A Lightweight Segmentation Network Based on Weak Supervision for COVID-19 Detection

Fangfang Lu[1], Tianxiang Liu[1](✉), Chi Tang[1], Zhihao Zhang[1], Guangtao Zhai[2], Xiongkuo Min[2], and Wei Sun[2]

[1] College of Computer Science and Technology,
Shanghai University of Electric Power, Shanghai, China
lufangfang@shiep.edu.cn,
{liutianxiang,tangchi,zhangzhihao}@mail.shiep.edu.cn

[2] Institute of Image Communication and Network Engineering,
Shanghai Jiao Tong University, Shanghai, China
{zhaiguangtao,minxiongkuo,sunguwei}@sjtu.edu.cn

Abstract. The Coronavirus Disease 2019 (COVID-19) outbreak in late 2019 threatens global health security. Computed tomography (CT) can provide richer information for the diagnosis and treatment of COVID-19. Unfortunately, labeling of COVID-19 lesion chest CT images is an expensive affair. We solved the challenge of chest CT labeling by simply marking point annotations to the lesion areas, i.e., by marking individual pixels for each lesion area in the chest CT scan. It takes only a few seconds to complete the labeling using this labeling strategy. We also designed a lightweight segmentation model with approximately 10% of the number of model parameters of the conventional model. So, the proposed model segmented the lesions of a single image in only 0.05 s. In order to obtain the shape and size of lesions from point labels, the convex-hull based segmentation (CHS) loss function is proposed in this paper, which enables the model to obtain an approximate fully supervised performance on point labels. The experiments were compared with the current state-of-the-art (SOTA) point label segmentation methods on the COVID-19-CT-Seg dataset, and our model showed a large improvement: IoU improved by 28.85%, DSC improved by 28.91%, Sens improved by 13.75%, Spes improved by 1.18%, and MAE decreased by 1.10%. Experiments on the dataset show that the proposed model combines the advantages of lightweight and weak supervision, resulting in more accurate COVID-19 lesion segmentation results while having only a 10% performance difference with the fully supervised approach.

Keywords: Weakly supervised segmentation · Lightweight · COVID-19

G. Zhai et al. (Eds.): IFTC 2022, CCIS 1766, pp. 87–105, 2023.
https://doi.org/10.1007/978-981-99-0856-1_7

1 Introduction

An outbreak of COVID-19 began in December 2019 and has since spread throughout the world [1]. In early 2020, the virus had spread to 33 countries and had begun a global pandemic. As of October 27 2022, more than 620 million confirmed cases of COVID-19 have been reported worldwide, with more than 6.5 million deaths, accounting for 1.0% of confirmed cases [2]. Clinical manifestations of the disease range from asymptomatic to the more severe acute respiratory distress syndrome (ARDS). The most common symptoms are fever, dry cough, and malaise. A small number of patients will develop dyspnea. Reverse transcription polymerase chain reaction (RT-PCR) is the gold standard for the diagnosis of COVID-19, but the quality of the RT-PCR sampling process affects the positive rate of the test, and RT-PCR requires a strict testing environment and testing equipment to ensure the correctness of the test results.

Although RT-PCR is the gold standard for COVID-19 diagnosis, its relatively low sensitivity and high specificity can lead to negative RT-PCR results in early-stage patients with no obvious symptoms [3]. The COVID-19 detection sensitivity of chest CT scan is relatively high. Chest CT scans accurately reflect the severity of COVID-19 patients, and radiologists can determine the type of patient based on the chest CT scan changes. Consequently, a chest CT scan can assist in the diagnosis of COVID-19. It has been shown to be effective in diagnosing the disease, as well as determining the prognosis for recovered patients, making it an important adjunct to the gold standard [4].

Typically, asymptomatic or mildly patients have no additional clinical symptoms. When the disease reaches a moderate or advanced stage, the chest CT image reveals a slight increase in lung tissue density, which is less than consolidation, with a blurred, cloudy appearance; however, the internal blood vessels and bronchial tissue are still visible, a phenomenon known as ground grass opacity (GGO). Additionally, chest CT scans can detect subpleural patchy shadows and interstitial pneumonia [3]. Pulmonary fibrosis can also be detected on CT scans of severe patients [5,6].

During the COVID-19 pandemic, there was a lack of experienced radiologists due to the high medical demand and physician shortage. Therefore, we required a computer-assisted system to assist radiologists in automatically analyzing chest CT scans and rapidly segmenting lesion areas to provide diagnostic clues to physicians, thereby alleviating the difficult problems of medical pressure and the shortage of physicians, which played a crucial role in preventing the spread of COVID-19 and treating patients promptly.

Fortunately, there have been numerous successful applications of deep learning techniques for computer-assisted COVID-19 screening. However, the use of conventional deep learning models to aid physicians in COVID-19 screening is suboptimal. Existing advanced models typically have a large number of parameters and calculation, which can lead to easy overfitting, slow inference, and inefficient deployment of the models, and is not conducive to practical applications;Second, the majority of the most widely used and effective models are fully supervised methods, which require complete labeling. However, labeled public datasets are scarce, and obtaining the complete labeling is laborious and time-consuming.

Given that we require a rapid screening method to effectively relieve medical pressure and to rapidly screen for COVID-19, it is urgent to design a COVID-19 screening system with low computation and rapid inference. In this paper, we are inspired by Gao et al. [7] to reduce the number of model channels to 256 and reduce the number of model parameters by group convolution and point-wise convolution using D-Block, a dilation convolution block. In order to avoid the problem of detail loss and model performance degradation caused by reducing the number of covariates, we also introduce the dilated convolution module D block, which enlarges the model's receptive field by adjusting the dilation rate. This improves the model's ability to learn multi-scale information without adding extra parameters, and we also add residual connections to address the issue of model performance degradation.

