IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 22, 2022, accepted May 7, 2022, date of publication May 11, 2022, date of current version May 19, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3174199

A Fast and Efficient Data Augmentation for Sematic Segmentation Based on LQE Head and BAS-DP

FAN WANG[1] **AND ZHENYU WANG**[2]
[1]School of Information Management, Wuhan University, Wuhan 430072, China
[2]School of Electrical and Automation Engineering, Nanjing Normal University, Nanjing 210023, China

Corresponding author: Fan Wang (2021101040028@whu.edu.cn)

ABSTRACT Aiming at the problem that it takes too long to manually label numerous semantic segmentation data sets of vehicle images, a fast and effective data augmentation for semantic segmentation is proposed. Firstly, to solve the problem that traditional data augmentation algorithms are difficult to generate vehicle images and corresponding labels at the same time, a vehicle image data augmentation for semantic segmentation based on FCN (Fully Convolutional Network) and GCIoU (Generally Contour Intersection over Union) is proposed, which can simultaneously generate vehicle images and corresponding labels. Then, aiming at the problem that some low-quality data exist in the generated dataset, a data set quality discriminator based on LQE (Label Quality Evaluation) head is proposed. The discriminator can distinguish between low-quality and high-quality label files. Finally, aiming at the problem that the excessive weight of the label file causes the calculation speed to decrease, a lightweight algorithm for the label file based on BAS-DP (Beetle Antennae Search Douglas-Peucker Algorithm) is proposed. The lightweight algorithm can greatly decrease parameters of the label file and improve availability of data-augmented results. Experimental results show that the proposed data augmentation algorithm is better than DCGAN (Deep Convolutional Generative Adversarial Networks), WGAN (Wasserstein Generative Adversarial Networks) and other data augmentation algorithms in accuracy. The AP50 and AP75 of the proposed algorithm reach 0.924 and 0.41, respectively. In addition, the proposed data augmentation algorithm still performs well in scenes with single-object, multi-object and ultra-multi-object. Simultaneously, the proposed data augmentation algorithm has three advantages, which are higher accuracy, faster speed, and less training data required.

INDEX TERMS Semantic segmentation, beetle antennae search algorithm, neural network, data augmentation.

I. INTRODUCTION

With the development of modernization, vehicles are the main means of transportation in cities, and vehicle identification and detection are widely used in intelligent transportation [1]. Pixel-level labeling of vehicle data sets and semantic segmentation of vehicle images are of great significance to the study of various characteristics of vehicles. For example, vehicle object detection, vehicle classification, license plate recognition, vehicle speed measurement and vehicle color recognition, etc [2]–[4].

The current vehicle image related data sets include: KITTI, UA-DETRAC BDD100K data set, etc [5]. However, these data sets are mostly forward vehicles, which are not suitable for vehicle recognition tasks in all actual traffic situations [6]. When carrying out vehicle recognition in some specific scenarios, it is necessary to personally collect vehicle image data sets in the specific environment and label these vehicles. For the task of semantic segmentation of vehicle images, the required data set is huge. Compared with the labeling time required for classification and detection tasks, the time required for labeling semantic segmentation datasets is often in hours. The amount of data required to train a high-precision semantic segmentation deep neural network is about 6000 or more. However, only 100 images of vehicles in a transportation environment require more than 12 hours of labeling time for a person. Manually labeling the data set is time-consuming and labor-intensive, and the quality

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Sharif.

of the labeled vehicle data in the fatigue state is low [7]. This method is difficult to quickly and conveniently obtain high-quality, large-quantity and satisfactory vehicle data sets. Through the data augmentation method, the vehicle image and the corresponding label file can be generated at the same time, which greatly reduces the manual labeling time of the data set [8].

Therefore, this article proposes a novel data augmentation approach to generate numerous vehicle images and corresponding label files. The vehicle image data augmentation for semantic segmentation in this paper is as follows. First, a vehicle image semantic segmentation label generation neural network based on GCIoU and SDE head (Semantic segmentation data augmentation head) is proposed. The neural network can learn how to label unlabeled data sets from a small number of labeled samples. We use this neural network to generate a large amount of unlabeled data that is automatically labeled, and generate a large amount of vehicle semantic segmentation labeled files. Then, in order to obtain high-quality label files, a label quality discriminator based on LQE head is proposed, which can distinguish the generated high-quality and low-quality data sets. Through this discriminator, a high-quality semantic segmentation label data set can be obtained. Finally, using BAS (Beetle Antennae Search) algorithm optimized Douglas–Peucker algorithm proposed a BAS-DP-based label lightweight algorithm, which can reduce the parameters of the label file while ensuring the accuracy of the data [9], [10]. This algorithm can improve the usability of the generated data set.

Therefore, the proposed data augmentation algorithm can quickly generate high-quality and lightweight vehicle image semantic segmentation data sets. At the same time, migration learning is also used in the neural network training process, which reduces the training time and the amount of training data for the data augmentation network.

The main contributions of the paper are summarized below:

- Aiming at the problem that the manual labeling of vehicle semantic segmentation data takes too long, a vehicle semantic segmentation data augmentation network is proposed, which can automatically generate labeled high-quality vehicle semantic segmentation data. First, combining the SDE head and residual network, a data augmentation network is proposed to generate a vehicle semantic segmentation data. Then, in order to solve the problem of uneven quality of the generated vehicle semantic segmentation data, a discriminator based on LQE head is proposed. Finally, the discriminator and vehicle semantic segmentation data augmentation network are merged to form an end-to-end high-quality data augmentation network that can simultaneously generate vehicle images and corresponding high-quality label files. The high-quality data augmentation network has the advantages of high data quality and less training data required.
- Aiming at the problem that the weight of the data augmentation result increases the training time, a lightweight algorithm for label files based on BAS-DP is proposed. On the basis of the above data augmentation network, the lightweight algorithm further reduces the weight of each label file of the data augmentation result. On the premise of maintaining the accuracy of the data augmentation results, the lightweight algorithm can improve the network computing speed and increase the availability of the data augmentation results. Finally, the lightweight algorithm based on BAS-DP is combined with the above data augmentation network to form a novel high-quality vehicle data augmentation for semantic segmentation.

