. 4. The video demos are available at https://bit.ly/3bdXqSv

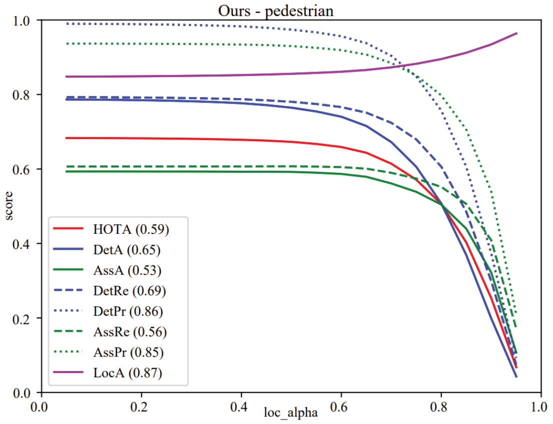


Fig. 3. HOTA score and error components at different threshold loc_alpha.

B. Comparison with State-of-the-art Online Tracker

This subsection describes the main performance of the proposed approach on the MOT17 test set, shown in Table II. MT, ML indicates Mostly Tracked Trajectories, Mostly Lost Trajectories. FP, FN denotes the number of False Positives and False Positive. IDS is the number of Identity Switches. The bold font shows the best performance among all state-of-the-art trackers.

Our method surpasses all online trackers by a large margin, which achieves 67.4% of the MOTA score. Specifically, the proposed method outperforms DMAN [19], MOTDT [23], Tracktor [24], Tracktor with CT detection, DeepSORT [9] and CTracker [2]. Moreover, our DMW, PSO did not affect the inference time, which applies to any tracking method.

VI. CONCLUSION

This paper introduced Dynamic Multi-Loss Weighting for balancing losses, dynamically adapted by the task-dependent, and the range of each task during training. Additionally, the position-sensitive operation is used for normalizing pixel coordinate to be spatially variant. The proposed method

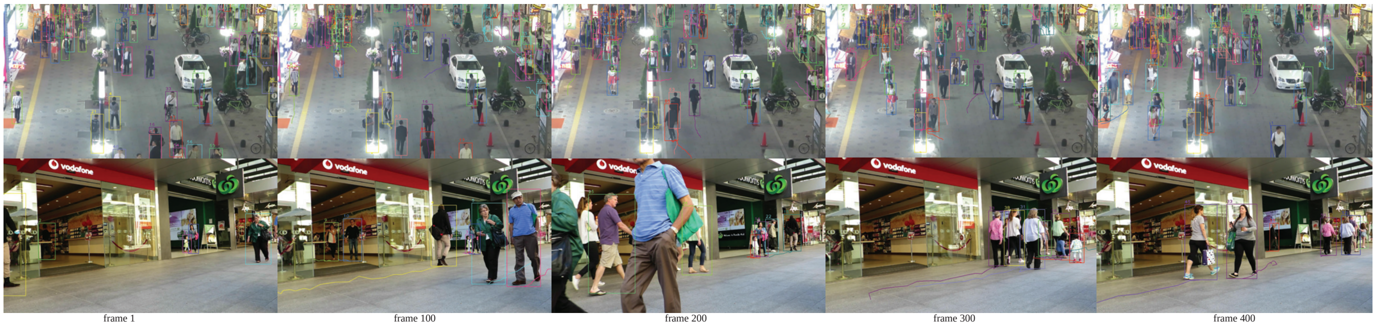


Fig. 4. The visualization of the proposed method with some frames.

is evaluated on the challenging benchmark MOT17, which outperforms state-of-the-art trackers without affecting model complexity. In the future, the proposed method will evaluate the performance of MOT16, MOT20, and PETS datasets.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government. (MSIT)(2020R1A2C2008972)

References

[1] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[2] J. Peng, C. Wang, F. Wan, Y. Wu, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Fu, "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking," in *European Conference on Computer Vision*. Springer, 2020, pp. 145–161.