To compensate for the lack of publicly available data, we trained the model using point labels. According to Ma et al. [8], the fully supervised label annotator requires an average of 1.6 min to label a CT slice, however, the point label data annotator requires only 3 s to label a single pixel point in a lesion region, which greatly reduces the difficulty of acquiring point label data. However, point labels lack semantic information such as shape and size, and the prediction accuracy of the fully supervised model trained by them is low, making it challenging to use point labels to complete the semantic segmentation task effectively. Meanwhile, when the model is trained using a point-level loss function, the loss function only encourages the model to supervise a small number of pixel points, resulting in the model only predicting a very small region. Therefore, we refer to the novel loss function as convex-hull-based segmentation loss (CHS), which encourages the model to make accurate predictions by obtaining shape information from a small number of points in lesion regions. Our experiments on the point-annotation dataset demonstrate that our method outperforms the conventional point-level loss function, and we also demonstrate that this weakly supervised method performs similarly to the fully supervised method.

Our contribution is shown as follows:

1. In combination with D-Block, we proposed a weakly-point D-block network (WPDNet) for efficient and rapid segmentation of COVID-19 lesions on point-annotation datasets.
2. We propose the CHS loss function for point-annotation datasets, which can provide semantic information such as shape and size to the model and enhance segmentation performance.
3. We offer a method for labeling points that requires marking only a small number of points in each lesion area.

2 Related Work

In this section, we review the methods to accomplish COVID-19 segmentation on chest CT scans. Then, we discuss lightweight models and weakly supervised segmentation methods.

2.1 COVID-19 Segmentation Methodology

In recent years, the COVID-19 lesion segmentation method has become one of the most popular tasks within the field of medical image analysis due to its high application value. This task classifies each pixel of a chest CT image as either background or lesion, typically segmenting the lesion area from each CT slice, thereby providing the physician with the information required to diagnose the disease. Traditional lesion segmentation methods segment images using features such as the boundary gray gradient, gray value threshold, and image region.

Among them, the threshold-based [9–11] segmentation method utilizes the contrast information of CT images, which is the quickest but has a tendency to miss abnormal tissues and is suitable for segmenting images with a clear contrast between the object and the background. Region-based [12] segmentation methods are quick and produce more accurate segmentation results, but they are susceptible to noise, which can result in over segmentation and contour loss. Due to the robust feature extraction capability of deep learning, its image segmentation results are vastly superior to those of conventional methods; consequently, COVID-19 segmentation methods based on deep learning are widely used.

Due to the large variation of lesion size in COVID-19 chest CT images, multi-scale learning plays a crucial role in COVID-19 segmentation. Therefore, SOTA deep learning methods aim to design fully convolutional networks to learn multi-scale semantic information. Currently, U-Net [13] is one of the most popular medical image segmentation models. It proposes a symmetric encoder-decoder structure that employs skip-connections to fuse multi-scale information at different stages. Numerous COVID-19 segmentation methods are based on U-Net or its variants (Unet++ [14], Vnet [15], Attention-Unet [16], VBnet [17]). Wu et al. [18] designed a U-shaped COVID-19 segmentation network and proposed Enhanced Feature Module (EFM) and Attentional Feature Fusion (AFF) to improve the network feature representation, achieving a Dice score of 78.5%. Paluru et al. [19] proposed COVID-19 segmentation network with symmetric encoder-decoder structure using skip-connections to fuse multi-scale features at different levels to improve network performance, and the Dice score of this network was 79.8%.

The DeepLabs V2 [20] based segmentation methods has superior multi-scale learning capability because it uses the Atrous Spatial Pyramid Pooling (ASPP) module, which is comprised of dilated convolution with different dilatation rates, to learn abundant multi-scale semantic information from the input image. Xiao et al. [21] proposed SAUNet++ network segmentation COVID-19 based on Unet and Unet++ using ASPP and squeeze excitation residual (SER) modules and obtained a Dice score of 87.38%. In addition, Gao et al. [7] rethink the strategy for expanding the network's receptive field by designing a structure with two parallel 3×3 convolutions with different dilation rates and repeating this structure in the backbone to expand the network's receptive field without adding a context module after the backbone, while preserving the local information. Enshaei et al. [22] improved the model's multi-scale learning capability by incorporating the Content Perception Boosting Module (CPB) and achieved a Dice score performance of 80.69%.

In addition to the ROI (region of interest), there is a lot of redundant information in chest CT scan images. An attention mechanism is used to address this issue by instructing the models to concentrate on the most important data. Raj et al. [23] suggested using DenseNet rather than standard convolution and Attention Gate (AG) to ignore the background and increase segmentation accuracy. OuYang et al. [24] proposed a new dual-sampling attention network to automatically diagnose COVID-19, in which the attention module can effectively mitigate the imbalance problem in chest CT images. Feature fusion can synthesize multiple image features for complementary information and more robust and precise segmentation results. Shi et al. [25] trained 3DU-Net and 2D-UNet models with directional fields to fuse segment lesion features. Wu et al. [18] proposed a joint classification and segmentation (JCS) system for diagnosis that enhances robustness by fusing classification network information via the AFF module. These fully supervised methods are computationally intensive and time-consuming to deploy, despite their high accuracy. Some lightweight models are proposed to resolve this issue.

2.2 Lightweight Model

The design of lightweight models can solve two efficiency issues: (a) storage issues, as ordinary models require a great deal of storage space; (b) speed issues, as ordinary models are second-level and do not meet practical application requirements.

A common technique for lightweight design is model compression, in which trained models are pruned and quantized to solve storage and speed issues problems, but at the expense of performance. In recent years, the lightweight design of models has been considered mainly from a "computational approach" perspective. The "computational approach" for convolutional operations is divided into (a) spatial-based convolutional operations [26–28] and (b) shift-based convolutional operations [29,30], which reduces the number of parameters without affecting network performance. The shift convolution requires no additional parameters or operations and uses 1×1 convolution to simulate each type of convolution to reduce the number of parameters and computational complexity. Nevertheless, displacement convolution necessitates high-end hardware, and training results are not always optimal. Therefore, spatially-based convolution parameter methods are more common. Common spatial convolution operations include dilated convolution and depth wise separable convolution.