The organization of the paper is follows. Firstly, the related work of the predecessors was introduced in Section II. Then, the general framework of the vehicle image data augmentation for semantic segmentation is established in the section III, and describes the specific implementation process. The section IV is quantitative and qualitative comparative experiments. Through comparative experiments, it is proved that the proposed data augmentation algorithm has the advantages of less training data, high speed and high precision. Section V is the conclusion and expectation of this article.

II. RELATED WORK

More and more researchers are constantly conducting research on vehicle object detection and semantic segmentation algorithms, which makes deep learning-based object detection algorithms widely used [11], [12]. For example, Faster R-CNN, Mask R-CNN and other object detection algorithms have greatly improved intelligent transportation and vehicle detection [13], [14]. However, these algorithms require a large number of labeled datasets to train a precise vehicle detection model [15], [16]. In response to the problem of semantic segmentation methods requiring a large amount of labeled data, researchers have given different solutions. Yebes *et al.* used a lightweight two-stage object detection network to achieve vehicle detection and object classification on urban roads [17]. However, these methods reduce the required training data through simpler network structures, so the disadvantage is that the accuracy is reduced. Hu *et al.* used a semi-supervised method to improve the accuracy of the Mask R-CNN [18]. Wang *et al.* proposed a multi-domain joint learning method using a semi-supervised learning framework, which can reduce the interference of noise while requiring a small amount of training data [19]. Bang proposed a novel image-to-image translation network that reduces the amount of training data required and improves the robustness of the algorithm [20]. Kim *et al.* Use unsupervised lightweight neural network for vehicle detection. However, the accuracy of these methods is not high, and pre-trained models for transfer learning still require a large number of labeled datasets for training [21]. Although, to a certain

extent, the above methods can reduce the amount of training data required, there is a bottleneck in the accuracy of these methods. A more effective way to improve the accuracy of the semantic segmentation network is to directly augment the dataset [22], [23].

Semantic segmentation data augmentation methods are mainly divided into two categories. One is traditional data augmentation methods. Data augmentation is carried out by flipping and transforming a small number of manually labeled samples, random trimming, random pasting, color dithering, translation transformation, scale transformation, contrast transformation, noise disturbance and reflection transformation [24]. When traditional data augmentation algorithms perform color dithering and noise disturbance, they will generate samples with a large gap from the real samples, which will cause negative optimization of the network. In addition, the method of randomly cutting and pasting the vehicle object may cause the loss of the associated information between the object and the environment. For example, randomly cutting and pasting vehicle objects into sky areas, roofs, indoors, etc., will destroy the associated information between the vehicle and the road, and will more likely cause the network to misjudge the vehicle objects.

Another type of method is based on a generative adversarial network, which generates more types of vehicle image samples from a small number of samples [25]–[27]. Therefore, the generated samples are easily constrained by the original samples. The images and labels generated by the generated confrontation network all come from a small number of labeled original samples. Numerous highly similar samples make the network prone to overfitting. Moreover, the poor quality of the labels generated by this method leads to the need for manual secondary screening. Dumagpi *et al.* used DGAN for data augmentation, which increases the data volume of X-ray images and improves the detection accuracy of X-ray images in Fast R-CNN. Ke *et al.* annotated numerous images through WGAN, which increased the number of images and improved the performance of deep convolutional neural networks. Li *et al.* used DCGAN to augment the dataset of spectral images, so that the deep learning model can be better trained, thereby improving the accuracy of the model [28]. Fang *et al.* used DCGAN to generate numerous unlabeled samples and trained image recognition model. This method can enhance the classification model and effectively improve the accuracy of image recognition [29]. Zhou *et al.* used LP-WGAN to generate competitive images earlier and got higher evaluation scores [30].

In recent years, there are also some articles related to generative adversarial networks. Zhu *et al.* used GAN generate numerous photo-realistic SAR images. The approach can ally ''Clever Hans'' phenomenon greatly caused by the spurious relationship between generated SAR images and the corresponding classes [31]. Zhang *et al.* augmented the dataset of images by CF-GAN, which has higher accuracy than other data augmentation approaches [32]. Wei *et al.* used GAN to augment cancer images, increasing cancer classification accuracy to 92.6% [33]. These data augmentation algorithms based on generative adversarial networks are only suitable for data augmentation of similar samples, perform poorly in complex samples, and difficult to complete the labeling task of semantic segmentation data.

Both of data augmentation algorithms have some shortcomings, so that neither of them is suitable for data augmentation of vehicle image semantic segmentation datasets. Therefore, the article proposes a high-quality data augmentation algorithm for semantic segmentation data. And, we compare the proposed algorithm with other data augmentation algorithms and verify the performance advantages of the proposed algorithm.

III. OVERALL FRAMEWORK OF VEHICLE IMAGE DATA AUGMENTATION FOR SEMANTIC SEGMENTATION

The information recorded in the label file mainly has two types of information, one is the key points on the outline surrounding the object, and the other is the object category. If you want to obtain high-quality label files, the accuracy of the labeling of these two kinds of information must be very high. Fig.1 is a vehicle image and corresponding tag information.

The semantic segmentation label data augmentation algorithm of vehicle image is shown in Fig.2. The specific steps of the vehicle image semantic segmentation data augmentation method are mainly divided into two parts: (1) The first part is the vehicle image semantic segmentation data augmentation network training. First, build a semantic segmentation label generation network and a label file quality judgment network, and build a vehicle semantic segmentation generation network. Then, a very small amount of labeled data is selected to train the data augmentation network. When training the network, the pre-training model of the coco data set is used for migration learning, reducing training time and preventing network overfitting. Finally, select the appropriate vehicle image feature extraction network and the best hyperparameters to make the accuracy of the data augmentation model reach the best. (2) The second part is the data augmentation of the vehicle semantic segmentation label file. First, preprocessing such as scale normalization is performed on a large number of unlabeled vehicle images. Then, the semantic segmentation label generation model in the previous part of the data augmentation model is used to initially generate a large number of semantic segmentation label files. At the same time, the discriminant model in the data augmentation network is used to filter out high-quality label files and discard low-quality label files. Finally, use the BAS-DP-based label file lightweight algorithm to reduce the size of the label file while maintaining the accuracy of the label file, and enhance the lightness of the label file. Finally, the vehicle semantic segmentation tags and unsupervised data are combined to form a large number of high-quality vehicle image semantic segmentation data sets.

