AutoPedestrian: An Automatic Data Augmentation and Loss Function Search Scheme for Pedestrian Detection

Yi Tang, Baopu Li, Min Liu, *Member, IEEE*, Boyu Chen, Yaonan Wang, and Wanli Ouyang, *Senior Member, IEEE*

***Abstract*— Pedestrian detection is a challenging and hot research topic in the field of computer vision, especially for the crowded scenes where occlusion happens frequently. In this paper, we propose a novel AutoPedestrian scheme that automatically augments the pedestrian data and searches for suitable loss functions, aiming for better performance of pedestrian detection especially in crowded scenes. To our best knowledge, it is the first work to automatically search the optimal policy of data augmentation and loss function jointly for the pedestrian detection. To achieve the goal of searching the optimal augmentation scheme and loss function jointly, we first formulate the data augmentation policy and loss function as probability distributions based on different hyper-parameters. Then, we apply a double-loop scheme with importance-sampling to solve the optimization problem of data augmentation and loss function types efficiently. Comprehensive experiments on two popular benchmarks of CrowdHuman and CityPersons show the effectiveness of our proposed method. In particular, we achieve 40.58% in MR on CrowdHuman datasets and 11.3% in MR on CityPersons reasonable subset, yielding new state-of-the-art results on these two datasets.**

***Index Terms*— Pedestrian detection, automatic data augmentation, loss function search.**

I. Introduction

PEDESTRIAN detection has been found its wide applications in video surveillance, robot navigation, personnel re-identification, autonomous driving, and so on. In recent years, with the developments of deep learning and convolutional neural network (CNN), many impressive progresses have been witnessed in pedestrian detection [1]–[13].

However, considering different challenges in practical environments such as occlusion, large variation of scales, poor illumination *et al*., there is still much room for improvement in pedestrian detection. For most deep learning related approaches, one common method to overcome the above challenges is data augmentation, which has been popular in deep learning for a few years since it can effectively enlarge the training data sets and greatly improve a model's generalization ability. Some general image data augmentation operations, including flip, scale, rotation, crop, cutout [14] and mixup [15], have been found to be popular in many image processing and computer vision problems [16].

For pedestrian detection, some data augmentation operations, like random resize cropping, brightness adjusting and some simulated occlusion, are used to alleviate the issues above. However, how to combine these operations and set magnitude of each operation require extensive experience and great effort [16]–[18]. Besides, some other operations may not be suitable here due to the specificity and unique characteristics of pedestrian. For example, rotation is inappropriate for pedestrian detection in most cases since the pose of a pedestrian tends to be vertical in an image. As such, it is nontrivial to find an effective augmentation combination of different operations, which is known as data augmentation policy [16], [17]. As far as we know, finding an effective data augmentation policy automatically for pedestrian detection remains under-explored. Hence, we intend to investigate a suitable data augmentation policy for CNN based pedestrian detector to alleviate the above issues that are common to pedestrian detection.

In addition, it is well known that loss function acts as a pivotal role in the field of machine learning and computer vision, and there are some widely used loss functions for object detection [19], [20] and pedestrian detection [5], [6], [21]. Generally speaking, designing a specific loss function for different problems needs much knowledge of human expertise, and its related hyper-parameter adjustment is time-consuming. To address these issues, we design a genetic loss function for pedestrian detection. Specifically, we apply different hyper-parameters to represent different loss functions. Moreover, it has been shown that the model performance

Manuscript received May 10, 2021; revised August 10, 2021 and September 9, 2021; accepted September 9, 2021. Date of publication October 7, 2021; date of current version October 13, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 62073126 and Grant 61771189 and in part by Hunan Provincial Natural Science Foundation of China under Grant 2020JJ2008. The work of Wanli Ouyang was supported in part by the Australian Research Council under Grant DP200103223 and Grant FT210100228, in part by the Australian Medical Research Future Fund under Grant MRFAI000085, and in part by the CRC-P Projects "ARIA—Bionic Visual-Spatial Prosthesis for the Blind" and "Smart Material Recovery Facility (SMRF)—Curby Soft Plastics," and SenseTime. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Weiyao Lin. *(Yi Tang and Baopu Li contributed equally to this work.) (Corresponding author: Min Liu.)*

Yi Tang, Min Liu, and Yaonan Wang are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China, and also with the National Engineering Laboratory for Robot Visual Perception and Control Technology, Changsha 410082, China (e-mail: liu_min@hnu.edu.cn).

Baopu Li is with Baidu USA, Sunnyvale, CA 94089 USA.

Boyu Chen and Wanli Ouyang are with the School of Electrical and Information Engineering, The University of Sydney, Sydney, NSW 2006, Australia.

This article has supplementary downloadable material available at https://doi.org/10.1109/TIP.2021.3115672, provided by the authors.

Digital Object Identifier 10.1109/TIP.2021.3115672

depends greatly on the suitable weight design for multi-task learning problems [22]. For pedestrian detection, the two tasks of classification and regression can also be considered as a specified multi-task problem. Hence, we also take the loss weight into consideration and aim to search the optimal loss function combination automatically in an effective manner.

Both data augmentation and loss functions play important roles in most computer vision tasks. However, data augmentation and loss function tend to have some synergistic effects [18]. For example, when more occlusion examples (hard examples) are generated by augmentation, the imbalance between hard examples and easy examples may be aggravated. A straightforward solution for this problem is to adjust the loss weights of hard and easy examples or design a specific loss function [23]. Therefore, we argue that a joint search manner for data augmentation and loss function may yield better results for pedestrian detection.

With the goal of finding optimal data augmentation policy and loss function simultaneously, we propose an AutoPedestrian scheme that can effectively search them automatically. Specifically, we propose a search space of data augmentation and loss function for pedestrian detection, and then model the data augmentation policy and the loss function policy as two distributions with different hyper-parameters.

Then, our goal is to search the respective optimal hyper-parameters. To solve this problem, we propose a double loop scheme. The inner loop minimizes the loss of a detection network with the joint sampled data augmentation operations and loss functions, while the outer loop maximizes rewards with respect to the data augmentation and the loss function distributions. Meanwhile, parameters with the best model will be broadcast to each parallel model synchronously. For outer loop optimization, a common solution is to apply REINFORCE [24] to optimize the related hyper-parameters by policy gradient estimation [17], [25]. However, common REINFORCE methods may suffer from poor sampling efficiency and unstable training process [26]. Therefore, we adapt our updating rules to off-policy learning using importance sampling to alleviate the above issue. We extensively test the performance of our proposed scheme on two benchmark datasets CityPersons [27] and CrowdHuman [28], validating its superior performance of pedestrian detection.

The contributions of our work are summarized as follows:

- 1) So far as we know, we propose the first framework to automatically search the optimal policy of data augmentation and loss function jointly for pedestrian detection.
- 2) We model the data augmentation policy and the loss function policy as two distributions with different hyper-parameters by the proposed search space, and then propose an AutoPedestrian scheme with importance sampling to solve the optimization problem efficiently, yielding the best data augmentation policy and loss function.
- 3) From the experiments on widely used benchmarks of CrowdHuman and CityPersons, we achieve 40.58% in MR on CrowdHuman datasets, and when no extra visible annotations are introduced, we obtain 11.3% in MR on CityPersons reasonable subset, thus yielding the new state-of-the-art results on these two datasets. The comprehensive ablation study also shows the effectiveness of our proposed method.

II. Related Works

A. Pedestrian Detection

Significant progresses have been made for object detection [23], [29]–[32]. Most of pedestrian detection methods are based on general object detection, but it still remains a challenging problem to detect occluded pedestrian in crowd scenes. Many approaches have been presented to handle the problem of occlusion for pedestrian detection [33]–[35]. Common approaches to this problem are part-based methods [2], which design a sequence of body-part based detectors to handle the occluded instances. Besides, some works also utilize new loss functions to handle the pedestrian detection in crowded scenes. The researchers of [5] come up with an Aggregation Loss to make the proposals as close as possible to the corresponding ground truth. The authors in [6] propose a Repulsion Loss to avoid proposals overlapping with multiple ground truths. Different from these related works, we intend to further overcome the problem of occlusion by means of automatic data augmentation as well as loss search in one framework.

B. Data Augmentation

Typical data augmentation operations include flip, scale, rotation, crop and so on, and have been widely used to increase the size of datasets and improve the generalization of networks. However, it is not easy to find a suitable data augmentation policy for different computer vision tasks. With the appearance of neural architecture search (NAS) [36]–[39] and Auto Machine Learning (Auto ML), some automatic data augmentation methods [16]–[18], [40], [41] are proposed to solve this problem. References [16], [40] take a recurrent neural network (RNN) as the controller to find the best data augmentation policy. Reference [42] proposes a efficient Bayesian hyperparameter optimization with density matching to speed up the process of searching. For data augmentation in pedestrian detection, [43] utilizes a rendering-based reshaping method to generate large number of synthetic training samples. Similar to [43], [44] also aims to generate synthetic training samples by generative adversarial networks from 3D game engines. However, the gap between synthetic pedestrians and real pedestrians could pose negative effects on detectors [45]. To solve this problem, [45] proposes a shape transformation-based framework to obtain more realistic-looking pedestrians. Different from these methods, we focus on searching a suitable augmentation policy and loss function for pedestrian detection, yielding better components for the CNN based pedestrian detection algorithms. So far as we know, automatically identifying a suitable data augmentation policy for pedestrian detection has not been well studied before, which is one of the focuses of this work.