To achieve parametric reduction, many well-known network architectures, such as Inception networks [31], Xception [32], and MobileNets [33], use deep separable convolution. Among them, MobileNet V2 uses point-wise convolution (1×1 convolution) to further reduce the number of parameters and downscale the input feature channels. ESPNetV2 [28] employs depth wise separable dilated convolution instead of depth wise separable convolution and hierarchical feature fusion (HFF) to eliminate the grid residual problem, thereby reducing the network's computational complexity and enhancing its receptive field. Miniseg [34] employs dilated convolution in conjunction with point-wise convolution to decrease the model parameters while expanding the model's receptive field. Anam-Net [19] reduces the parameters using point-wise convolution and adds residual connections to prevent network performance degradation. Unfortunately, these methods require fully supervised labels, yet publicly available datasets are rare. To solve this problem, weakly supervised semantic segmentation is proposed.

2.3 Weakly Supervised Semantic Segmentation

Weakly supervised semantic segmentation can reduce the reliance on fully supervised labels. The common weakly supervised labels used for semantic segmentation are (a) image-level label [35,36]; (b) scribble label [37]; (c) bounding boxes [38,39]label; and (d) point label [40,41].

Weakly supervised segmentation utilizing image-level labels typically employs Class Activation Maps (CAM) to obtain initial lesion localization; however, this initial localization is imprecise and the final segmentation accuracy is low. Since CAM at different scales requires complex network structures and extensive post-processing, to solve this problem, Tang et al. [36] proposed the dual weakly supervised segmentation method M-SEAM-NAM, which proposes a Self-supervised Equivalent Attention Mechanism (SEAM) with Neighborhood Affinity Module (NAM) for exact segmentation. However, M-SEAM-NAM still has the problem of easily identifying the thoracic skeleton as a lesion. Scribble labels perform better than CAM in COVID-19 segmentation, but their acquisition time is relatively lengthy and there are no standard setting criteria. The acquisition of point labels for each image takes only 22.1 s, whereas the acquisition of fully annotated takes 239 s, which is an order of magnitude faster than the acquisition of point labels [42]. Laradji et al. [40] trained a COVID-19 segmentation network based on weakly supervised consistent learning (WSCL) on point labels, but the network showed over-segmentation. In order to solve this problem, this paper implements WPDNet and provides a point label setting method to complete COVID-19 semantic segmentation using weak supervision.

3 Materials and Methods

In this section, we first introduce the dataset used in this paper and then give the point label setting method. Then, the structure of our proposed WPDNet model is described in detail. Finally, our proposed loss function CHS for weakly supervised point label segmentation is introduced.

3.1 Data Collection

Since public datasets are relatively scarce and the majority of COVID-19 segmentation studies are based on private datasets, it is crucial to collect public datasets and evaluate the performance of various models.

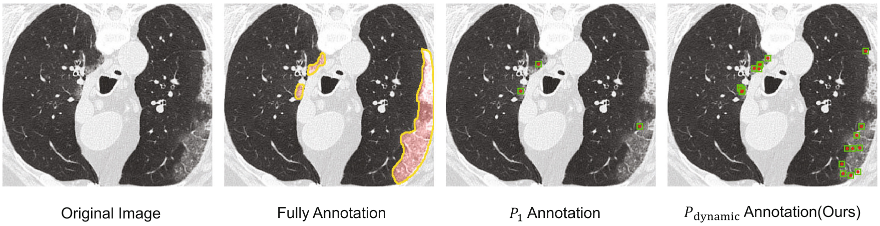

Fig. 1. Different annotations strategy. We demonstrate the distinction between fully supervised labeling, P_1 labeling strategy, and $P_{dynamic}$ (ours) labeling strategy. Several points within the lesion region are labelled.

Dataset A. This dataset [43] contains the results of 110 CT scans of over 40 patient outcomes; the original images were downloaded from the public dataset of the Italian Society of Medical and Interventional Radiology and annotated using MedSeg by three radiologists.

Dataset B. This dataset [44] consists of over 800 CT slices from 9 patients on Radiopaedia, of which approximately 350 slices were determined positive by the radiologist and annotated using MedSeg, and over 700 slices were labeled for the lung parenchyma.

Dataset C. The dataset [45] consists of CT scans of 20 patients. Two radiologists labeled CT images of the left lung parenchyma, right lung parenchyma, and infected areas, which were then validated by experienced radiologists.

Dataset D. This dataset [46] contains anonymous chest CT scans provided by the Moscow Municipal Hospital in Moscow, Russia, with 50 labeled chest CT scans.

Point Label Setting. As point labeling gains popularity, there are already point labeling setup strategies [47,48] in place. Typically, they sample pixel points within the ROI and classify them as objects or backgrounds. Previous sampling strategies were manual [42,49,50] clicks by markers, but manual clicks are more subjective, and when the contrast between COVID-19 lesions and the surrounding background is low (especially in mild patients), markers may over focus on areas of significant contrast, which may result in monolithic data. According to Bearman et al. [51], random points on ground truth (GT) are more suitable than manual clicks for training weakly segmentation models.

The most common way to label is to mark one pixel in the center of the lesion, which is called P_1. COVID-19 lesion size varies frequently, and we want to reflect the lesion size on point labels. Therefore, we use a random sampling strategy to dynamically set the sampling points according to the area of each lesion region: two points are sampled when the number of lesion pixels is less than 288; four points are sampled when the number of lesion pixels is between 228 and 2386; and nine points are sampled when the number of lesion pixels is greater than 2386 (228 and 2386 are the lower quartile and upper quartile of all lesion areas, respectively). As depicted in Fig. 1, we refer to this method as $P_{dynamic}$ and mark only the focal pixels while ignoring the background pixels.