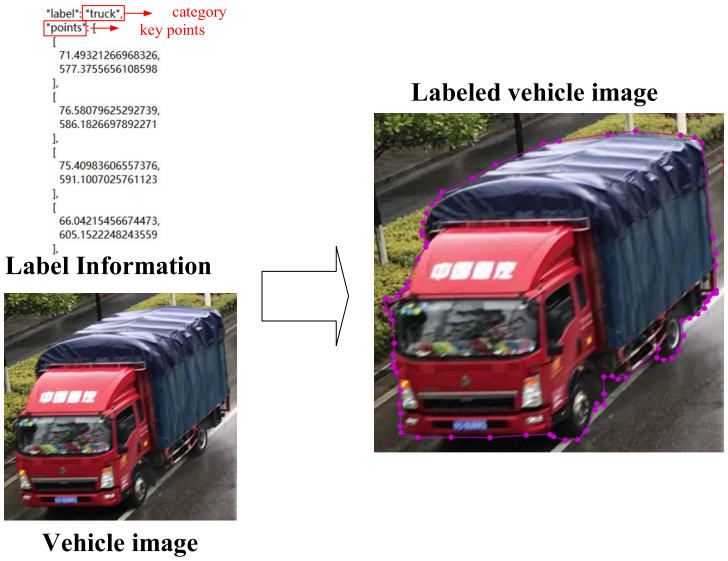


FIGURE 1. Vehicle image and corresponding label file.

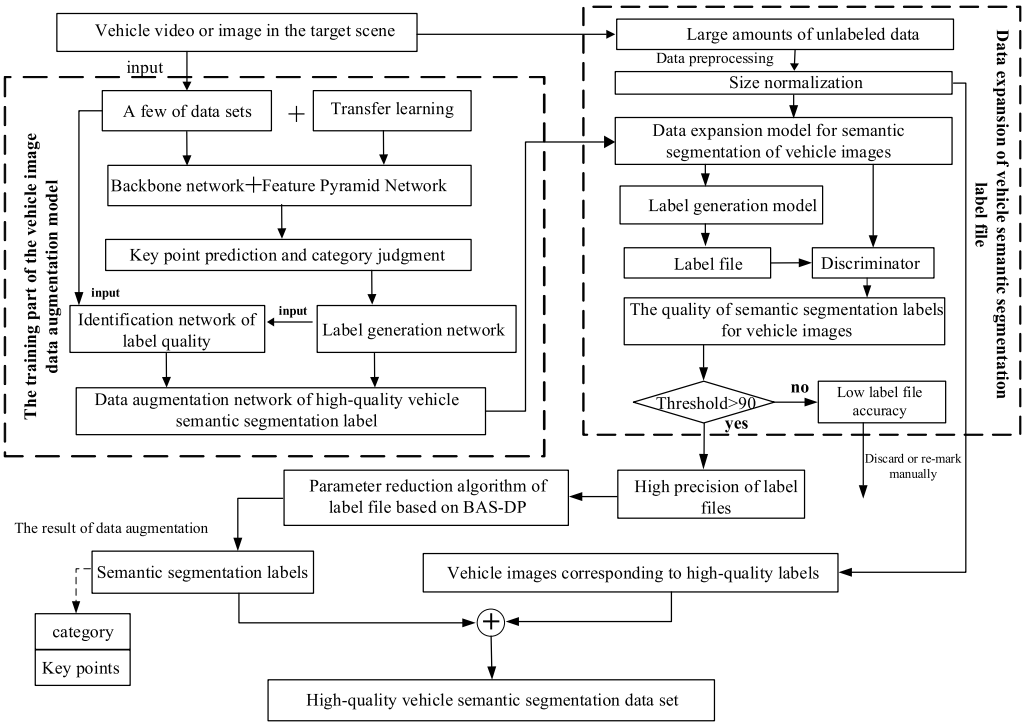


FIGURE 2. Data augmentation algorithm for vehicle semantic segmentation label.

IV. VEHICLE SEMANTIC SEGMENTATION LABEL FILE DATA AUGMENTATION NETWORK

The semantic segmentation data augmentation network of vehicle images is shown in Fig.3, which mainly includes three parts. The first part is the feature extraction stage of the vehicle image. Used to extract semantic features in vehicle images. The second part is the semantic segmentation label file generation part, which is used to generate a large number of semantic segmentation label files. The third part is a discriminator, which is used to judge the quality of the tags generated in the second part, and select high-quality vehicle image semantic segmentation tag files as the result of data augmentation. The following describes each part of the network in detail.

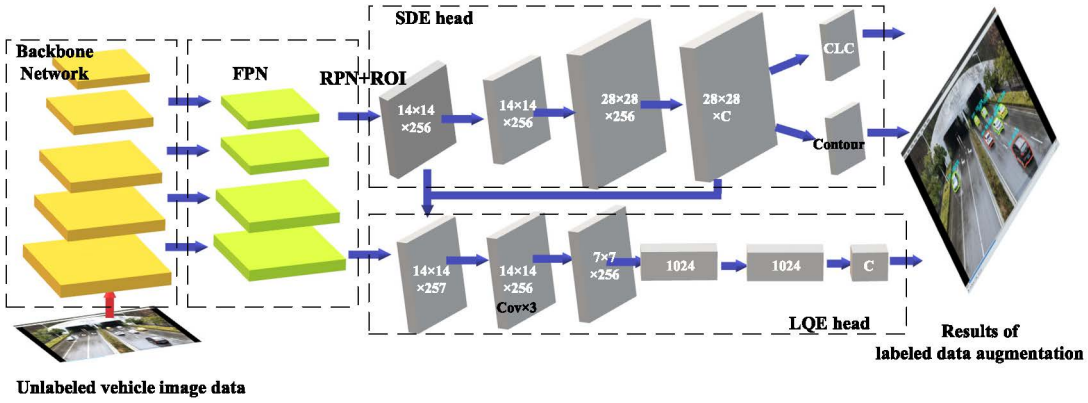


FIGURE 3. Vehicle image semantic segmentation data augmentation network.

A. FEATURE EXTRACTION BASED ON TRANSFER LEARNING AND DEEP RESIDUAL NETWORK

Features are extracted from unlabeled vehicle images through the backbone network, and multi-scale feature layers are formed through FPN (Feature Pyramid Network) to enhance the network's ability to recognize small objects.