C. Loss Function

Loss function is of great significance to computer vision problems. For object detection, the bounding box regression

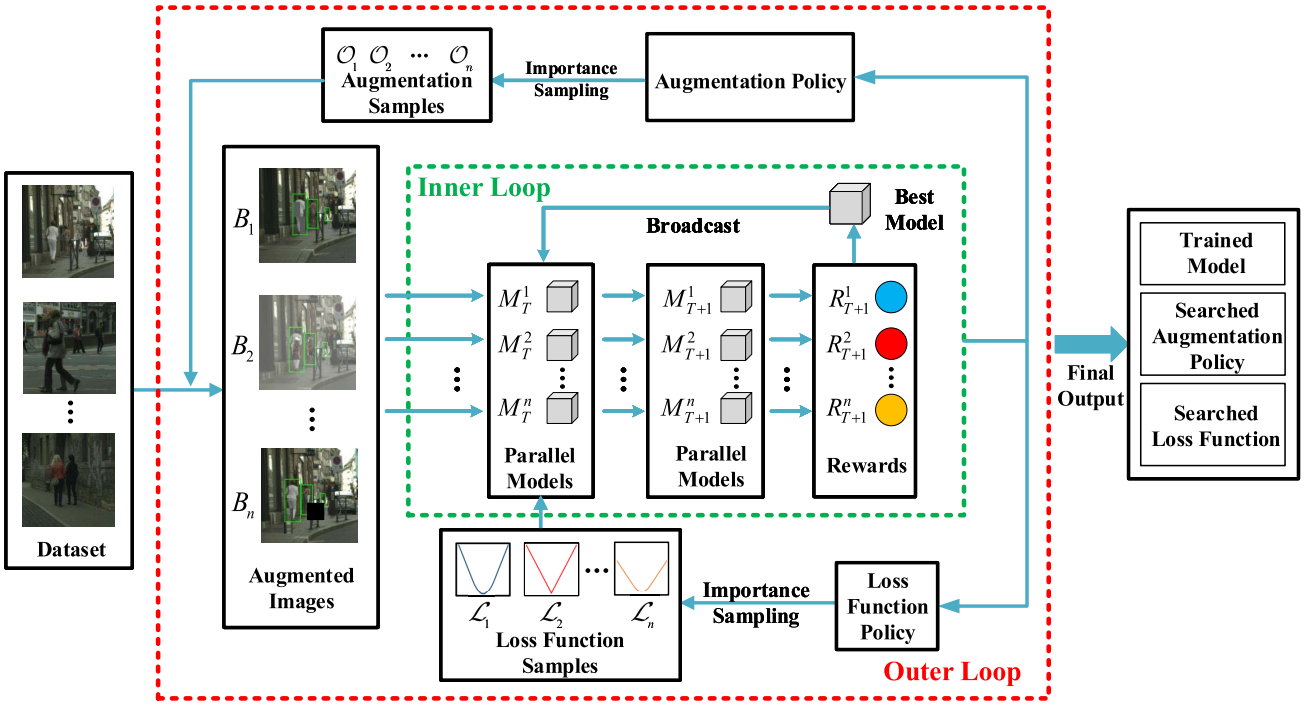

Fig. 1. Overview of our proposed AutoPedestrian scheme. In this bilevel optimization scheme, the inner loop is to optimize the parameters of detection network model with sampled data augmentation operations and loss functions. Meanwhile, the parameters of the best detector model will be broadcast to each parallel model synchronously. The outer loop is to optimize the hyper-parameters of data augmentation policy and loss function policy. Finally, our scheme will output the searched data augmentation, loss functions and trained model.

loss plays an important role in target location. Commonly used bounding box regression loss functions are SmoothL1 loss [30], IoU loss [20], GIoU loss [19], etc. As for loss weight, there have been some efforts to find suitable loss weights for multi-task learning [22]. However, designing loss functions and the corresponding weight of loss for each task is generally time-consuming and requires significant human expertise. Recently, some automatic loss function search methods are advanced to solve this problem [25], [46], which leverage REINFORCE [24] to search suitable parameters within a unified search space, but the search space is designed for classification task like personnel re-identification and face recognition. In contrast to these methods, we search the possible combinations of different loss functions that are specifically designed for pedestrian detection, thus leading to possible better detection performance.

III. Methods

In this section, we first give the problem formulation for our AutoPedestrian, followed by the proposed search space introduction and parameterization. Then, we present the solution of our problem based on bilevel optimization. The whole pipeline of the proposed scheme is illustrated in Fig. 1.

A. Problem Formulation

Let us assume that we have a detector model M_w parameterized by w. The training set and validation set are represented as $\mathcal{D}_{tr} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$ and $\mathcal{D}_{val}$ respectively. Then, we model the data augmentation policy and loss functions as two distributions. The data augmentation policy $\mathcal{O}_\theta$ is parameterized by θ and the loss function $\mathcal{L}_\varphi$ is parameterized by φ. The process of parameterization will be presented in Section III-B.4 and Section III-C.3. Thus, our objective function is to maximize the model M_w's rewards $R\left(M_{\boldsymbol{w}}; \mathcal{D}_{val}\right)$ with respect to θ and φ, which is defined as:

$$\begin{aligned} \theta, \varphi &= \arg\max_{\theta,\varphi} R\left(M_{\boldsymbol{w}^*}; \mathcal{D}_{val}\right), \\ \text{s.t.} \quad w^* &= \arg\min_{w} \sum_{(x,y)\in\mathcal{D}_{tr}} \mathcal{L}_\varphi(M_{\boldsymbol{w}}(\mathcal{O}_\theta(x)), y). \end{aligned} \tag{1}$$

It can be regarded as a standard bilevel optimization problem [47] if we consider the weight of the whole model w and the hyper-parameters θ and φ as two optimization goals.

To solve this bilevel optimization problem, we propose a double-loop optimization framework, which approximates the inner loop solution $\boldsymbol{w}^*$ using only I steps of training rather than the whole training scheme, and optimizes the hyper-parameters θ and φ in the outer loop.

Different from genetic object detection, the commonly used pedestrian detection evaluation metric is log-average miss rate (denoted as MR) on false positive per image (FPPI) in $\left[10^{-2}, 10^{0}\right]$, and the lower it is, the better it will be. To ensure that a better reward can yield a relative lower MR, our reward $R\left(M_{\boldsymbol{w}}; \mathcal{D}_{val}\right)$ could be represented as:

$$R\left(M_{\boldsymbol{w}}; \mathcal{D}_{val}\right) = 1 - MR\left(M_w; \mathcal{D}_{val}\right), \tag{2}$$

where $MR(\cdot)$ represents the performance of detection in terms of MR.

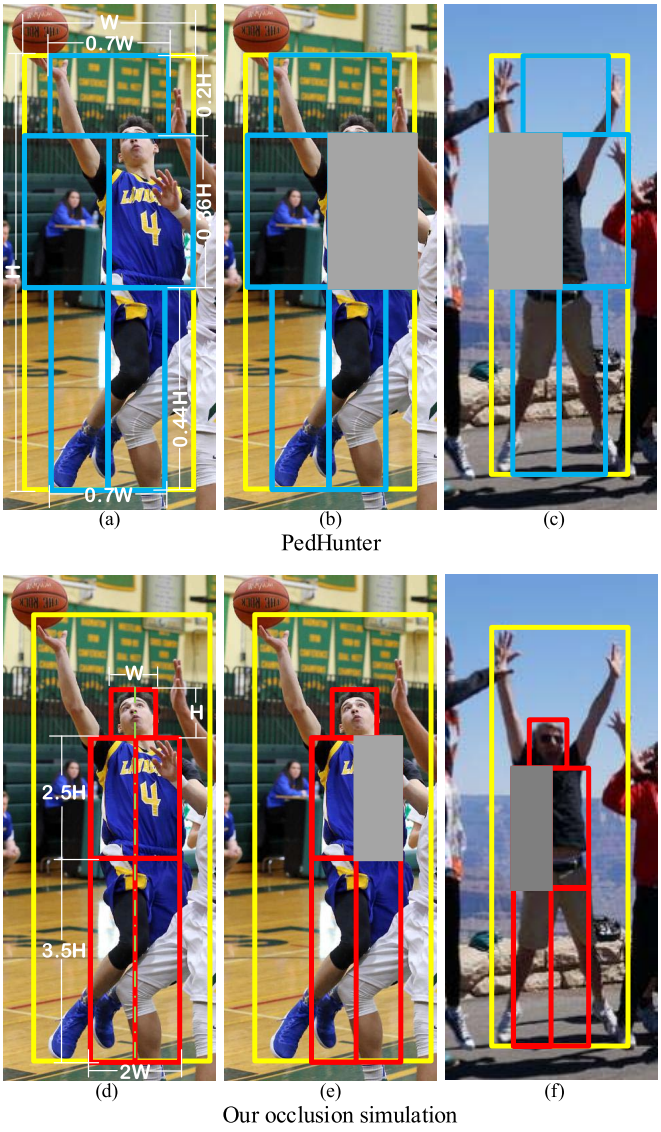

Fig. 2. Different partitions of ground truth between PedHunter [48] and our occlusion simulation. (a) is the partition of PedHunter, and it may occlude the main information of a person((b) and (c)) due to the rough partitions. (d) is our partition of ground truth. It may not occlude the main feature like head as shown in (e) and (f) and may be more accurate than PedHunter. The occlusion regions in PedHunter are filled with mean-value of ImageNet [50], while our occlusion regions are filled with mean-value of each image and thus yield different values on occlusion regions.

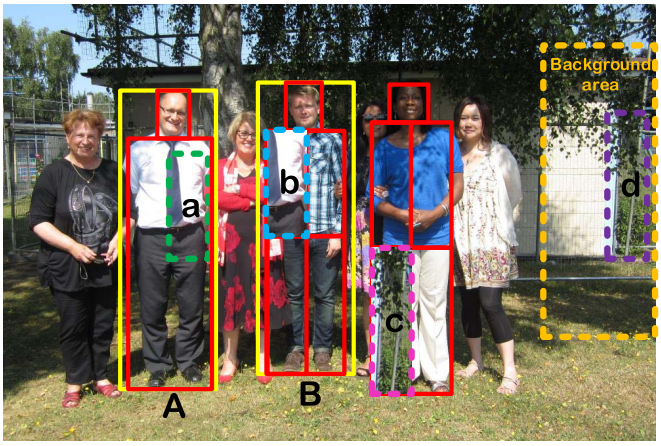

Fig. 3. Intra-class occlusion augmentation and inter-class occlusion augmentation. The intra-class occlusion means utilizing pedestrian region to block another pedestrian region (a and b), while the inter-class occlusion means applying non-pedestrian region to occlude the pedestrian region (c and d).