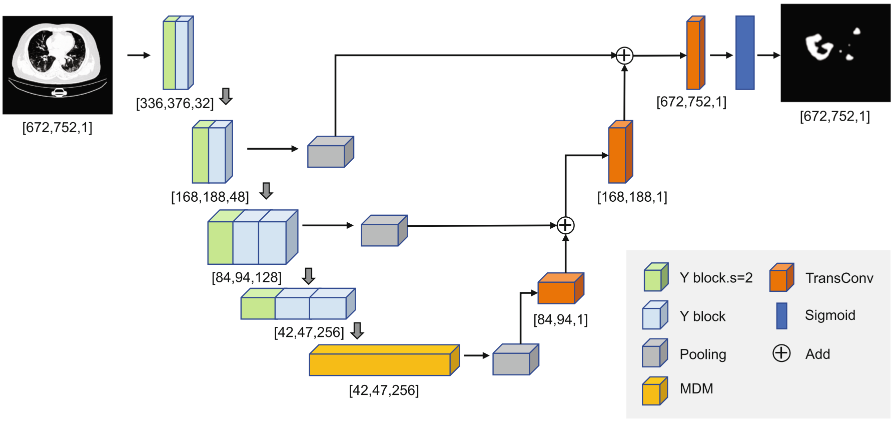

Fig. 2. Network structure. 10 Y blocks are used for feature extraction, where green block represents Y block when s = 2, and blue block represents Y block. D block is set with different dilation rates to improve the model receptive field. (Color figure online)

3.2 Network Structure

Our network backbone integrates the Y block and Multi-Scale D-Block Module (MDM), as shown in Fig. 2. In this section, we describe the main components of the model: the backbone and the MDM.

Backbone. The WPDNet network consists of five layers; the first four layers use the Y block in RegNet [52] to extract features, while the fifth layer employs the MDM module to expand the model's receptive field. As shown in Fig. 3, the Y block adds the SE block attention mechanism to remove redundant information after standard convolution and then uses point-wise convolution to combine weighted features in depth direction and adds point-wise convolution on residual connections to avoid noise interference while eliminating network performance degradation. The standard kernel size for convolutional is 3×3, and the number of channels increases from 32 to 256. In order to obtain spatial information at downsampling while boosting the size invariance of the encoder, we use a convolution with stride = 2 in the Y block while adding an average pooling with stride = 2 and a 1×1 convolution on the residual connection. This paper adds Multi-Scale D-block Module (MDM) to expand the encoder's receptive field without

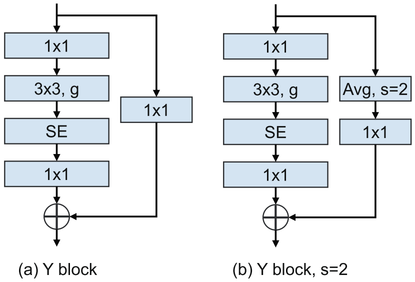

Fig. 3. Y block structure

sacrificing detail, as different receptive fields are required to extract features from different regions of an image. Finally, the segmentation results are obtained by fusing the features of different layers of the backbone during upsampling and restoring them to the original resolution.

Table 1. Multi-scale D-block module

Module	d1	Stride	Channels
MDM_1	2	1	256
MDM_4	4	1	256
MDM_6	8	1	256

Multi-scale D-Block Module. MDM consists primarily of D blocks; depending on how MDM is utilized at different network layers, different dilation rates and numbers of D blocks are set. We assume that the D block is repeated N times in MDM, we denote this as MDM_N, where $N \in \{1, 4, 6\}$. The D block is comprised of a parallel standard convolution and a dilated convolution, where the dilation rate of the dilated convolution can be set to various values depending on the MDM_N in which it is located, Table 1 shows their detailed parameter settings. When the stride of the D block is 2, the average pooling operation with stride = 2 is used at the jump connection. Figure 4 illustrates the MDM_4 with the D block structure.

3.3 Loss Function

Localization-based counting loss [47] (LC) is comprised of four components: image-level, point-level, split-level, and false-positive. Its primary application scenario is instance localization and counting, but it can force the model to predict the semantic segmentation label of each pixel in the image. To encourage

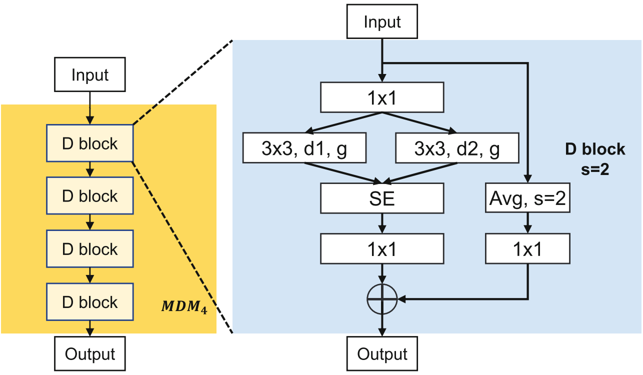

Fig. 4. D block structure.

the model to learn lesion information through point labels, the loss function CHS proposed in this paper is modified based on the LC loss function by adding convex-hull supervision and removing split-level supervision.

Given the output matrix X of the network after softmax, where X_{iy} denotes the probability that pixel i belongs to class y (lesion or background), Y denotes the point label matrix, where the object's position is 1 and all other positions are 0. Our proposed loss function can be expressed as follows:

$$\mathcal{L}(X,Y) = \underbrace{\mathcal{L}_I(X,Y)}_{\text{Image-level loss}} + \underbrace{\mathcal{L}_P(X,Y)}_{\text{Point-level loss}} + \underbrace{\mathcal{L}_F(X,Y)}_{\text{False Positive loss}} + \underbrace{\mathcal{L}_{CH}(X,Y)}_{\text{Convex Hull loss}} \quad (1)$$

Image-Level Loss. Let $\mathcal{C}$ denote the set of classes present in the image, the $\mathcal{C}^{'}$ denote the set of classes not present in the image. The image-level loss is shown below:

$$\mathcal{L}_I(X,Y) = -\frac{1}{|\mathcal{C}|}\sum_{y\in\mathcal{C}}\log(X_{t_y y}) - \frac{1}{|\mathcal{C}'|}\sum_{y\in\mathcal{C}'}\log(1 - X_{t_y y}) \quad (2)$$

where $t_y = \arg\max_{j\in\mathcal{I}_a} X_{jy}$, $\mathcal{I}_a$ denote all pixels in the image. For each category present in the $\mathcal{C}$, at least one pixel should be labeled as that class; For each category present in the $\mathcal{C}^{'}$, none of the pixels should belong to that class.