Among them, the backbone network can be a feature extraction network composed of arbitrary convolutional layers, or a commonly used deep convolutional neural network (Such as ResNet 50, ResNet101, VGG19, etc.). Through the convolution operation, the size of feature map is transformed from $1980 \times 1080 \times 3$ into the $32 \times 32 \times 2048$, which is used as the input of the feature pyramid network. In order to get the best feature extraction, corresponding comparative experiments are carried out on three high-precision backbone networks: ResNet101, ResNet50, and MobilNetV1 in the experiment and analysis part.

Transfer learning is to assign initial values to the weights in the network, which can effectively prevent overfitting, reduce training data, and reduce training time. At the same time, because most of the weights in the network are in the feature extraction part. In order to improve the effectiveness of extracting features and the training speed of the network. First, train the backbone network part in the public data set to obtain a pre-trained model. Then, transfer learning is used to provide initial values for the feature extraction part of the network to reduce the training time of the network and prevent overfitting.

The images in the coco dataset are taken on urban roads and contain many non-vehicle image categories, which can improve the accuracy of the background and foreground classification of vehicle images by the network. Therefore, the coco data set is selected to train the pre-training model, and the ablation experiment is carried out in the experimental part to prove the effectiveness of transfer learning.

This element pyramid network has five layers. From the first layer to the fifth layer, the scale of the feature map will gradually decrease. In this way, feature maps of different scales are generated. Then, adjacent feature maps are merged with each other to obtain a new feature map. The new feature map not only contains the details in the low-level feature map, but also contains the large receptive field of the high-level feature map, thereby improving the feature extraction ability of the small object network. We choose this new feature map as the subsequent network input.

B. SEMANTIC SEGMENTATION LABEL DATA AUGMENTATION BASED ON FCN AND GCIOU

Firstly, the positive and negative samples in the new feature map are distinguished through the classification area of the interested area, where the area belonging to the object is the positive sample, and the area belonging to the background is the negative sample.

Then, FCN is used to deconvolve the fourth-layer feature map to obtain the mask of the positive samples in the original image input by the network. Connect the points around the mask to obtain the contour information of each vehicle object, that is, the points on the object contour. This is also one of two types of important information in the label file. At the same time, the fully connected layer is used to classify each different object area category in the feature map, and the category information in the label file is obtained. Finally, the object contour information and category information are integrated into the label file to generate the vehicle semantic segmentation label file.

The loss function is also vital to the training results of the data augmentation network. The contour information in the semantic segmentation label is to record the curve surrounding the object. Since the contour of the object is surrounded by an irregular shape, the measurement of the accuracy of the object contour is often converted into the measurement of the accuracy of the irregular area surrounding the object in deep learning. The traditional method of measuring the accuracy of irregular images is MIoU, which imitates the IoU method to judge the difference between the predicted area and the real area by the value of the intersection ratio. The MIoU is shown

in formula 1.

$$\begin{cases} M_{\mathrm{IoU}} = \dfrac{|A \cap B|}{|A \cup B|} \\ L_{\mathrm{MIoU}} = 1 - M_{\mathrm{IoU}} \end{cases} \tag{1}$$

Among them, A is the real irregular object area. B is the irregular object area predicted by the neural network. M_{IoU} is the value of MIoU. L_{MIoU} is the loss function that measures the coincidence degree of A and B.

There are two problems with MIoU as a metric function and loss function for irregular object areas. First, if A and B do not overlap, M_{IoU} will be 0 and will not reflect the distance between the two objects. In this case of non-overlapping objects, if L_{MIoU} is used as a loss function, there will be a situation where the gradient is 0 and cannot be optimized. Second, M_{IoU} cannot distinguish between the alignment of the predicted bounding box and the true bounding box, and the M_{IoU} of overlapping objects with the same intersection level in different directions will be completely equal. As shown in Fig.4, the edge quality of the left image and the right image are obviously different, but their corresponding M_{IoU} values are the same, which will affect the further optimization of the network.

FIGURE 4. MIoU values under different overlapping conditions.

GCIoU is a new method for evaluating the quality of the contour generation part of the label file. L_{GCIoU} is the value of its loss function, and its calculation formula is shown in formula 2.

$$\begin{cases} G_{\mathrm{CIoU}} = M_{\mathrm{IoU}} - \dfrac{|C_{\mathrm{AB}} - (A \cup B)|}{|C_{\mathrm{AB}}|} \\ L_{\mathrm{GCIoU}} = 1 - G_{\mathrm{CIoU}} \end{cases} \tag{2}$$

where, A is the real edge encircled area of the object, B is the encircled area of the object edge predicted by the neural network, and C_{AB} is the smallest matrix area enclosing the two bounding boxes of $_A$ and $_B$. G_{CIoU} includes calculating the minimum closed area of A and B and the minimum circumscribed matrix area. Even if A and B do not intersect, G_{CIoU} is still not zero. At the same time, it can accurately reflect the degree of overlap between the predicted box B and the real bounding box A.

Fig.5 shows the calculation method of GCIoU. Fig.5 shows the corresponding GCIoU values under different overlapping conditions. It can be seen from Fig.5 that GCIoU can distinguish overlapping objects with the same intersection level, which is conducive to the optimization of the semantic segmentation network.

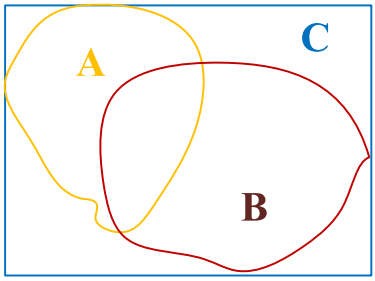


FIGURE 5. GCIoU calculation schematic diagram.

C. DATA SET QUALITY DISCRIMINATOR BASED ON LQE HEAD

It is not enough to use the second part to simply generate label files for data augmentation. Any data generation network will produce some poor quality data. Failure to judge and process the generated data will make the results of data augmentation mixed into a large amount of low-quality data. Therefore, a discriminator is proposed next to distinguish the quality of the generated data and obtain a high-quality labeled data set.

Therefore, a discriminator is designed, which is used to evaluate the quality of the label file. The content of the evaluation mainly includes the accuracy of the contour of the object and the accuracy of the object classification. Then, by setting the quality threshold, the labels whose quality is lower than the threshold are discarded, and the labels whose quality is higher than the threshold are retained. Finally, the tags above the threshold and the corresponding vehicle image data are combined to form a vehicle semantic segmentation data set, which is also the result of data augmentation of the network.