B. Search Space for Data Augmentation

1) Occlusion Simulation Augmentation: There are two types of occlusions in pedestrian detection: inter-class occlusion and intra-class occlusion [6]. Some occlusion simulation augmentation operations have been proposed in [48] for pedestrians to solve these problems. However, it shows its weakness for pedestrians with different poses. As illustrated in Fig. 2 (a), PedHunter [48] divides bounding box into 5 parts (i.e., head, left upper body, right upper body, left leg and right leg), but wrongly divides each part when persons are in varied poses. Besides, occlusion simulation based on this manner easily makes main features in person like head be blocked (e.g., Fig. 2 (b) and Fig. 2 (c)), leading the network confused about such main features. Therefore, we divide a person in a more accurate manner. As shown in Fig. 2 (d), we take the head bounding box as a reference and assume the width and height of head are W, H respectively, and the upper body width, height and leg width, height are roughly $2W$, $2.5H$, $2W$ and $3.5H$ respectively according to [49]. All of these parts are limited in ground truth bounding box. Fig. 2 (e) and Fig. 2 (f) also show that our division manner is more accurate and robust than [48]. The particular part from left upper body, right upper body, left leg and right leg are represented by an index from 0 to 3 respectively. The indexes are the same for different persons and can be searched in our paper. After partition of body parts, we take one part by index and occlude it with the mean-value of each image. Such a dynamic changed occlusion value may produce better generalization ability than the fixed value used by PedHunter [48] since it yields more data variation in the training process. This kind of occlusion simulation augmentation is called as mean value occlusion augmentation in this paper.

To improve the generalization of networks for these occlusions in real scenes, we propose another two types of data augmentation operations to simulate these kinds of occlusions: the intra-class occlusion augmentation and inter-class occlusion augmentation. For clarity, we show some examples in Fig. 3. For intra-class occlusion augmentation, we first select two ground truth bounding boxes A and B randomly when the number of ground truth is greater than 2 (as shown in Fig. 3 by the solid yellow bounding box A and B). Then, we crop the same shape with the selected index of B (the dashed blue box of b in Fig. 3) in body-part of A (the dashed green box of a in Fig. 3) and paste it to the selected area of B by index. The inter-class occlusion augmentation operation crops the same shape with the selected part of a person (the dashed purple box of d in Fig. 3) in some background area without pedestrians (the dashed orange bounding box in Fig. 3) and pastes it into the particular part of body by index (the magenta green box of c in Fig. 3). To avoid too much invalid or impractical occlusion like sky, roof of a building,

etc., the cropped background area is limited to the specified range of an image. i.e., $\pm 10\%$ of the height and width of the image and within the border of image. The subsequent results also validate that our proposed occlusion augmentation search space leads to better results for pedestrian detection.

2) Intensity and Color Augmentation: Intensity and color augmentation candidates include auto-contrast, equalization, equalization in bounding boxes, posterization, solarization, solarization-add, color balance, contrast, brightness, sharpness, cutout and cutout in bounding boxes. For fair comparison, we set the same range of magnitude as [40].

3) Geometrical Augmentation: Geometrical augmentation candidates consist of flip, flip in bounding boxes, resize operation, horizontal translation, vertical translation, horizontal shear, vertical shear, horizontal translation in bounding boxes, and vertical translation in bounding boxes. It should be noted that the rotation is not in our candidate list since pedestrians are mostly vertical in images. The range of magnitudes are also the same as [40] for fair comparison in the tests. Besides, we add the resize operation into our search space, because the pedestrians tend to have large variations of scales, and the resize adjusting ratio is in the range of [0.5, 2.0].

4) Parameterization of Data Augmentation Search Space: As a result, the total number of our augmentation operations is 114. Following [16], [40], we only apply two possible augmentation operations for each image. In addition, the same augmentation operation can repeat for each image. So the above operations may result in $114^2 \approx 1.3 \times 10^4$ possible augmentations. The probability of each operation $p_\theta(O_i)$ is a normalized sigmoid function of θ_i:

$$p_\theta(O_i) = \frac{\left(\frac{1}{1+e^{-\theta_i}}\right)}{\sum_j \left(\frac{1}{1+e^{-\theta_j}}\right)}, \tag{3}$$

where O_i is i^{th} operation sampled from candidate data augmentation operations. Thus, we have parameterized the probability distribution of the data augmentation in our proposed search space. This probability distribution is also our data augmentation policy in this paper.

C. Search Space for Loss Function

The loss function of deep learning based object detection generally includes two parts, that is, the classification loss functions and bounding box regression loss. We will construct a corresponding search space based on these two kinds of loss functions.

1) Loss Function for Classification: The widely used classification loss functions in object detection are cross entropy loss and focal loss [23]. However, it may be hard to choose which one is suitable for pedestrian detection with data augmentation. Therefore, we aim to search the best loss function for this task by a general loss function represented as:

$$L_{cls}(p) = -\left(y(1-p)^\gamma \log(p) + p^\gamma (1-y)\log(1-p)\right), \tag{4}$$

where p is the positive probability predicted by network, and y is the label. For the hyper-parameters γ that we are searching, the search space of γ is in the range of [0, 5], because γ must be positive and the upper bound of $\gamma = 5$ is enough according to [23]. From Eq.(4), when $\gamma = 0$, the loss function is cross entropy loss, and when $\gamma \neq 0$, it becomes focal loss with different focusing parameter γ.

2) Loss Function for Bounding Box Regression: The loss function for bounding box regression is composed of a pull term and a push term. The pull term aims to pull the predicted boxes to ground truth boxes, while the push term is to push the predicted boxes away from neighboring ground-truth boxes that are not targets since the pedestrians often occlude each other in crowd scenes. Hence, the loss function for bounding box regression can be represented as:

$$L_{reg} = \lambda_1 L_{pull} + \lambda_2 L_{push}, \tag{5}$$

where L_{pull} and L_{push} represent the pull term and the push term respectively. λ_1 and λ_2 are the loss weights of each term and are in the range of (0, 1].

For the pull term, we pull the predicted boxes in ground-truth boxes from three different perspectives. Firstly, the $SmoothL_1$ distance loss is the most commonly used one for bounding box regression, and we apply it as the first term in the pull loss function. Secondly, the intersection over union (IoU) is widely used to measure the accuracy for bounding box regression, and we introduce the IoU loss to refine the predicted boxes. Thirdly, to alleviate the occlusion caused by background, we take the ratio of width and height into consideration. It is because the box predictor is easy to infer the visible part but not the full body if pedestrians are occluded by objects, and one of the key differences between visible parts and full boxes is the ratio of width and height. Therefore, our pull loss term could be designed as:

$$L_{pull}(P,G) = \alpha_1 \cdot SmoothL_1(P,G) + \alpha_2 \cdot [1 - IoU(P,G)] + \alpha_3 \cdot \frac{4}{\pi^2}\left(\arctan\frac{p_w}{p_h} - \arctan\frac{g_w}{g_h}\right)^2. \tag{6}$$

where $P = \{p_x, p_y, p_w, p_h\}$ and $G = \{g_x, g_y, g_w, g_h\}$ are the proposal bounding box and the ground truth bounding box respectively, which are represented by their coordinates of left-top points as well as their widths and heights. α_1, α_2 and α_3 are the weights of each part, and the range of these parameters are [0, 1].

For the push term, we push the predicted boxes away from neighboring ground-truth boxes by two parts. On one hand, the predicted boxes should be slightly overlapped with other ground-truth boxes which are not targets. According to [6], it is better to utilize intersection over ground-truth (IoG) than intersection over union, because the predictor may learn to optimize the loss by simply enlarging the bounding box size to increase the overlap. Similar to IoU, the definition of IoG could be represented as:

$$IoG = \frac{area(P_i \cap G_i)}{area\ G_i}, \tag{7}$$

where G_i is the corresponding ground truth box of the predicted boxes P_i, and $(P_i \cap G_i)$ is the overlap between predicted boxes P_i and the corresponding ground truth box G_i.

However, RepLoss [6] does not take into consideration the inherent overlap existed in each ground-truth boxes, and it may be easy to push some predicted boxes that have well matched the ground-truth boxes away from the ground-truth boxes due to the inherent overlap in crowd scenes. Hence, we multiply a discount factor in the push term, and this term should be adaptive to the inherent overlap between ground-truth boxes. As a result, our push loss can be written as:

$$L_{push} = d(IoG\left(P_i, G_i\right)) \frac{\sum_{i \neq j} IoG\left(P_i, G_j\right)}{\sum_{i \neq j} \mathbb{1}\left[IoG\left(P_i, G_j\right) > 0\right] + \xi}, \tag{8}$$

where

$$d\left(x\right) = \left(1 - x\right). \tag{9}$$

$\mathbb{1}$ indicates the identity function and ξ is a small constant to prevent division by zero. From Eq.(8), the discount factor $d(IoG\left(P_i, G_i\right))$ can be very small when the predicted boxes match the ground-truth boxes well (i.e., $IoG\left(P_i, G_i\right)$ is large), while discount factor can be relatively large when the predicted boxes have a low intersection over ground-truth. Therefore, it could not only avoid pushing the exact predicted boxes away from the ground-truth in crowd scenes but also may be effective to the inexact predicted boxes with high overlap on other neighboring ground-truth boxes. Overall, the loss function to be searched could be represented as:

$$L = \eta_1 \cdot L_{cls} + \eta_2 \cdot L_{reg}, \tag{10}$$

where η_1 and η_2 are the weights for corresponding loss part, and the range is (0, 1].

3) Parameterization of Loss Function Search Space: We utilize the set φ to represent the parameters to be searched, and $\varphi = \{\alpha_1, \alpha_2, \alpha_3, \gamma, \lambda_1, \lambda_2, \eta_1, \eta_2\}$. For convenience of illustration, we use φ_k $(k = 0, 1, 2, \cdots 7)$ to represent the elements (e.g., α_1, α_2, *et al.*) in set φ. Since the search space of φ_k are continuous, the probability distribution of each φ_k can be modeled as an independent Gaussian distribution of mean μ_k and standard deviation σ [25], [46], described by:

$$\varphi_k \sim \mathcal{N}\left(\mu_k, \sigma^2\right). \tag{11}$$

Thus, our loss function policy could be parameterized by these Gaussian distributions. Then, the best loss function can be searched by optimizing the Gaussian distribution of mean μ_k in our double-loop scheme, which is introduced in the next section.