Point-Level Loss. $\mathcal{I}$ denotes the set of points in the point label. We apply the standard cross-entropy (CE) loss function to encourage the model to correctly predict the set of pixel points in $\mathcal{I}$. The point-level loss is shown below:

$$\mathcal{L}_P(X,Y) = -\sum_{j\in\mathcal{I}}\log(X_{jY_j}) \quad (3)$$

where X_{jY_j} denotes the probability that pixel j in the output matrix X belongs to the class of pixel j in the GT of the point label.

False Positive Loss. To reduce the number of false positive lesions, $\mathcal{L}_F$ discourages the network from predicting lesion regions without point annotations. The loss function is described as follows:

$$\mathcal{L}_F(X, Y) = -\sum_{j \in B} \log(X_{j0}) \tag{4}$$

where B represents the collection of all connected regions that lack point annotations and X_{i0} represents the probability that pixel i belongs to the background.

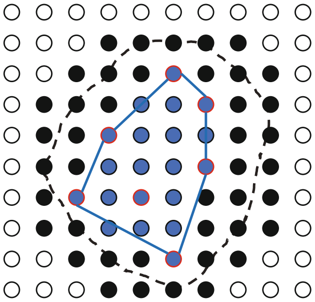

Fig. 5. Convex-Hull Loss Function. In a 10 × 10 network prediction output (the darker color represents the lesion prediction), the red color represents the GT of the point label, the blue line represents the boundary line obtained by convex hull of the point labels, and the blue pixels are all included in CH_i (the gray dashed line is only to suggest the true lesion area, it does not play any role in the actual work). (Color figure online)

Convex-Hull Loss. We define the set CH_i, $i \in \{1, \ldots, N\}$ where n is the number of lesions in the image. It is the collection of convex hull pixels generated by point labels in each lesion region, as depicted in Fig. 5. Then, we use a CE loss function to encourage the model's segmentation results to be as close as possible to the GT. The loss function is described as follows:

$$\mathcal{L}_{CH}(X, Y) = -\sum_{i=1}^{N} \sum_{j \in CH_i} \log(X_{jY_j}) \tag{5}$$

4 Experiments

In this section, we describe our experimental setup, evaluation metrics, and comparison experiments in detail.

4.1 Experimental Setup

Implementation Detail. We implemented the WPDNet model using the Pytorch framework and trained it on an Nvidia RTX 2080 Ti 11 GB GPU using the Adam optimizer, with a 1e−4 weight decay. 300 epochs were trained with the initial learning rate set to 1e−5 and the batch size set to 2. We use the same settings to train other networks on Dataset C to ensure fairness.

Evaluation Metrics. In this paper, we evaluated the COVID-19 segmentation methods using the following metrics:

Intersection over Union(IoU). To calculate the cross-merge ratio between the predicted and the GT:$IoU = \frac{TP}{TP+FP+FN}$, where TP, FP, and FN are the numbers of pixels of true positive, false positive, and false negative, respectively.

Dice Similarity Coefficient (DSC). Similar to IoU, DSC also calculates the similarity between prediction and GT, but the coefficient of TP in DSC is 2 rather than 1:$DSC = \frac{2*TP}{2*TP+FP+FN}$.

Sensitivity (Recall). To quantify the probability that the model predicts a lesion with a true positive GT:$Sens. = \frac{TP}{TP+TN}$.

Specificity. Measures the percentage of true negative in correct predictions: $Spec. = \frac{TN}{TN+FP}$.

4.2 Comparison

We compare the performance of the point-supervised loss function on various point label setting strategies, test the performance of our proposed loss function using different backbone networks. Finally, compare the WPDNet with image segmentation networks to conclude.

Table 2. Comparison of different point labeling strategies. The effect of different point labeling strategies on segmentation results was tested using our method (WPDNet) with FCN-8S, and the bolded part is our method.

Methods	Loss functions	Points strategy	Dice(%)	IoU(%)	Sens.(%)	Spec.(%)
FCN-8s	LC	P_1	67.74	60.09	63.99	99.95
FCN-8s	LC	$P_{dynamic}$	76.14	68.78	73.92	99.92
WPDNet	LC	P_1	69.33	61.80	66.41	99.94
WPDNet (Ours)	LC	$P_{dynamic}$	**76.18**	**69.01**	**79.14**	**99.80**

Points Number. As shown in Table 2, we evaluated the performance of two models, FCN-8s and WPDNet, using the same loss function for the two point labeling strategies. Notably, because the P_1 strategy cannot utilize the convex hull supervision in CHS, we evaluate the performance of both strategies using LC as a fairness benchmark.P_1 only labels a single pixel at the center of the lesion, whereas $P_{dynamic}$ labels points according to the area of each lesion, which can laterally reflect the size and shape of the lesion. Hence, Table 2 shows that our $P_{dynamic}$ strategy is almost comprehensively ahead of the P_1 strategy in both methods, with a maximum improvement of about 12% in the sensitivity metric when using WPDNet as the backbone and a different degree of lead for all other metrics.

Table 3. Comparison of different loss functions. The effect of LC and our loss function (CHS) on segmentation results was tested using our method (WPDNet) with FCN-8S, and each model's significant values are highlighted in bold.