The first is to evaluate the accuracy of the object contour in the label file. Since the contour of the object is surrounded by an irregular shape, using the regression principle of the convolutional neural network, a LQE head is designed to regress the accuracy of the object contour in the generated data. The convolutional neural network can not only extract the features in the image, but also can be used to regress the similarity of the two images, using the LQE head to regress the true contour (Truth contour) and the predicted contour (predict contour), and calculate each The GCIoU value of the difference between the real contour of the object and the predicted contour. Then, normalize GCIoU to get S_{IoU}. S_{IoU} is the evaluation quality of the contour, and its range is between 0 and 1. By setting different S_{IoU} thresholds, object contours of different quality can be obtained. The closer the S_{IoU} value is to 1, the better the object contour prediction effect.

1) THE STRUCTURE OF THE DISCRIMINATOR BASED ON LQE HEAD

It consists of 4 convolutional layers and 3 fully connected layers. For 4 convolutional layers, the kernel size and the

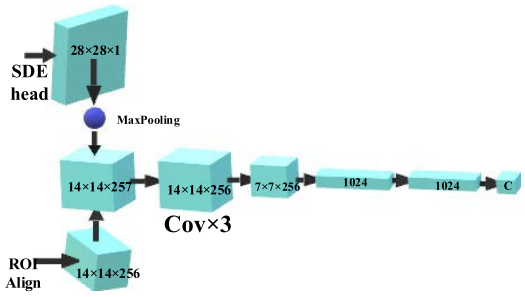


FIGURE 6. LQE head structure.

number of filters of all convolutional layers are set to 3 and 256, respectively. For 3 fully connected layers, set the output of the first two fully connected layers to 1024 to connect all neurons, and the C of the last fully connected layer is the number of categories to be classified. Finally, the LQE head outputs the contour quality S_{IoU} of each object.

2) INPUT STRUCTURE OF LQE HEAD

Take Truth-contour and predict-contour together as the input of LQE head. Among them, Truth-contour has a feature map, and predict-contour is the contour output by SDE head. Since the output size of the SDE head is different from the size of the feature map, two input structures are designed. Fig.7 shows the two input structures of LQE head.

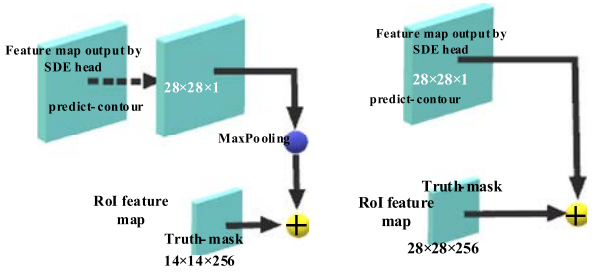


FIGURE 7. Input structure of LQE head.

Where, in the left picture of Fig.7, the input structure of the designed LQE head is that the feature layer output by the SDE head is subjected to maximum pooling through a convolution kernel with a size of 2 and a stride of 2. Then, the pooling result is concatenated by a feature map to get the input of the LQE head, and the feature map size is $14 \times 14 \times 256$. In the right picture of Fig.7 is another LQE head input structure. The feature layer output by the SDE head is directly concatenated to the feature map with a larger size to obtain the input of the LEQ head. This structure does not need to go through max pooling. Both structures can be used as input to the LQE header.

Since the label file includes two kinds of information, the object contour and the object category, the discriminator is required to evaluate both types of information well: (1) The contour accuracy of the object in the label file needs to be evaluated. (2) It is necessary to evaluate the classification accuracy of the object in the label. First, LQE head is used to obtain the profile accuracy evaluation index S_{IoU}. Then, the confidence of the object classification result is normalized between 0 and 1. Take it as the category accuracy evaluation index of the generated data, which is Scls. In order to use one objective function to express the two tasks, the two indicators are multiplied, as shown in formula 3, which is the final output result of the discriminator for each object.

$$S_{\mathrm{DE}} = S_{\mathrm{cls}} \times S_{\mathrm{IoU}} \tag{3}$$

Among them, S_{DE} is the accuracy of the discriminator for each object in the tag file. S_{cls} represents the category accuracy index in the label file. S_{IoU} represents the contour accuracy index in the label file. The set S_{DE} threshold is 0.9. When the S_{DE} of each object in a label file is higher than 0.9, the quality of the generated label is higher. When the S_{DE} of the label file is lower than 0.9, the quality of the generated label is lower.

D. LOSS FUNCTION OF DATA AUGMENTATION NETWORK

The vehicle image semantic segmentation data augmentation network is mainly composed of feature extraction part, SDE head, LQE head and other parts. Therefore, the loss function formula of this network is shown in formula 4.

$$L = L_{\mathrm{class}} + L_{\mathrm{p}} + L_{\mathrm{r}} + L_{\mathrm{LQE}} \tag{4}$$

where, L_{class} is the category loss in the generated label, L_{p} is the loss of the feature extraction network, and L_{r} is the weight regularization loss. L_{LQE} is the LQE head loss function.

E. A LIGHTWEIGHT ALGORITHM FOR LABEL FILES BASED ON BAS-DP

Through the vehicle semantic segmentation data augmentation network, a large number of high-precision vehicle image semantic segmentation data sets can be obtained. Since the label file mainly records the category information of the object and the outline information of the object. If you save all the points on the object edge into the label file, it will cause too many parameters in the label file. Too many label parameters will cause the neural network to take a long time when reading the label file, which is not conducive to the use of the label file. Therefore, a lightweight algorithm for label files based on BAS-DP is proposed. The algorithm converts the contour curve surrounding the object into the best approximation positive polygon. The number of coordinate points contained in the polygon is small, and the number of parameters in the label can be reduced while the accuracy of the generated label is guaranteed, so that the label file is lighter.

Fig.8 shows the result of reducing the number of key points on the object edge. Using the best approximation positive polygon to replace the curve is the most direct and effective method. Therefore, it is necessary to use a polygonal approximation algorithm to convert the contour curve of the object into a polygon, and then re-record the key point coordinates on the polygon in the label file.

FIGURE 8. Optimization of the number of key points on the edge of the object.