D. Optimization Process

1) Optimization of Inner Loop: We follow similar network training principle in [17] to approximate $\boldsymbol{w}^*$. We use I to represent the training steps in the inner loop, $T_{\max}$ to denote the max update times in the outer loop, and B to represent the batch size of each step. The trajectory $\mathcal{T} = \left\{O^T; \mathcal{L}^T\right\}$ is utilized to represent all the augmentation operations and loss functions sampled in the T^{th} period. The T^{th} optimization of the model parameters can be written as follows:

$$w_T^{j+1} = w_T^j - \beta \nabla_w \mathcal{L}^T(O^T(x_B), y_B), \tag{12}$$

where β is the learning rate of the detection model and j is the j^{th} iteration of training steps I, while x_B and y_B are the mini-batch input and corresponding ground truth, respectively.

To ensure that the mini-batch and the detector model are the same for different sampling trajectories, we utilize parallel models in the inner loop, so that each parallel reward is only affected by the augmentation policy and loss function of each parallel model. Through this manner, the gradient estimations of augmentation parameters and loss function parameters are accurate and not influenced by other factors. In addition, parallel model training is more efficient than sequential model training, because multiple models can be trained simultaneously.

2) Optimization of Outer Loop: As illustrated in Eq. (1), our goal is to maximize the rewards with an optimal data augmentation probability distribution (or policy) donated by π_θ and a loss function probability distribution donated by π_φ. For clarifications, we take the augmentation parameter θ as an illustration since the loss function can be searched in the same way. The gradient ascent is used to update the parameters of data augmentation, which could be written as:

$$\begin{aligned} \theta_{T+1} &= \theta_T + \eta_\theta \nabla_\theta R\left(\theta_T\right) \\ &= \theta_T + \eta_\theta \nabla_\theta \mathbb{E}_{\mathcal{T} \sim \pi_\theta}[R(\mathcal{T})], \end{aligned} \tag{13}$$

where η_θ represents the learning rate for the augmentation distribution parameters. The above process is updated iteratively until reaching the maximum updating times T_{max} or when the network model is converged.

There are two problems needed to be solved in Eq. (13). Firstly, the reward $R(\mathcal{T})$ is non-differentiable with respect to θ. Secondly, it is impractical to calculate the whole trajectory $\mathcal{T}$ analytically. A common solution is to apply REINFORCE [24] with policy gradient to approximate the gradient by Monte-Carlo sampling. However, Monte Carlo sampling uniformly samples the entire region with equal weight. The greater the number of trajectories sampled, the more accurate the results, but with more computational costs. Unlike Monte Carlo sampling, importance sampling is equivalent to giving an attention to the sampling process, and the region with larger contributions to the reward are sampled with larger weights, so that importance sampling can explore the region where the better policy is likely to be obtained with fewer trajectories [51]. Therefore, we adapt our updating rule to off-policy learning using importance sampling to improve the sampling efficiency.

For importance sampling, the old policy $\pi_{\theta_{old}}$ can be used directly, and the policy π_θ can be represented by a probability distribution $P(\theta)$, so we have:

$$\begin{aligned} \nabla_\theta \mathbb{E}_{\mathcal{T} \sim \pi_\theta}[R(\mathcal{T})] &= \nabla_\theta \mathbb{E}_{\mathcal{T}' \sim \pi_{\theta_{old}}}[\frac{\pi_\theta}{\pi_{\theta_{old}}} R(\mathcal{T}')] \\ &= \nabla_\theta \sum_{\mathcal{T}'} \frac{P\left(\mathcal{T}'; \theta\right)}{P\left(\mathcal{T}'; \theta_{old}\right)} R\left(\mathcal{T}'\right) \\ &= \sum_{\mathcal{T}'} \frac{P(\mathcal{T}'; \theta) R(\mathcal{T}')}{P\left(\mathcal{T}'; \theta_{old}\right)} \log \nabla_\theta P(\mathcal{T}'), \end{aligned} \tag{14}$$

where $\mathcal{T}'$ is a trajectory of $\pi_{\theta_{old}}$.

We sample N trajectories from $\pi_{\theta_{old}}$ to approximate the gradients by importance sampling. For a sampled trajectory $\mathcal{T}$, it is composed of a sequential augmentation operations sampled by the probability distribution p_θ, and it can be calculated as $P(\mathcal{T}) = \prod_{j=1}^{I\times B} p_\theta\left(O_j\right)$ (where B is the batch size) since each operation sampling probability of each batch in each iteration is independent of each other. Then, Eq. (14) can be rewritten as:

$$\begin{aligned}
&\nabla_\theta \mathbb{E}_{\mathcal{T}'\sim\pi_{\theta_{old}}}[\frac{\pi_\theta}{\pi_{\theta_{old}}}R(\mathcal{T}')] \\
&\approx \frac{1}{N}\sum_{i=1}^{N}\frac{P\left(\mathcal{T}';\theta\right)R_i}{P\left(\mathcal{T}';\theta_{old}\right)}\log \mathrm{P}\left(\mathcal{T}';\theta\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{I\times B}\frac{\prod_{j=1}^{I\times B} p_\theta\left(O_j\right)}{\prod_{j=1}^{I\times B} p_{\theta_{old}}\left(O_j\right)}\nabla_\theta \log p_\theta\left(O_j\right)R_i, \qquad (15)
\end{aligned}$$

where R_i is the reward of the i^{th} sampled trajectory. For convenience, $\sum_{j=1}^{I\times B}\frac{\prod_{j=1}^{I\times B} p_\theta(O_j)}{\prod_{j=1}^{I\times B} p_{\theta_{old}}(O_j)}$ is denoted as r_θ, and it is also the probability ratio between the target policy π_θ and the sampling policy $\pi_{\theta_{old}}$. Our final gradient estimation equation is:

$$\nabla_\theta \mathbb{E}_{\mathcal{T}\sim\pi_\theta}[R(\mathcal{T})] \approx \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{I\times B}\nabla_\theta \log p_\theta\left(O_j\right)R_i r_\theta^i, \qquad (16)$$

where

$$r_\theta^i = \sum_{j=1}^{I\times B}\frac{\prod_{j=1}^{I\times B} p_\theta\left(O_j\right)}{\prod_{j=1}^{I\times B} p_{\theta_{old}}\left(O_j\right)}. \qquad (17)$$

Similar to the gradient estimation of data augmentation policy, the gradient estimation of loss function can be represented as:

$$\nabla_{\mu_k}\mathbb{E}_{\varphi_k\sim\mathcal{N}(\mu_k,\sigma^2)}[R(\mathcal{T})] \approx \frac{1}{N}\sum_{i=1}^{N}\nabla_{\varphi_k}\log\left(g\left(\varphi_k^i\right)\right)R_i r_{\varphi_k}^i, \qquad (18)$$

where

$$r_{\varphi_k}^i = \frac{g\left(\varphi_k^i\right)}{g\left(\varphi_{k_{old}}^i\right)}, \qquad (19)$$

where $g\left(\varphi_k\right)$ is the probability density function of Gaussian distribution, which could be used to represent the probability of each sampled parameter φ_k in a continuous space. It should be noted that to avoid the frequent change of the loss function value that may lead to an unstable training process, we sample the loss before the T^{th} training epoch and maintain the sampling loss during the T^{th} training epoch.

To make sure each step of optimization is limited in a trust region and obtain a better reward and policy in a steady manner [26], the probability ratio r_θ and $r_{\varphi_k}^i$ are clipped by a clip ratio ϵ as suggested in [51], which is:

$$r_\theta^i, r_{\varphi_k}^i = \mathrm{clip}(1-\epsilon; 1+\epsilon). \qquad (20)$$

For better illustration, the whole optimization pipeline is described in Algorithm 1.

Algorithm 1 AutoPedestrian for Data Augmentation and Loss Function Search

input : The training set D_{tr}, validation set D_{val}, detection model M_w.
initialize θ_0, μ_0 and the same w for N parallel models; $T = 0$;
while $T \le T_{\max}$ **do**
  for each $\mathcal{T}_i$ that $(1 \le i \le N)$ **do**
    Sample $\mathcal{L}_i$ from $\varphi_k \sim \mathcal{N}\left(\mu_k, \sigma^2\right)$;
    for j that $1 \le j \le I \times B$ **do**
      Sample $\mathcal{O}_{T,i}^j$ from π_{θ_T};
      Update $w_{T,i}^j$ by Eq. (12);
    end
    Fix $w_{T,i}$, evaluate $M_{w_{T,i}}$ and get the reward R_i ;
  end
  Calculate $\nabla_\theta R\left(\theta_T\right)$ and $\nabla_\mu R\left(\mu_T\right)$ by Eq. (16) and Eq. (18);
  $\theta_{T+1} \leftarrow \theta_T + \eta_\theta \nabla_\theta R\left(\theta_T\right)$;
  $\mu_{T+1} \leftarrow \mu_T + \eta_\mu \nabla_\mu R\left(\mu_T\right)$;
  Select w_T from$\{w_{T,i}\}_{i=1:N}$ with the best reward;
  Broadcast w_T to all the parallel models;
  $T = T + 1$;
end
output: Optimized model, data augmentation policy and loss function.

IV. Experiments

A. Datasets and Evaluation Metrics

1) CrowdHuman: The CrowdHuman dataset [28] is a benchmark dataset to validate the performance of pedestrian detectors in crowd scenarios with an average of 22.6 persons in each image. It is divided into training (15,000 images), validation (4,370 images) and testing (5,000 images) subsets. We evaluate our proposed method on full body detection. Our models are trained on the training subsets and evaluated on validation subsets.

2) CityPersons: The CityPersons dataset [27] is a pedestrian detection dataset with high quality bounding box annotations of full body and visible body. It consists of 5,000 images, 2,975 for training, 500 for validation, and 1,525 for testing. We evaluate our proposed method on full body detection. The validation set is divided into reasonable (**R**) subset and heavy occlusion (**HO**) subset according to visibility ratio [6]. The visibility ratio of reasonable subsets is larger than 65% and the visibility ratio of heavy occlusion subsets ranges from 20% to 65%. Our models are trained on reasonable training subset. All of our results on CityPersons are reported in **R** and **HO** subsets.