Methods		Loss functions		Metrics(%)			
FCN-8s	WPDNet	LC	CHS	DSC	IoU	Sens.	Spec.
✓		✓		76.14	68.78	73.92	**99.92**
✓			✓	**80.15**	**73.18**	**88.32**	99.63
	✓	✓		76.18	69.01	79.14	**99.80**
	✓		✓	**81.00**	**73.44**	**95.44**	99.53

Loss Functions. As shown in Table 3, we evaluated the effect of our CHS loss function on FCN-8s and WPDNet, respectively, and found that our loss function provided superior performance compared to the point-supervised loss function LC. When FCN-8s used CHS, DSC, IoU, and Sens. improved by 4.01%, 4.4%, and 14.4%, respectively. With a maximum of 16.3%, the Sens. improvement was greatest when the WPDNet utilized CHS. Our CHS loss results in a more competitive performance for other metrics.

Figure 6 visualizes the segmentation results, comparing the LC loss with the CHS loss. Due to the fact that CHS adds convex-hull loss to LC to provide size and shape information to the model and removes split-level loss, it does not force the model to segment connected regions containing two or more point labels, allowing the network to learn more about the lesion. It can be observed that CHS improves the problem of more false negatives in LC and has better true positive results.

Quantitative Evaluation. To compare with SOTA methods, we constructed a performance comparison experiment using three fully supervised image segmentation models and two weakly supervised segmentation models, including Unet, AnamNet, COVID-Rate, WSCL, and M-SAME-NAM, COVID-Rate, AnamNet,

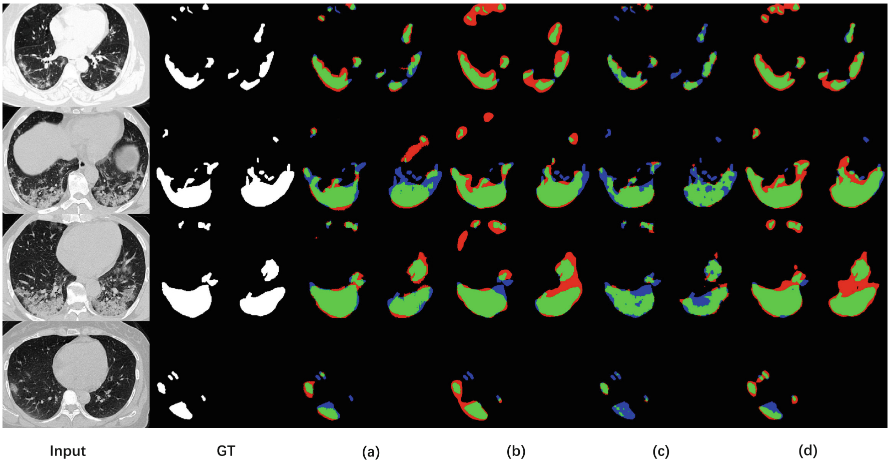

Fig. 6. Effect of loss function on segmentation performance. (a) WPDNet+LC; (b) WPDNet+CHS; (c) FCN-8s+LC; (d) FCN-8s+CHS; Red: false positive; green: true positive; blue: false negative. (Color figure online)

WSCL, and M-SAME-NAM are specifically designed for COVID-19 lesion segmentation, whereas M-SAME-NAM and WSCL are weakly supervised networks trained with classification labels and point labels, respectively.

Table 4. Compare the number of parameters, FLOPs, and inference speed of WPDNet with other SOTA methods.

Method	Backbone	#Param	FLOPs	Speed
Unet	–	31.04M	421.49G	16.3 fps
AnamNet	–	4.47M	196.12G	29.0 fps
COVID-Rate	–	14.38M	297.92G	19.2 fps
WSCL	FCN-8s	134.26M	317.25G	20.2 fps
M-SAME-NAM	ResNet38	106.40M	403.78G	2.8 fps
WPDNet	–	**11.03M**	**21.96G**	**24.6 fps**

The results of WPDNet compared with other networks in terms of number of parameters, FLOPs, and speed are shown in Table 4. These fully supervised segmentation networks (Unet, FCN8s, and Covid-Rate) have high accuracy, but the number of parameters and FLOPs of these methods are relatively high. Using more skip connections affects inference speed. The FLOPs of Unet reached 421 GFLOPs, and the inference speed was only 16.3 FPS (frames per second), while the FLOPs of Covid-Rate were only about half of those of Unet, but the inference speed was only increased by 2.9 FPS. AnamNet, a lightweight segmentation network, achieves the fastest inference speed of 29 FPS, but it requires fully supervised labeling, and the accuracy of the weakly supervised method presented in this paper is not significantly different. In the training phase, WSCL requires two FCN8s networks, so the training time is lengthy. Even though only one FCN8s is required to complete segmentation in the inference stage, the network's FLOPs still reach 317 GFLOPs, and the inference speed of 20 FPS is not dominant. M-SEAM-NAM needs to calculate the CAM and affinity matrix during inference and use the random walk algorithm to get the result, so its inference speed is only 2.8 FPS. WPDNet uses point-wise and D block to reduce the model parameters, and the FLOPs are only 21.96 GFLOPs while the inference speed reaches 24 FPS.

Table 5. Comparison of WPDNet with the advanced COVID-19 segmentation method, where the bolded black is the method in this paper.

Label	Methods	DSC(%)	IoU(%)	Sens.(%)	Spec.(%)
Full	Unet	87.58	82.23	86.11	99.93
	AnamNet	86.83	77.27	94.10	99.67
	COVID-Rate	83.79	80.13	88.22	99.85
Point	WSCL	52.09	44.59	81.69	98.35
Classification	M-SAME-NAM	51.00	50.13	53.50	80.19
Point	WPDNet **(Ours)**	**81.00**	**73.44**	**95.44**	**99.53**

A comparison of the evaluation metrics of WPDNet with the above networks is shown in Table 5. On DataSet C, WPDNet achieved the best results compared to other weakly supervised methods, with DSC and IOU leading by more than 29% and 28%, respectively, demonstrating the effectiveness of the weakly supervised method in this paper. Compared with the fully supervised method, the performance of WPDNet is slightly inferior; from the IoU point of view, the lowest difference of the method in this paper is only 3.8%, but the performance of Sens. in this paper is 9.3%, 1.3%, and 7.2%, respectively. Therefore, we can conclude that our method is rapid, effective, precise, simple to implement in practice, and crucial for assisting in the diagnosis of COVID-19.