The Douglas-Peucker algorithm is a classic polygon approximation algorithm that can approximate a closed curve as a polygon and reduce the number of points as much as possible. It has the advantages of translation and rotation invariance. The calculation steps of the Douglas-Puck algorithm are as follows:

Step1: Calculate the two points M and N with the furthest distance on the closed curve, and connect the two points M and N to form a line segment DMN.

Step2: Find the point q with the largest distance from the DMN line segment among the remaining points of the closed curve, and calculate the distance Dq between q and the DMN.

Step3: Compare the distance Dq with the predetermined threshold Dthreshold. If Dq is less than Dthreshold, then use the line segment DMN as a straight line similar to the curve.

Step4: If the distance Dq is greater than Dthreshold, use point q to divide the curve into two straight lines, Mq and Nq (points M, N, and q are called key points), and perform steps 1 to 3 for Mq and Nq respectively.

Step5: When all the curves have been processed, connect the key points to form a polygon, which is the approximation of the original closed curve.

When the Douglas–Peucker algorithm calculates Dq, it needs to solve all the key points on the curve one by one, which requires a lot of calculation time. The bionic algorithm can reduce the calculation time by optimizing the Douglas–Peucker algorithm.

The BAS algorithm is a bionic algorithm that realizes efficient optimization by simulating the beetle foraging. The BAS algorithm does not need to know the specific form of the function, and does not need gradient information to achieve optimization. Compared with other population algorithms, BAS algorithm only needs one individual to realize the optimization. This reduces the calculation time of the algorithm. The specific steps of the BAS algorithm are as follows.

Step1: Initialization of parameters in the BAS algorithm. Initialize the attenuation factor Eta, the step size $Step$, the ratio of the step size and the whisker c, the number of iterations n, and the number of parameters k.

Step2: According to formula 5, randomize the direction d_{ir} of longhorn beetle and the distance d_0 between the two antennas of beetle.

Step3: According to formula 5, calculate the function values fl and fr corresponding to the position of the left antennae xl and the position of the right antennae xr of the beetle, and calculate the value of the position x of the beetle in the next step.

Step4: Repeat steps 2 and 3 for n times. Finally, the optimized function value corresponding to the last position x of longhorn beetle is obtained as the optimal solution.

$$\begin{cases} d_{ir} = rand(k, 1); \quad d_0 = step/c \\ x_{\mathrm{l}} = x + d_0{}^{*}dir/2; \quad x_{\mathrm{r}} = x - step^{*}d_{ir}/2 \\ f_{\mathrm{l}} = f(x_{\mathrm{l}}); \quad f_{\mathrm{r}} = f(x_{\mathrm{r}}) \\ x = x - s_{tep}{}^{*}d_{ir}{}^{*}s_{ign}(f_l - f_r) \end{cases} \tag{5}$$

Then, we use the BAS algorithm to optimize the Douglas–Peucker algorithm, and propose a lightweight algorithm for label files based on the BAS-DP algorithm. This lightweight algorithm can reduce the weight of data augmentation results by reducing the number of parameters in the label file. The calculation process of the label file lightweight algorithm is shown in Fig.9.

This algorithm can greatly reduce the parameter amount of the label file, improve the usability and lightness of the label file, and maintain the accuracy of the label file.

Finally, the BAS-DP algorithm is combined with the data augmentation network proposed in the previous section to form a vehicle image data augmentation algorithm for semantic segmentation, which can generate high-quality vehicle image semantic segmentation datasets and corresponding high-quality label files. At the same time, the proposed algorithm has the advantages of requiring less training data, fast running speed and high accuracy.

V. VERIFICATION AND ANALYSIS

In order to verify that the proposed data augmentation algorithm has the advantages of less training data, fast running speed and accuracy, this paper conducts relevant experiments in this chapter.

Part A is the data acquisition and data preprocessing of the training data. Part B is an ablation experiment with hyperparameter selection. Firstly, the optimal hyperparameters and backbone network of the proposed data augmentation algorithm are selected. Then, the comparative experiments of Test 9 and Test 10 verify that the proposed data augmentation algorithm has the advantage of requiring less training data. Part C is the qualitative analysis experiment. Comparing the results generated by data augmentation with the results of manual labeling, it is qualitatively verified that the proposed data augmentation method has the advantages of high accuracy and fast speed. Part D is the speed comparison experiment. In a quantitative way, to verify that the proposed data augmentation algorithm has a faster speed. Part E is the precision comparison experiment. The proposed data augmentation algorithm is compared with DCGAN and WCGAN, and it is quantitatively verified that the proposed data augmentation algorithm has the advantage of high precision.

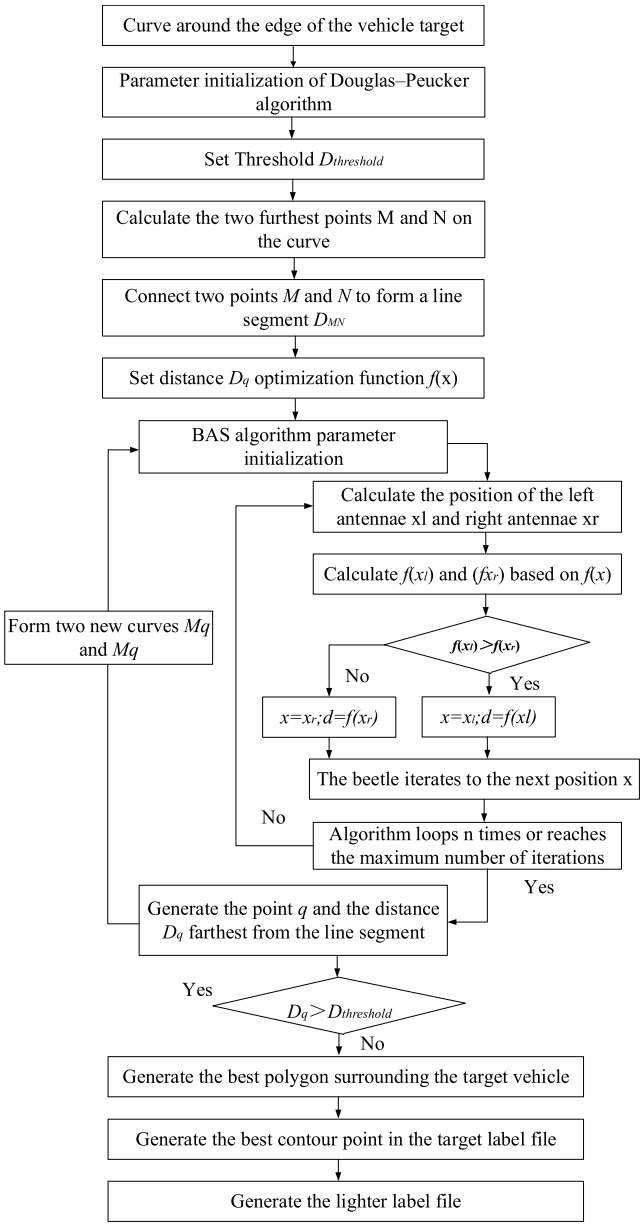


FIGURE 9. A lightweight algorithm for label files based on BAS-DP.