3) Caltech: The Caltech dataset [52] is a popular pedestrian detection dataset. It consists of 11 sets of video sequences, first 6 sets for training and last 5 sets for testing. Following [7], [53], we sample the frames of the training video sequence at 10 Hz and the test video sequence at 1 Hz. As a result, there are 42782 images for training and 4024 images for

testing. Similar to CityPersons [27], the test set is divided into reasonable (**R**) subset and heavy occlusion (**HO**) subset according to visibility ratio. All of our results on CityPersons are reported in **R** and **HO** subset.

4) HiEve: The HiEve dataset [54] is a challenging dataset for human-centric video analysis. It consists of 32 video sequences, 19 for the training, 13 for the testing. We sample all the frames from training video sequences as our training sets (32944 images in total). Our results are reported in HiEve test subsets.

5) Valuation Metrics: The main evaluation metric is the MR mentioned in Section III-A. Results are reported on full body detection, and the ignored regions are not calculated in the evaluation. Besides, average precision (AP) and recall are also included for comprehensive evaluation. For HiEve dataset, Jaccard index (JI) is introduced to evaluate how much the predictions overlap with the ground truths, and a larger JI indicates better performance.

B. Implementation Details

For CrowdHuman, we choose the Faster R-CNN [30] (F-RCNN) based on the Feature Pyramid Network (FPN) [32] with ResNet-50 [55] pre-trained on ImageNet [50] as our two stage detector baseline. We use the RoI Align [31] instead of RoI Pooling [30] for more precise feature extraction, and the anchor aspect ratios are set as [1.0, 2.0, 3.0]. For one stage detector, we choose RetinaNet [23] with ResNet-50 [55] pre-trained on ImageNet as our baseline. We resize their short edges to 800 pixels and the long edges to no more than 1400 pixels for the images of CrowdHuman are varied in size. The basic data augmentation is a standard horizontal flipping with a probability of 0.5. Regarding the network training, we use standard SGD optimizer with 0.9 momentum and set the mini-batch size as 2. Considering the computational burden and convergence of our parameters, we train our model for 120 epochs in total as a trade-off. In addition, the initial learning rate is set as 0.0025 and is decreased by 0.1 at 80^{th} and 110^{th} epochs.

For CityPersons, we use the F-RCNN with the same settings as those of [27], but remove the fourth max-pooling layer in VGG-16 [56] backbone for small pedestrian detection. The anchor aspect ratio is set as 2.44 and the anchor sizes are the same as [27]. We use the standard SGD optimizer with 0.9 momentum and set the mini-batch size as 2. The number of training epochs is set as 200. The initial learning rate is set to 0.0025 and decreased by 0.1 at 160^{th} and 190^{th} epochs. Since there is no head bounding box annotation in CityPersons, we still use the partition manner in PedHunter [48]. After partition, we can apply mean, intra-class and inter-class occlusions as occlusion simulated augmentations.

For the double-loop optimization process, the number of trajectories and the clip ratio in two datasets are set as 8 and 0.2, respectively. The investigation of these two hyper-parameters will be illustrated in Section IV-C.3. As for the two parameters η_θ and η_μ, their learning rate are set to 0.05 according to [17], [25], [46]. Due to the modularization and flexibility of MMdetection [57], we choose it as our toolbox of detection model. Our whole scheme is trained and tested on 4 NVidia RTX 2080Ti GPUs.

TABLE I

IMPACT OF EACH COMPONENT AND SEARCHING MANNERS ON CROWDHUMAN

Methods	AA*	ALF*	Separate	Joint	MR(%)	AP(%)	Recall(%)
Baseline [28]	-	-	-	-	50.42	84.95	90.24
Baseline(our)	-	-	-	-	48.52	86.58	91.74
	✓	-	✓	-	46.85	87.74	93.04
	-	✓	✓	-	47.05	87.31	92.57
	✓	✓	✓	-	46.47	87.92	93.36
	✓	✓	-	✓	**45.53**	**88.23**	**93.74**

*: AA and ALF mean our automatic augmentation and automatic loss function respectively.

C. Ablation Study

In this section, we evaluate our AutoPedestrian by conducting an ablation study on the CrowdHuman dataset.

1) Impact of Each Component and Searching Manners: We first assess the impact of our automatic augmentation (AA) module and automatic loss function (ALF) module separately. The respective results on CrowdHuman are shown in Table I. Our F-RCNN baseline is implemented in MMdetection and it performs a little better than the F-RCNN baseline in [28]. As shown in Table I, after adding the module of AA, the performance in terms of MR, AP, and Recall can be boosted by 1.67%, 1.16%, 1.30%, respectively. Moreover, when only the ALF module is added, the corresponding performance can be improved by 1.47%, 0.73%, 0.83% respectively compared with the baseline. Furthermore, to investigate the best search manner in our scheme after combining the AA module and ALF module, we test separate manner (Separate in Table I) and joint manner (Joint in Table I) on CrowdHuman. The separate manner means that we first search the data augmentation policy separately, and then search the loss function with the searched data augmentation policy. The joint manner means that we search the data augmentation and loss function jointly. It can be seen that the joint manner is much better than the separate manner (45.53% MR VS 46.47% MR). Such a result also indicates that there are some synergistic effects between data augmentation and loss function, and the best way to search these two components is the joint manner.

2) Impact of the Search Space: To investigate the influence of the proposed search space for our AutoPedestrian, we compare our baseline with different search space. As illustrated in Table II, different search space are tested and their respective detection performance are recorded. When only basic augmentation search space (BASS) is adopted, our model obtains 0.89%, 0.71%, 0.83% gains over the baseline model in terms of MR, AP and Recall, respectively. While adding our occlusion simulation augmentation search space (OSASS), the performance can be further improved by 0.78%, 0.45%, 0.47% compared to basic augmentation search space, respectively. As for loss function, our classification loss search space (CLSS) could yield 0.74%, 0.14%, 0.45% improvement in terms of MR, AP and Recall respectively compared to our baseline (the first row in Table II). In addition, when

TABLE II
IMPACT OF THE DIFFERENT SEARCH SPACE* ON CROWDHUMAN

Methods	MR(%)	AP(%)	Recall(%)
Baseline	48.52	86.58	91.74
Baseline + BASS	47.63	87.29	92.57
Baseline + BASS + OSASS	46.85	87.74	93.04
Baseline + CLSS	47.78	86.72	92.19
Baseline + CLSS + RLSS	47.05	87.31	92.57

*: BASS denotes the basic augmentation search space, which only includes intensity, color and geometrical data augmentations as mentioned in Section III-B2 and Section III-B3. OSASS denotes the occlusion simulation augmentation search space, which includes our mean occlusion augmentation, intra-class and inter-class occlusion augmentation as mentioned in Section III-B1. CLSS denotes the classification loss search space as mentioned in Section III-C1. RLSS denotes the regression loss search space as mentioned in Section III-C2.

TABLE III
COMPARISON OF DIFFERENT CLIP RATIO ϵ ON CROWDHUMAN IN TERMS OF MR(%)

Baseline	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$	$\epsilon = 0.5$
48.52	46.32	**45.53**	46.85	47.04	47.16

TABLE IV
COMPARISON WITH OTHER AUTOMATIC DATA AUGMENTATION METHODS

Methods	MR(%)	AP(%)	Recall(%)
Baseline	48.52	86.58	91.74
AutoAugment	47.95	87.12	92.36
OHL (Our impl.*)	47.48	87.48	92.89
Ours	**46.85**	**87.74**	**93.04**

*: Our impl. means that this method has not been tested on target datasets, and we re-implement it on target datasets and report the corresponding results.

integrating the regression loss search space (RLSS), the performance can be further increased by 0.73%, 0.59%, 0.38% in terms of MR, AP, and Recall, respectively. Therefore, our proposed search space of data augmentation and loss function is effective and leads to better results in pedestrian detection.

3) Impact of Different Hyper-Parameters Settings: There are two key hyper-parameters that need to be investigated in our AutoPedestrian method: the clip ratio ϵ in Eq. (19) and the number of sampled trajectories N. We report the results of our AutoPedestrian method with different clip ratios in Table III. When setting ϵ as 0.2, the AutoPedestrian yields the best performance of 45.53% in terms of MR on CrowdHuman datasets. In addition, the results with different number of sampled trajectories are shown in Fig. 4. It is clear that the model can obtain a better performance when the number of sampled trajectories is larger. This is because the gradient estimation may be more precise when the number of sampled trajectories is larger. However, more trajectories yield a heavier computational burden. In addition, as shown in Fig. 4, the reduction of MR is minor from $N = 8$ to $N = 16$. Hence, we select $N = 8$ as the number of sampled trajectories to make a trade-off between performance and computation burden.

4) Comparison With Other Automatic Data Augmentation Methods: We make a comparison with AutoAugment [40] and OHL [17], which are the popular automatic data augmentation methods in Auto ML. The results are reported in Table IV. Since the AutoAugment [40] is very

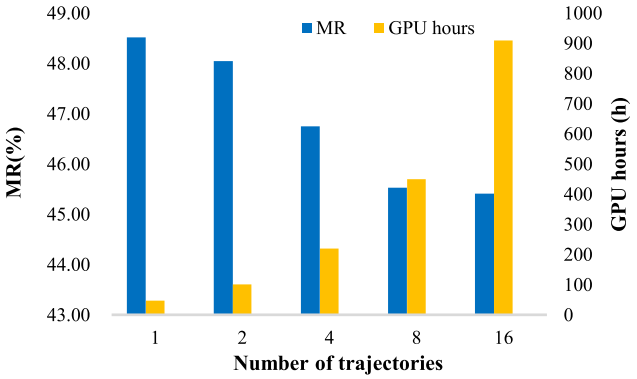

Fig. 4. Impact of different trajectory numbers on performance and GPU costs. The blue bins are the MR of different trajectory numbers. The model can obtain a better performance when the number of sampled trajectories is larger. The yellow bins are the GPU hours of different trajectory numbers.