[3] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," in *European Conference on Computer Vision*. Springer, 2020, pp. 474–490.

[4] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[5] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9627–9636.

[6] 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*, vol. 39, no. 6, pp. 1137–1149, 2016.

[7] R. Liu, J. Lehman, P. Molino, F. P. Such, E. Frank, A. Sergeev, and J. Yosinski, "An intriguing failing of convolutional neural networks and the coordconv solution," *arXiv preprint arXiv:1807.03247*, 2018.

[8] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE international conference on image processing (ICIP)*. IEEE, 2016, pp. 3464–3468.

[9] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.

[10] Z. Wang, L. Zheng, Y. Liu, and S. Wang, "Towards real-time multi-object tracking," *arXiv preprint arXiv:1909.12605*, vol. 2, no. 3, p. 4, 2019.

[11] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "Fairmot: On the fairness of detection and re-identification in multiple object tracking," *arXiv preprint arXiv:2004.01888*, 2020.

[12] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7482–7491.

[13] Q. Cai, Y. Pan, Y. Wang, J. Liu, T. Yao, and T. Mei, "Learning a unified sample weighting network for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 173–14 182.

[14] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "Solo: Segmenting objects by locations," in *European Conference on Computer Vision*. Springer, 2020, pp. 649–665.

[15] X.-T. Vo, L. Wen, T.-D. Tran, and K.-H. Jo, "Bidirectional non-local networks for object detection," in *International Conference on Computational Collective Intelligence*. Springer, 2020, pp. 491–501.

[16] X.-T. Vo and K.-H. Jo, "Enhanced feature pyramid networks by feature aggregation module and refinement module," in *2020 13th International Conference on Human System Interaction (HSI)*. IEEE, 2020, pp. 63–67.

[17] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, pp. 1–10, 2008.

[18] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, and B. Leibe, "Hota: A higher order metric for evaluating multi-object tracking," *International journal of computer vision*, vol. 129, no. 2, pp. 548–578, 2021.

[19] J. Zhu, H. Yang, N. Liu, M. Kim, W. Zhang, and M.-H. Yang, "Online multi-object tracking with dual matching attention networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 366–382.

[20] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4696–4704.

[21] R. Henschel, L. Leal-Taixé, D. Cremers, and B. Rosenhahn, "Improvements to frank-wolfe optimization for multi-detector multi-object tracking," *arXiv preprint arXiv:1705.08314*, vol. 8, 2017.

[22] S. Sun, N. Akhtar, H. Song, A. Mian, and M. Shah, "Deep affinity network for multiple object tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, pp. 104–119, 2019.

[23] L. Chen, H. Ai, Z. Zhuang, and C. Shang, "Real-time multiple people tracking with deeply learned candidate selection and person re-identification," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2018, pp. 1–6.

[24] P. Bergmann, T. Meinhardt, and L. Leal-Taixe, "Tracking without bells and whistles," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 941–951.

[25] T. Meinhardt, A. Kirillov, L. Leal-Taixe, and C. Feichtenhofer, "Trackformer: Multi-object tracking with transformers," *arXiv preprint arXiv:2101.02702*, 2021.

[26] P. Sun, Y. Jiang, R. Zhang, E. Xie, J. Cao, X. Hu, T. Kong, Z. Yuan, C. Wang, and P. Luo, "Transtrack: Multiple-object tracking with transformer," *arXiv preprint arXiv:2012.15460*, 2020.

[27] F. Zeng, B. Dong, T. Wang, C. Chen, X. Zhang, and Y. Wei, "Motr: End-to-end multiple-object tracking with transformer," *arXiv preprint arXiv:2105.03247*, 2021.

[28] F. Yu, W. Li, Q. Li, Y. Liu, X. Shi, and J. Yan, "Poi: Multiple object tracking with high performance detection and appearance feature," in *European Conference on Computer Vision*. Springer, 2016, pp. 36–42.