A. COLLECTION AND PREPROCESSING OF TRAINING DATA

Before training the vehicle data augmentation algorithm for semantic segmentation, the training data should be collected and preprocessed.

First, a segment of vehicle video is collected, and two frames of vehicle images are extracted every second as a training dataset. The collected training dataset is a semantic segmentation dataset of vehicle pictures, in which the amount of data is 1200. The categories of the dataset are Bus, Car, MicroBus, SUV (Sports Utility Vehicle), Truck and Sportscar.

Then, since the semantic segmentation dataset not only has vehicle pictures, but also has corresponding label files, which mark the categories and contours of vehicles. So, the vehicle images were manually labeled using software called 'Labelme'. The dataset of vehicle data augmentation algorithm for semantic segmentation is shown in Fig.10. The left side of Fig.10 shows the labeling results for each categories of vehicle. The right side of Fig.10 is the labeling result of each image.

B. ABLATION EXPERIMENTS FOR BEST HYPERPARAMETERS AND BEST BACKBONE NETWORK SELECTION

In part B, 13 groups of ablation experiments were performed, which completed the following3 tasks:

- Through 10 groups of ablation experiments, the optimal hyperparameters of the proposed algorithm are obtained, and it is verified that the proposed algorithm has the advantage of requiring less training data.
- It is verified that transfer learning has a positive impact on the proposed algorithm.
- Through three groups of ablation experiments, the optimal backbone network of the proposed algorithm is obtained.

First, ResNet50 is selected as the backbone network, and 10 groups of ablation experiments are performed. The parameters and results of the 10 groups of experiments are shown in table1. Among them, *mAP* and *mIoU* are used to evaluate the performance of the proposed algorithm. *mAP* is short for mean average precision, and *mIoU* is short for mean Intersection over Union. *mAP* is the average of the accuracy precision for each category and is often used to evaluate the performance of an algorithm. To more rigorously evaluate the performance of the proposed algorithm, the thresholds of *mAP* are set to 0.5 and 0.7, respectively. Those greater than or equal to the threshold are true positive, while those less than the threshold are false positive. In Table 1, the *mAP* and *mIoU* of each group of experiments are shown below the last three rows, which are used to analyze the experimental results and algorithm performance in detail.

The 10 groups of ablation experiments are shown in table 1. O_{Train} and O_{Val} correspond to the number of vehicle objects in the training dataset and validation dataset, respectively. I_{Train} and I_{Val} correspond to the number of images in the training dataset and the validation dataset, respectively. E_{pochs} corresponds to the number of iterations during training, and C_{mini} corresponds the minimum area to wrap the object. I_{size} corresponds to the size of the input image, and R_{Scales} corresponds to the scale of the anchor point. *Pretraining model* indicates whether to add a pre-training model, which is obtained by training the coco dataset. In the field of image processing, the coco dataset is a commonly used public dataset. The following will analyze the results of 10 groups of ablation experiments

Test1 and Test2 use different E_{pochs}, but other parameters are the same. E_{pochs} of Test1 and Test2 are 100 and 200 respectively. With the increase of E_{pochs}, the value of

FIGURE 10. Vehicle dataset of data augmentation for semantic segmentation.

TABLE 1. Ablation experiment of vehicle data augmentation for semantic segmentation.

	Test1	Test2	Test3	Test4	Test5
O_{Train}	2081	2081	2520	3005	3005
O_{Val}	537	537	632	826	826
I_{Train}	680	680	820	1014	1014
I_{Val}	120	120	140	180	180
E_{pochs}	100	200	100	400	400
C_{mini}	56*56	56*56	56*56	56*56	56*56
I_{size}	1024*800	1024*800	1024*800	1024*800	1920*1080
R_{Scales}	(32, 64, 128, 256)	(32, 64, 128, 256)	(32, 64, 128, 256)	(32, 64, 128, 256)	(32, 64, 128, 256)
Pretraining model	NO	NO	NO	NO	NO
mIoU	0. 485	0. 492	0. 535	0. 498	0. 294
mAP (IoU$>$0. 5)	0. 569	0. 586	0. 495	0. 565	0. 395
mAP (IoU$>$0. 7)	0. 472	0. 488	0. 406	0. 485	0. 289
	Test6	Test7	Test8	Test9	Test10
O_{Train}	3005	3005	3005	1573	1573
O_{Val}	826	826	826	537	537
I_{Train}	1014	1014	1014	480	480
I_{Val}	180	180	180	120	120
E_{pochs}	100	100	100	100	100
C_{mini}	28*28	28*28	28*28	28*28	28*28
I_{size}	1024*800	1920*1080	1920*1080	1920*1080	1920*1080
R_{Scales}	(32, 64, 128, 256)	(16, 32, 64, 128)	(8, 16, 32, 64)	(8, 16, 32, 64)	(8, 16, 32, 64)
Pretraining model	NO	NO	NO	NO	Yes
mIoU	0. 545	0. 565	0. 652	0. 429	0. 684
mAP (IoU$>$0. 5)	0. 558	0. 573	0. 716	0. 345	0. 725
mAP (IoU$>$0. 7)	0. 489	0. 493	0. 575	0. 294	0. 585

mAP (IoU $>$ 0. 5) increased from 0. 569 to 0. 586. The number of E_{pochs} is doubled, and the *mAP* is only increased by 0.017, which has great stability. Therefore, when the number of E_{pochs} is small, the proposed algorithm is still easy to converge, which explains the great convergence of the proposed algorithm.

In Test3, more amount of data is increased compared to Test1, and other parameters remain unchanged. The results show that both *mAP* (IoU $>$ 0.5) and *mAP* (IoU $>$ 0.7) are reduced. Then, by increasing the number of Epochs in Test4, the value of *mAP* is improved, and its *mAP* (IoU $>$ 0.5) is 0.565.