TABLE V
RESULTS OF SEARCHED LOSS FUNCTION PARAMETERS ON CROWDHUMAN

Parameter	α_1	α_2	α_3	λ_1	λ_2	γ	η_1	η_2
Value	1.00	0.06	0.47	0.56	0.43	1.20	0.67	0.74

time-consuming (i.e., 19200 GPU hours on COCO [58] dataset) and cannot be re-implemented by most other researchers. In addition, they did NOT open source their codes and only showed the searched optimal policy. As such, we implemented the results of AutoAugment with the searched policy presented in [40] on COCO dataset. Besides, the OHL is a data augmentation method for classification, but we re-implement it with our designed augmentation search space in pedestrian detection. The training time of these three methods are the same for fair comparison. As illustrated in Table IV, our automatic data augmentation method outperforms AutoAugment by 1.10% and OHL by 0.63% in terms of MR, respectively. Therefore, our AutoPedestrian scheme is more suitable than other automatic data augmentation methods for pedestrian detection.

D. Empirical Analysis

1) Analysis for Data Augmentation: As for data augmentation, since there are about 114 types of operations in our scheme, it is impractical to show all the probability distributions of each operation. We illustrate a part of representative probability distributions in Fig. 5. It could be observed that the probability of operation has converged. The probability of some operations tend to increase (more red), while some become discarded (more blue) during training. This phenomenon suggests that the suitable data augmentation operations will be sampled with higher probability, while some inappropriate operations will be sampled with lower probability. As a result, the data augmentation policy could be represented by the final converged probability distributions.

2) Analysis for Loss Functions: The convergence of loss function parameter is shown in Fig. 6. It could be seen that the distribution parameters of loss functions tend to converge

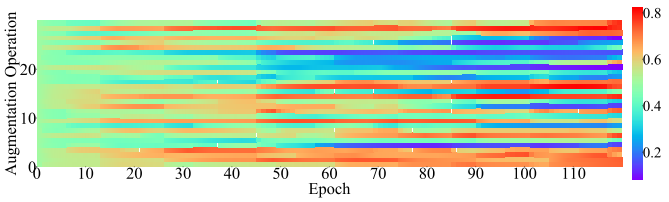

Fig. 5. Probability convergence process of augmentation operations. The probability of different data augmentation operations tends to be higher (more red) or lower (more blue).

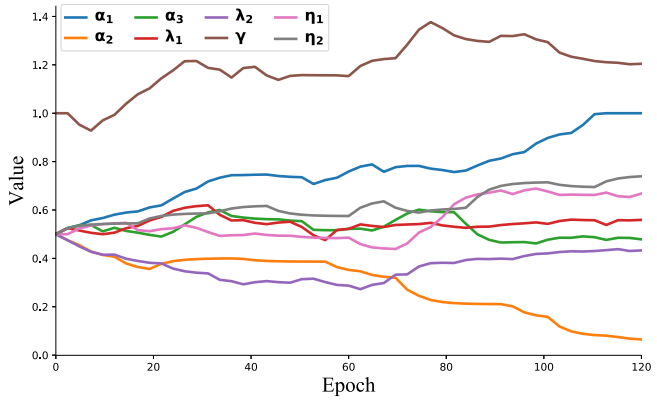

Fig. 6. Convergence process of loss function parameters. The distribution parameters of loss functions tend to converge to specific values as the epochs increase.

to specific values as the epochs increase. For further analysis, the value of α_1 is close to 1 (blue line), while the value of α_2 is close to 0 (orange line), which indicates that the $SmoothL_1$ loss is more suitable than the IoU loss in CrowdHuman dataset. The α_3 is close to 0.5 (green line), meaning that the ratio of width and height may be helpful to handle the occlusion problem, and its weight is half of $SmoothL_1$ loss. Moreover, the focusing parameter γ converges to 1.2 (brown line), which indicates that focal loss is more suitable than Cross-Entropy loss in crowd scenes, and the focusing parameter could not be too large. As for the loss weights of pull term and push term, the searched results indicate that their weights should be close to each other. In addition, the loss weights of classification and regression are both close to 0.7. Besides, all parameters incline to converge to specific values, which suggest that our scheme can find a stable parameter for each term of loss function. The searched parameters of loss function are reported in Table V, which could be helpful for the design of loss function in pedestrian detection community.

E. Comparison With State-of-the-Art Methods on CrowdHuman Dataset

There are many different baseline results reported by different papers on CrowdHuman. For example, the result of the two-stage detector F-RCNN reported by the CrowdHuman official paper [28] is 50.42% in MR, and Adaptive NMS [12] can only obtain 52.35% in MR, but PBM [59] and CrowdDet [10] can achieve 46.28% and 42.9% in terms of MR, respectively. The main reason is because the authors of the official CrowdHuman do not open source the codes of their model. Moreover, there are many hyper-parameters to be set besides those reported in their paper. Therefore, we apply our AutoPedestrian scheme in the best-performing baseline, i.e., the CrowdDet [10] baseline. In addition, we also conduct experiments on some one-stage detectors to verify the generalizability of our scheme.

TABLE VI

COMPARISON WITH THE STATE-OF-THE-ART METHODS ON CROWDHUMAN IN TERMS OF MR(%)

	Methods	MR(%)
Two Stage Detector	F-RCNN [28] (official)	50.42
	Adaptive NMS [12]	49.73
	PBM [59]	43.35
	CrowdDet [10] baseline	42.90
	CrowdDet [10]	41.40
	CrowdDet [10] Baseline + our *	40.65
	CrowdDet [10] Baseline + ours	**40.58**
One Stage Detector	RetinaNet [28] (official)	63.33
	ATSS [29]	61.89
	FCUS [60]	58.76
	RetinaNet (our impl.)	54.94
	RetinaNet (impl. by [10])	54.81
	ATSS [29] + our *	60.08
	ATSS [29] + our	60.04
	FCUS [60] + our *	56.42
	FCUS [60] + our	56.37
	RetinaNet (our impl.) + ours *	52.57
	RetinaNet (our impl.) + ours	**52.46**

*: It means that the model is trained from scratch with our final searched hyper parameters .

Most existing two-stage detectors are based on F-RCNN on CrowdHuman datasets. We make comparisons with the official F-RCNN [28], Adaptive NMS [12], PBM [59] and CrowdDet [10]. As illustrated in Table VI, CrowdDet baseline together with our AutoPedestrian outperforms the existing best method CrowdDet by 0.82% in terms of MR. For one stage detector, it should be noted that our RetinaNet baseline implemented by MMdetection performs much better than the CrowdHuman official baseline (i.e., 54.94% MR vs 63.33% MR). As shown in Table VI, RetinaNet plus our AutoPedestrian achieves the best result in one-stage detector and outperforms the top-performed RetinaNet in [10] by 2.35% in MR. Such results also show that although the baseline model of CrowdDet have achieved impressive results on CrowdHuman datasets, our AutoPedestrian can still improve the best model by the searched data augmentation policy and loss functions.

In addition, we conduct the experiment of using the final searched hyper parameters to train a model from scratch in TABLE VI. It can be seen that the models trained from scratch by our searched parameters can achieve the same or comparable performance with the final best model (40.65% MR vs 40.58% MR for Crowddet baseline, 60.08% MR vs 60.04% MR for ATSS, 56.42% MR vs 56.37% MR for FCUS, 52.57% MR vs 52.46% MR for RetinaNet). Therefore, the promotion of our AutoPedestrian to the performance of one-stage and two-stage detector baseline also reflects generalizability of our scheme.

F. Comparison With State-of-the-Art Methods on CityPersons Dataset

Our scheme is compared to the recent state-of-the-art methods such as Adapted Faster RCNN [27], ATT-part [4],

TABLE VII

COMPARISON WITH THE STATE-OF-THE-ART METHODS ON CITYPERSONS. THE MR(%) OF FULL BODY DETECTION ON VALIDATION SUBSETS IS REPORTED

Methods	Backbone	Scale	**R**	**HO**
Adapted Faster RCNN [27]	VGG-16	×1.0	15.4	_
ATT-part [4]	VGG-16	×1.0	16.0	56.7
Repulsion Loss [6]	ResNet-50	×1.0	13.2	56.9
Adaptive NMS [12]	VGG-16	×1.0	12.9	56.4
OR-CNN [5]	VGG-16	×1.0	12.8	55.7
LBST [3]	ResNet-50	×1.0	12.8	53.7
AdaptiveNMS + [5]	VGG-16	×1.0	11.9	55.2
F-RCNN(Our impl.)	VGG-16	×1.0	13.8	56.3
F-RCNN + ours	VGG-16	×1.0	**11.3**	**50.5**
Adapted Faster RCNN [27]	VGG-16	×1.3	12.8	_
Repulsion Loss [6]	ResNet-50	×1.3	11.6	56.9
LBST [3]	ResNet-50	×1.3	11.3	50.5
Bi-Box [9]	ResNet-50	×1.3	11.3	50.5
OR-CNN [5]	VGG-16	×1.3	11.0	51.3
AdaptiveNMS+ [12]	VGG-16	×1.3	10.8	54.0
CrowdDet [10]	ResNet-50	×1.3	10.7	_
F-RCNN(Our impl.)	VGG-16	×1.3	11.9	54.8
F-RCNN + ours	VGG-16	×1.3	**10.3**	**49.4**

R stands for the reasonable subsets and **HO** stands for the heavy occlusion subsets.

TABLE VIII

COMPARISON WITH THE STATE-OF-THE-ART METHODS ON CALTECH

Methods	**R**	**HO**
DeepParts [61]	11.9	60.4
F-RCNN +ATT-vbb [4]	10.3	45.2
MS-CNN [31]	10.0	59.9
RPN+BF [62]	9.6	74.4
Adapt F-RCNN [27]	9.2	57.6
Bi-Box [9]	7.6	44.4
SDS-RCNN [63]	7.4	58.6
MGAN [53]	6.8	38.2
F-RCNN (Our impl)	7.8	39.8
F-RCNN + Ours	**6.5**	**36.4**

R stands for the reasonable subsets and **HO** stands for the heavy occlusion subsets.

TABLE IX

COMPARISON WITH DIFFERENT METHODS ON HIEVE DATASET

Methods	JI(%)	AP(%)
F-RCNN [30]	51.56	59.35
Crowddet (our impl.) [10]	51.57	59.79
F-RCNN (our impl.)	50.26	59.81
F-RCNN (our impl.) + ours	**52.13**	**60.72**

Repulsion Loss [6], Adaptive NMS [12], LBST [3], OR-CNN [5], Adaptive NMS + AggLoss [4] and CrowdDet [10] on CityPersons dataset. The comparison results with different input scales are illustrated in Table VII. It should be noted that we only use the full body bounding box information for network training without any extra supervised information such as mask or visible body bounding box.