Qualitative Comparison. In order to demonstrate the efficacy of WPDNet, this paper compares visually two weakly supervised methods and one fully supervised method. As depicted in Fig. 7, we chose some representative images from Dataset C for comparison, and it can be seen that the WSCL method suffers from over segmentation and that the numerous lesion regions are nearly connected. M-SEAM-NAM uses CAM combined with the affinity's segmentation method. Even though M-SEAM-NAM adds an attention mechanism for CAM to focus more on the lesion, it is still easy to misidentify the thoracic skeleton as a lesion. The segmentation results of this paper's method have the fewest false

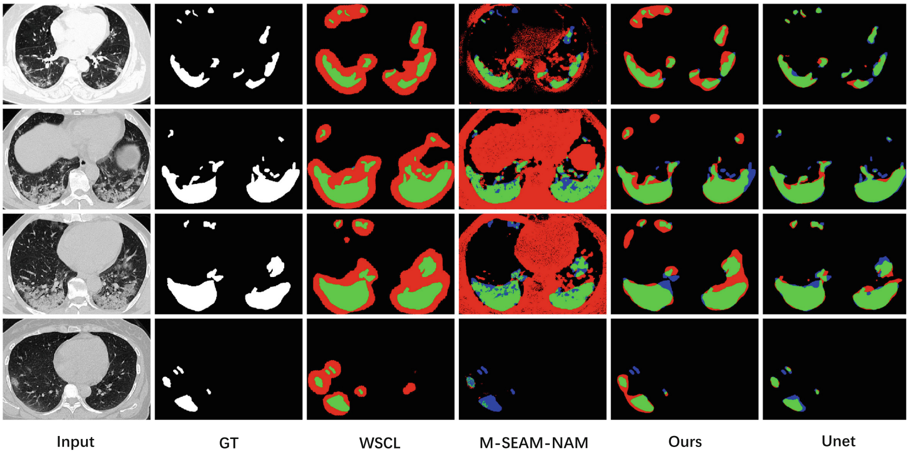

Fig. 7. The visualization comparison graphs of different segmentation methods. Red: false positive; green: true positive; blue: false negative. (Color figure online)

positives and are closest to GT, which is superior to the segmentation results of other weakly supervised methods. Compared to Unet, the segmentation results of this paper's method produce relatively more false positives, but are otherwise comparable to Unet.

5 Conclusion

Deep learning can lead to a detection and segmentation solution for COVID-19. In this paper, a lightweight, weakly-supervised point-label COVID-19 segmentation network called WPDNet is proposed.WPDNet fuses the D block into the network to expand the model's receptive field. The CHS loss function was designed based on LC to learn the shape and size of lesions under point labels.This paper presents a dynamic setting method for point labels, and experiments demonstrate that this method achieves the highest segmentation accuracy compared to the conventional method.In this paper, we used point-wise convolution and reduced the number of model channels, which significantly reduced the FLOPs of the model. Compared to other weakly supervised segmentations, the method described in this paper achieves the highest segmentation accuracy and is suitable for rapid COVID-19 lesion segmentation deployment.

References

1. Huang, C., et al.: Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. Lancet **395**(10223), 497–506 (2020)
2. Dong, E., Du, H., Gardner, L.: An interactive web-based dashboard to track COVID-19 in real time. Lancet. Infect. Dis **20**(5), 533–534 (2020)

3. Ai, T., et al.: Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in china: a report of 1014 cases. Radiology **296**, E32–E40 (2020)
4. Zu, Z.Y., et al.: Coronavirus disease 2019 (COVID-19): a perspective from china. Radiology **296**(2), E15–E25 (2020)
5. Iqbal, A., et al.: The COVID-19 sequelae: a cross-sectional evaluation of post-recovery symptoms and the need for rehabilitation of COVID-19 survivors. Cureus **13**(2), e13080 (2021)
6. Froidure, A., et al.: Integrative respiratory follow-up of severe COVID-19 reveals common functional and lung imaging sequelae. Respir. Med. **181**, 106383 (2021)
7. Gao, R.: Rethink dilated convolution for real-time semantic segmentation. arXiv preprint arXiv:2111.09957 (2021)
8. Ma, J., et al.: Towards efficient COVID-19 CT annotation: a benchmark for lung and infection segmentation (2020). https://arxiv.org/abs/2004.12537v1
9. Abualigah, L., Diabat, A., Sumari, P., Gandomi, A.H.: A novel evolutionary arithmetic optimization algorithm for multilevel thresholding segmentation of COVID-19 CT images. Processes **9**(7), 1155 (2021)
10. Shen, C., et al.: Quantitative computed tomography analysis for stratifying the severity of coronavirus disease 2019. J. Pharm. Anal. **10**(2), 123–129 (2020)
11. Oulefki, A., Agaian, S., Trongtirakul, T., Laouar, A.K.: Automatic COVID-19 lung infected region segmentation and measurement using CT-scans images. Pattern Recogn. **114**, 107747 (2021)
12. Joshi, A., Khan, M.S., Soomro, S., Niaz, A., Han, B.S., Choi, K.N.: SRIS: saliency-based region detection and image segmentation of COVID-19 infected cases. IEEE Access **8**, 190487–190503 (2020)
13. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
14. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: redesigning skip connections to exploit multiscale features in image segmentation. IEEE Trans. Med. Imaging **39**(6), 1856–1867 (2019)
15. Milletari, F., Navab, N., Ahmadi, S.A.: V-Net: fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571. IEEE (2016)
16. Oktay, O., et al.: Attention U-Net: learning where to look for the pancreas. arXiv preprint arXiv:1804.03999 (2018)
17. Han, M., et al.: Segmentation of Ct thoracic organs by multi-resolution VB-Nets. In: SegTHOR@ ISBI (2019)
18. Wu, Y.H., et al.: JCS: an explainable COVID-19 diagnosis system by joint classification and segmentation. IEEE Trans. Image Process. **30**, 3113–3126 (2021)
19. Paluru, N., et al.: ANAM-Net: anamorphic depth embedding-based lightweight CNN for segmentation of anomalies in COVID-19 chest CT images. IEEE Trans. Neural Netw. Learn. Syst. **32**(3), 932–946 (2021)
20. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. **40**(4), 834–848 (2017)
21. Xiao, H., Ran, Z., Mabu, S., Li, Y., Li, L.: SauNet++: an automatic segmentation model of COVID-19 lesion from CT slices. Vis. Comput., 1–14 (2022). https://doi.org/10.1007/s00371-022-02414-4