For the ×1.0 scale of input images, our F-RCNN baseline obtains 13.8% and 56.3% on **R** (reasonable) subsets and **HO** (heavy occlusion) subsets respectively and performs a little better than the F-RCNN in [27]. When our AutoPedestrian scheme is introduced, the MR of **R** subsets and **HO** subsets can be reduced by 2.5% and 5.6% compared to those of the F-RCNN baseline method, respectively, which validates the effectiveness of the proposed scheme and its robustness in heavy occlusion for pedestrian detection. Moreover, it can be noted that our search based scheme achieves the best results of 11.3%, 50.5% in terms of MR on **R** subsets and **HO** subsets respectively, outperforming the top-performed method Adaptive NMS + AggLoss by 0.6% on **R** subsets and 4.7% in terms of MR on **HO** subsets, respectively.

When enlarging the size of the input images as in [27] (i.e., ×1.3 scale), our F-RCNN baseline achieves 11.9% and 54.8% in MR on **R** subsets and **HO** subsets respectively. Besides, our AutoPedestrian can further improve the performance of F-RCNN baseline by 1.6% and 5.4% in MR on **R** subsets and **HO** subsets respectively. Such results indicate the robustness and effectiveness of our AutoPedestrian for different scale of images. In addition, our F-RCNN + AutoPedestrian also outperforms the top-performed method CrowdDet [10] in ×1.3 scale of input images.

G. Comparison With State-of-the-Art Methods on Caltech Dataset

Our AutoPedestrian scheme is also compared with the recent state-of-the-art methods such as DeepParts [61], F-RCNN+ATT-vbb [4], MS-CNN [31], RPN+BF [62], Adapt F-RCNN [27], Bi-Box [9], SDS-RCNN [63] and MGAN [53] on Caltech dataset. The related results are reported in Table VIII. When our AutoPedestrian scheme is introduced, the MR of **R** subsets and **HO** subsets can be reduced by 1.3% and 3.4% compared to those of the F-RCNN baseline method, respectively. What's more, our approach achieves the best results of 6.5%, 36.4% in terms of MR on **R** subsets and **HO** subsets respectively, outperforming the top-performed method MGAN by 0.3% on **R** subsets and 1.8% in terms of MR on **HO** subsets, respectively. Such results validate the effectiveness of the proposed scheme and its robustness in heavy occlusion for pedestrian detection.

H. HiEve Dataset

To investigate the generalization ability of our proposed method in realistic video surveillance scenarios, we further conduct the experiments on HiEve dataset [54]. We make comparisons with the official F-RCNN [54], CrowdDet [10] and our F-RCNN baseline. The related results are reported in Table IX. It can be observed that F-RCNN + our AutoPedestrian scheme achieves the best results of 52.13%, 60.72% in terms of JI and AP. Moreover, when our AutoPedestrian scheme is introduced, the performance of F-RCNN baseline can be improved by 1.87%, 0.91% in terms of JI and AP respectively. Such results confirm the generalization ability and effectiveness of our approach in realistic video surveillance scenarios.

I. Discussions

For the results reported in CityPersons, we do NOT compare our method with some other methods like Bi-Box [9], MGAN [53] and MGAN+ [7] on CityPersons dataset. This is because these kinds of methods not only use full body

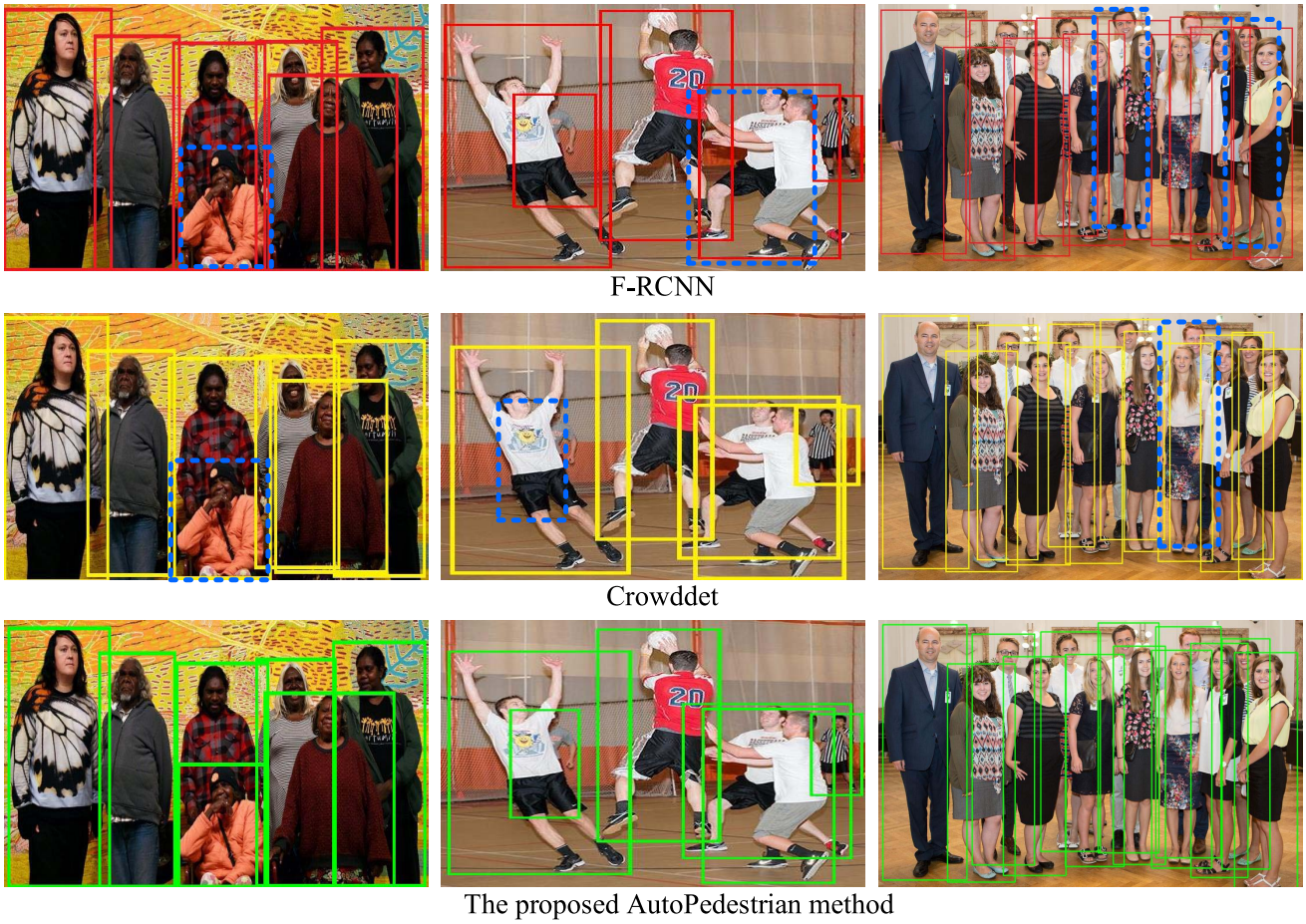

Fig. 7. Visualization of our detection results. The red and yellow bounding boxes represent the detection bounding boxes predicted by F-RCNN [30] and Crowddet [10] respectively, while the green bounding boxes represent the detection bounding boxes predicted by adding our method. The dashed blue bounding boxes are the missed detection ones.

bounding box as ground truth, but also apply the visible body bounding box as extra supervised information. The visible body bounding box provides the extra guidance for detector to predicate the full bounding box. It is unfair to compare with these methods since we only utilize the full body bounding box information. Moreover, some related papers that only use the full bounding box information are also NOT compared with them [3], [10], [12]. Although we use less supervised information, our method can be comparable with MGAN+ (11.3% MR vs 11.0% MR on $\times 1.0$ scale of input images in reasonable subset) and even outperform some methods that utilize the visible body bounding box information, such as Bi-Box (10.3% MR vs 11.2% MR on $\times 1.3$ scale of input images in reasonable subset) and MGAN (11.3% MR vs 11.5% MR on $\times 1$ scale of input images in reasonable subset). As for heavy occlusion subset, we also outperform MGAN by 1.2% in MR for $\times 1.0$ scale of input images and obtain a little better result for $\times 1.3$ scale of input images (49.4% MR and 49.6% MR). Such results indicate that the proposed scheme can effectively overcome the challenge of occlusions. Besides, using the full body bounding box only is more practical in real scenes since the visible body bounding box will cost extra manual work for annotation.

J. Results Visualization

We illustrate some typical detection results on CrowdHuman in this part. As shown in Fig. 7, the F-RCNN baseline may easily miss pedestrians that are occluded by other people, or predict a bounding box that contains two peoples. However, after adding our AutoPedestrian scheme, the F-RCNN baseline can achieve lots of improvement in the situations of crowded scenes and occlusions. It is because our searched augmentation policy can generate some occlusion simulated data including inter-class occlusion and intra-class occlusion for network training. Besides, the searched loss function can also pull the predicted bounding boxes to ground truth and push the predicted bounding boxes away from the neighboring ground-truth boxes without any harm for the accurate predicted bounding boxes. From these illustrations, our method obviously produces a lot of improvement for pedestrian detection in the cases of crowded scenes and occlusions.

V. Conclusion

Pedestrian detection in crowded scenes is a challenging task that still remains unresolved. To overcome this problem, we leverage Auto ML principle to automatically search the

suitable data augmentation policy and loss function policy for pedestrian detection. The choices of the corresponding policies are modeled as distributions with different hyper-parameters. Their suitable search spaces for these hyper-parameters are carefully designed, followed by its solution process that is based on a double-loop scheme with importance sampling. Extensive ablation studies and comparison tests on two benchmark datasets of CrowdHuman and CityPersons show the effectiveness of the proposed AutoPedestrian method, yielding new state-of-the-art results on these two datasets.

REFERENCES

[1] C. Lin, J. Lu, G. Wang, and J. Zhou, "Graininess-aware deep feature learning for robust pedestrian detection," *IEEE Trans. Image Process.*, vol. 29, pp. 3820–3834, 2020.

[2] W. Ouyang, H. Zhou, H. Li, Q. Li, J. Yan, and X. Wang, "Jointly learning deep features, deformable parts, occlusion and classification for pedestrian detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1874–1887, Aug. 2018.

[3] J. Cao, Y. Pang, J. Han, B. Gao, and X. Li, "Taking a look at small-scale pedestrians and occluded pedestrians," *IEEE Trans. Image Process.*, vol. 29, pp. 3143–3152, 2020.

[4] S. Zhang, J. Yang, and B. Schiele, "Occluded pedestrian detection through guided attention in CNNs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6995–7003.

[5] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Occlusion-aware R-CNN: Detecting pedestrians in a crowd," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 637–653.

[6] X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sun, and C. Shen, "Repulsion loss: Detecting pedestrians in a crowd," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7774–7783.

[7] J. Xie, Y. Pang, M. H. Khan, R. M. Anwer, F. S. Khan, and L. Shao, "Mask-guided attention network and occlusion-sensitive hard example mining for occluded pedestrian detection," *IEEE Trans. Image Process.*, vol. 30, pp. 3872–3884, 2021.

[8] X. Wang, M. Liu, D. S. Raychaudhuri, S. Paul, Y. Wang, and A. K. Roy-Chowdhury, "Learning person re-identification models from videos with weak supervision," *IEEE Trans. Image Process.*, vol. 30, pp. 3017–3028, 2021.

[9] C. Zhou and J. Yuan, "Bi-box regression for pedestrian detection and occlusion estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 135–151.

[10] X. Chu, A. Zheng, X. Zhang, and J. Sun, "Detection in crowded scenes: One proposal, multiple predictions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12214–12223.

[11] X. Wang, R. Panda, M. Liu, Y. Wang, and A. K. Roy-Chowdhury, "Exploiting global camera network constraints for unsupervised video person re-identification," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Dec. 9, 2021, doi: 10.1109/TCSVT.2020.3043444.

[12] S. Liu, D. Huang, and Y. Wang, "Adaptive NMS: Refining pedestrian detection in a crowd," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6459–6468.

[13] X. Wang, S. Li, M. Liu, Y. Wang, and A. K. Roy-Chowdhury, "Multi-expert adversarial attack detection in person re-identification using context inconsistency," 2021, *arXiv:2108.09891*. [Online]. Available: http://arxiv.org/abs/2108.09891

[14] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," 2017, *arXiv:1708.04552*. [Online]. Available: http://arxiv.org/abs/1708.04552

[15] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," 2017, *arXiv:1710.09412*. [Online]. Available: http://arxiv.org/abs/1710.09412

[16] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning augmentation strategies from data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 113–123.

[17] C. Lin *et al.*, "Online hyper-parameter learning for auto-augmentation strategy," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6579–6588.

[18] X. Zhang, Q. Wang, J. Zhang, and Z. Zhong, "Adversarial AutoAugment," 2019, *arXiv:1912.11188*. [Online]. Available: http://arxiv.org/abs/1912.11188

[19] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 658–666.

[20] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: An advanced object detection network," in *Proc. 24th ACM Int. Conf. Multimedia*, Oct. 2016, pp. 516–520.

[21] K. Chen, W. Lin, J. Li, J. See, J. Wang, and J. Zou, "AP-loss for accurate one-stage object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jun. 30, 2020, doi: 10.1109/TPAMI.2020.2991457.

[22] R. Cipolla, Y. Gal, and A. Kendall, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7482–7491.

[23] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[24] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, 1992.

[25] C. Li *et al.*, "AM-LFS: AutoML for loss function search," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8410–8419.

[26] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1889–1897.

[27] S. Zhang, R. Benenson, and B. Schiele, "CityPersons: A diverse dataset for pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3213–3221.

[28] S. Shao *et al.*, "CrowdHuman: A benchmark for detecting human in a crowd," 2018, *arXiv:1805.00123*. [Online]. Available: http://arxiv.org/abs/1805.00123

[29] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9756–9765.

[30] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2016.

[31] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[32] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[33] Y. Zhang, H. He, J. Li, Y. Li, J. See, and W. Lin, "Variational pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 11622–11631.

[34] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting masked faces in the wild with LLE-CNNs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 426–434.

[35] S. Ge, C. Li, S. Zhao, and D. Zeng, "Occluded face recognition in the wild by identity-diversity inpainting," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3387–3397, Oct. 2020.

[36] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," 2016, *arXiv:1611.01578*. [Online]. Available: http://arxiv.org/abs/1611.01578

[37] D. Zhou *et al.*, "EcoNAS: Finding proxies for economical neural architecture search," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11396–11404.

[38] Y. Ci, C. Lin, M. Sun, B. Chen, H. Zhang, and W. Ouyang, "Evolving search space for neural architecture search," 2020, *arXiv:2011.10904*. [Online]. Available: http://arxiv.org/abs/2011.10904

[39] B. Chen *et al.*, "GLiT: Neural architecture search for global and local image transformer," 2021, *arXiv:2107.02960*. [Online]. Available: http://arxiv.org/abs/2107.02960

[40] B. Zoph, E. D. Cubuk, G. Ghiasi, T.-Y. Lin, J. Shlens, and Q. V. Le, "Learning data augmentation strategies for object detection," 2019, *arXiv:1906.11172*. [Online]. Available: http://arxiv.org/abs/1906.11172

[41] M. Sun, H. Dou, B. Li, J. Yan, W. Ouyang, and L. Cui, "AutoSampling: Search for effective data sampling schedules," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 9923–9933.

[42] S. Lim, I. Kim, T. Kim, C. Kim, and S. Kim, "Fast autoaugment," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 6665–6675.

[43] L. Pishchulin, A. Jain, C. Wojek, M. Andriluka, T. Thormählen, and B. Schiele, "Learning people detection models from few training samples," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1473–1480.

[44] S. Huang and D. Ramanan, "Expecting the unexpected: Training detectors for unusual pedestrians with adversarial imposters," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2243–2252.
[45] Z. Chen, W. Ouyang, T. Liu, and D. Tao, "A shape transformation-based dataset augmentation framework for pedestrian detection," *Int. J. Comput. Vis.*, vol. 129, no. 4, pp. 1121–1138, Apr. 2021.
[46] X. Wang, S. Wang, C. Chi, S. Zhang, and T. Mei, "Loss function search for face recognition," 2020, *arXiv:2007.06542*. [Online]. Available: http://arxiv.org/abs/2007.06542
[47] B. Colson, P. Marcotte, and G. Savard, "An overview of bilevel optimization," *Ann. Oper. Res.*, vol. 153, no. 1, pp. 235–256, Sep. 2007.
[48] C. Chi *et al.*, "PedHunter: Occlusion robust pedestrian detector in crowded scenes," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 10639–10646.
[49] T. M. M. Versluys, R. A. Foley, and W. J. Skylark, "The influence of leg-to-body ratio, arm-to-body ratio and intra-limb ratio on male human attractiveness," *Roy. Soc. Open Sci.*, vol. 5, no. 5, May 2018, Art. no. 171790.
[50] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[51] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*. [Online]. Available: http://arxiv.org/abs/1707.06347
[52] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.
[53] Y. Pang, J. Xie, M. H. Khan, R. M. Anwer, F. S. Khan, and L. Shao, "Mask-guided attention network for occluded pedestrian detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4967–4975.
[54] W. Lin *et al.*, "Human in events: A large-scale benchmark for human-centric video analysis in complex events," 2020, *arXiv:2005.04490*. [Online]. Available: http://arxiv.org/abs/2005.04490
[55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[56] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[57] K. Chen *et al.*, "MMDetection: Open MMLab detection toolbox and benchmark," 2019, *arXiv:1906.07155*. [Online]. Available: http://arxiv.org/abs/1906.07155
[58] T. Y. Lin, M. Maire, S. Belongie, J. Hays, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[59] X. Huang, Z. Ge, Z. Jie, and O. Yoshie, "NMS by representative region: Towards crowded pedestrian detection by proposal pairing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10750–10759.
[60] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.
[61] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1904–1912.
[62] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 443–457.
[63] G. Brazil, X. Yin, and X. Liu, "Illuminating pedestrians via simultaneous detection and segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4950–4959.

Yi Tang received the bachelor's degree in automation from Hunan University, Changsha, China, in 2019, where he is currently pursuing the Ph.D. degree with the College of Electrical and Information Engineering. His research interests include computer vision, pedestrian detection, and AutoML.

Baopu Li received the Ph.D. degree from The Chinese University of Hong Kong. His current research interests include computer vision, deep learning, and robotics.

Min Liu (Member, IEEE) received the bachelor's degree from Beijing University and the Ph.D. degree in electrical engineering from the University of California at Riverside. He is currently a Professor with the College of Electrical and Information Engineering, Hunan University, China. He is the Deputy Director of the Key Laboratory of Visual Perception and Artificial Intelligence of Hunan Province. He was a Research Intern at HHMI Janelia Farm Research Campus and a Research Scientist at the University of California at Santa Barbara. His research interests include computer vision and biomedical image analysis.

Boyu Chen received the M.Sc. degree in signal and information processing in 2019. His research interests include neural architecture search and AutoML.

Yaonan Wang received the Ph.D. degree in electrical engineering from Hunan University, Changsha, China, in 1994. Since 1995, he has been a Professor with the College of Electrical and Information Engineering, Hunan University. From 1994 to 1995, he was a Postdoctoral Research Fellow with the Normal University of Defense Technology, Changsha. From 1998 to 2000, he was a Senior Humboldt Fellow supported by the Federal Republic of Germany at the University of Bremen, Bremen, Germany, where he was a Visiting Professor from 2001 to 2004. His research interests include robotics and image processing. He is a member of the Chinese Academy of Engineering.

Wanli Ouyang (Senior Member, IEEE) received the Ph.D. degree from the Department of Electronic Engineering, The Chinese University of Hong Kong (CUHK). He is currently an Associate Professor with the School of Electrical and Information Engineering, The University of Sydney, Sydney, Australia. His research interests include image processing, computer vision, and pattern recognition.