22. Enshaei, N., et al.: COVID-rate: an automated framework for segmentation of COVID-19 lesions from chest CT images. Sci. Rep. **12**(1), 1–18 (2022)
23. Raj, A.N.J., et al.: ADID-UNET-a segmentation model for COVID-19 infection from lung CT scans. PeerJ Comput. Sci. **7**, e349 (2021)
24. Ouyang, X., et al.: Dual-sampling attention network for diagnosis of COVID-19 from community acquired pneumonia. IEEE Trans. Med. Imaging **39**(8), 2595–2605 (2020)
25. Shi, T., Cheng, F., Li, Z., Zheng, C., Xu, Y., Bai, X.: Automatic segmentation of COVID-19 infected regions in chest CT images based on 2D/3D model ensembling. Acta Automatica Sinica **47**(AAS-CN-2021-0400), 1 (2021). https://doi.org/10.16383/j.aas.c210400, https://www.aas.net.cn/cn/article/doi/10.16383/j.aas.c210400
26. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobileNetV2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
27. Ma, N., Zhang, X., Zheng, H.T., Sun, J.: ShuffleNetV2: practical guidelines for efficient CNN architecture design. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 116–131 (2018)
28. Mehta, S., Rastegari, M., Shapiro, L., Hajishirzi, H.: EspNetv2: a light-weight, power efficient, and general purpose convolutional neural network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9190–9200 (2019)
29. Wu, B., et al.: Shift: A Zero FLOP, zero parameter alternative to spatial convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9127–9135 (2018)
30. Jeon, Y., Kim, J.: Constructing fast network through deconstruction of convolution. In: Advances in Neural Information Processing Systems, vol. 31 (2018)
31. Szegedy, C., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
32. Chollet, F.: Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1251–1258 (2017)
33. Howard, A.G., et al.: MobileNets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
34. Qiu, Y., Liu, Y., Li, S., Xu, J.: MiniSeg: an extremely minimum network for efficient COVID-19 segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence vol. 35(6), pp. 4846–4854 (2021). https://doi.org/10.1609/aaai.v35i6.16617
35. Zhou, Y., Zhu, Y., Ye, Q., Qiu, Q., Jiao, J.: Weakly supervised instance segmentation using class peak response. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3791–3800 (2018)
36. Tang, W., et al.: M-SEAM-NAM: multi-instance self-supervised equivalent attention mechanism with neighborhood affinity module for double weakly supervised segmentation of COVID-19. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12907, pp. 262–272. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87234-2_25
37. Liu, X., et al.: Weakly supervised segmentation of COVID-19 infection with scribble annotation on CT images. Pattern Recogn. **122**, 108341 (2022)
38. Khoreva, A., Benenson, R., Hosang, J., Hein, M., Schiele, B.: Simple does it: weakly supervised instance and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 876–885 (2017)

39. Hsu, C.C., Hsu, K.J., Tsai, C.C., Lin, Y.Y., Chuang, Y.Y.: Weakly supervised instance segmentation using the bounding box tightness prior. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
40. Laradji, I., et al.: A weakly supervised consistency-based learning method for COVID-19 segmentation in CT images. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2453–2462 (2021)
41. Laradji, I.H., Saleh, A., Rodriguez, P., Nowrouzezahrai, D., Azghadi, M.R., Vazquez, D.: Weakly supervised underwater fish segmentation using affinity LCFCN. Sci. Rep. **11**(1), 1–10 (2021)
42. Qian, R., Wei, Y., Shi, H., Li, J., Liu, J., Huang, T.: Weakly supervised scene parsing with point-based distance metric learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33(01), pp. 8843–8850 (2019). https://doi.org/10.1609/aaai.v33i01.33018843
43. MedSeg, Håvard, Bjørke, J., Tomas, S.: MedSeg COVID dataset 1 (2021). https://figshare.com/articles/dataset/MedSeg_Covid_Dataset_1/13521488
44. MedSeg, Håvard, Bjørke, J., Tomas, S.: Medseg COVID dataset 2 (2021). https://figshare.com/articles/dataset/Covid_Dataset_2/13521509
45. Ma, J., et al.: COVID-19 CT lung and infection segmentation dataset (2020). https://doi.org/10.5281/zenodo.3757476
46. Morozov, S.P., et al.: MosMedData: chest CT scans with COVID-19 related findings dataset. arXiv preprint arXiv:2005.06465 (2020)
47. Laradji, I.H., Rostamzadeh, N., Pinheiro, P.O., Vazquez, D., Schmidt, M.: Where are the blobs: counting by localization with point supervision. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 547–562 (2018)
48. Cheng, B., Parkhi, O., Kirillov, A.: Pointly-supervised instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2617–2626 (2022)
49. Mettes, P., van Gemert, J.C., Snoek, C.G.M.: Spot on: action localization from pointly-supervised proposals. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9909, pp. 437–453. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46454-1_27
50. Papadopoulos, D.P., Uijlings, J.R., Keller, F., Ferrari, V.: Training object class detectors with click supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6374–6383 (2017)
51. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: What's the point: semantic segmentation with point supervision. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 549–565. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_34
52. Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Dollár, P.: Designing network design spaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10428–10436 (2020)