In Test5, compared to Test4, in order to evaluate the influence of image width and height, the size of the training image is increased from 1024*800 to 1920*1080. The learning rate is from the default value of 0.001 to 0.02, and the rest of the parameters are like Test4. The results showed that *mAP* (IoU $>$ 0.5) was low. This explains that in the parameters of test5, the proposed algorithm is less adaptable to high-resolution images.

In Test6, compared to Test3, we still use images with a resolution of 1024*800 and reduce the size of C_{mini} from 56×56 to 28×28. The results show that the performance of the proposed algorithm is improved.

Therefore, in Test7, we reduced the value of R_{Scales} compared to Test5, and increased the C_{mini} to 28×28, while keeping the resolution of the image at 1920*1080. It is found that the performance of the proposed algorithm is greatly improved. This explains that reducing C_{mini} can improve the performance of the proposed algorithm on high-resolution images.

In Test8, compared with Test 7, we further reduce R_{Scales}, and other parameters remain. The results show that the performance of the proposed algorithm is greatly improved, and the best R_{Scales} are (8, 16, 32, 64).

In Test9, to explore the effect of reducing the number of training data on the performance of the algorithm, we only reduced the amount of training data compared to test8. The results show that the performance of the proposed algorithm degrades significantly. Therefore, we followed up with Test10 to improve the performance of the algorithm.

In Test10, compared with Test9, the amount of training data is further reduced, and other parameters remain unchanged. However, we used a pretrained model for transfer learning, which was previously trained on the COCO dataset. The results show that both the *mAP* (IoU $>$ 0.5) and *mAP* (IoU $>$ 0.7) values of Test10 are higher than those of Test8, indicating that Test10 has achieved higher precision. However, the amount of training data for Test10 is only half that of Test8. It is verified that the proposed algorithm has the advantage of requiring less training data.

Summarizing 10 groups of ablation experiments, it is found that the hyperparameters in Test10 are the best hyperparameters for the proposed algorithm. 100 Epochs is enough to achieve the convergence of the proposed algorithm. Further, transfer learning can improve the accuracy while greatly reducing the amount of training data required by the proposed algorithm. Therefore, through 10 groups of ablation experiments, the optimal hyperparameters of the proposed algorithm are obtained, and it is verified that the proposed algorithm has the advantage of requiring less training data.

Then, the backbone network is to better extract features from pictures, which has a greater impact on the performance of the proposed algorithm.Therefore, the backbone network selection experiments are next performed, and the parameters used are those in Test10. ResNet50, ResNet101 and MobileNet V1 are all high-precision convolutional neural network structures, which are composed of residual blocks and have better feature extraction capabilities. Through residual learning, they can simplify the architecture, reduce the computational cost, and solve the problem of gradient disappearance well. In order to choose the best backbone network and maintain a balance between speed and accuracy, these three backbone network networks were tested. Further, the performance is compared from 4 aspects: training time, data augmentation speed per image, model weight and accuracy.

TABLE 2. Performance comparison of three backbone networks.

Backbone Network	Train Time/h	Speed Speed/FPS	Model Weight /MB	Accuracy S_{ED} >90
ResNet50	**12. 65**	**6.25**	**186. 75**	**93. 4%**
ResNet101	20. 73	4. 6	268. 86	93. 8%
MobileNet V1	14. 61	5. 2	207. 82	84. 5%

As shown in table 2, In terms of training time, the shorter the time, the better the algorithm performance. Using ResNet50 as the backbone network, the training time is the shortest, which is 12.65 hours. In terms of data augmentation speed, the faster the speed, the better the algorithm performance. The speeds of these three backbone networks are 6.25 FPS, 4.6 FPS, and 5.2 FPS, respectively, and ResNet50 is the fastest. In terms of model size, the lighter the size, the better the network performance. The model size distributions for the three backbone networks are 186.75MB, 268.86MB, and 207.82MB. Using ResNet50 as the backbone network, the weight of the algorithm is the lightest. In terms of accuracy, using ResNe50, ResNet101 and MobileNet V1 as the backbone network, the accuracies are 93.4%, 93.8% and 84.5%, respectively. ResNet101 has the highest accuracy, and the accuracy of ResNet50 is close to that of ResNet101.

Summary of Backbone Network Comparative Experiments. In terms of training time, data augmentation speed per image, and model weights, the performance of ResNet50 is the best. In terms of accuracy, although ResNet50 is not the best, it is only 0.4% lower than the highest accuracy ResNet101, which is very close to ResNet101. Therefore, through comprehensive comparison, ResNet50 is selected as the best backbone network

Summarizing 13 sets of experiments, and the parameters in Test10 are determined as the best parameters of the proposed algorithm, and ResNet50 is determined as the best backbone network. At the same time, it is verified that the proposed algorithm has the advantage of requiring less training data. Next, we conduct qualitative and quantitative comparative experiments using the obtained optimal parameters and the optimal backbone network.

C. DATA AUGMENTATION RESULTS QUALITATIVE ANALYSIS AND COMPARATIVE EXPERIMENTS

There are two main types of evaluation methods for data augmentation. One approach is qualitative evaluation, which is judged by comparing the quality of images and labels in data augmentation results. Another method is quantitative evaluation, which evaluates the results of data augmentation through fixed parameters.

In Part C, we conduct qualitative analysis and comparative experiments. In Part D, we perform quantitative analysis and comparative experiments.

Usually vehicle semantic segmentation datasets are manually labeled. The data augmentation algorithm proposed in this paper is to replace manual labor, so the results of the proposed algorithm must be close to the results of manual labeling in accuracy. Firstly, the results of the proposed algorithm are qualitatively compared with those of the manual method.

In order to visually compare the difference between the results of the proposed method and the results of artificial methods, we qualitatively compare the two results in ''labelme''. Then, we use 'labelme' to open the output of the proposed data augmentation algorithm. Further, manually label the same vehicle image using a manual method, then also open with 'labelme'. The first is a single-object qualitative comparison experiment.

As shown in Fig.11, Fig.11(a) is the result of manual labeling, and Fig.11(b) is the result generated by the proposed data augmentation algorithm. As shown in the figure, the result of data augmentation is close to the result of manual labeling.

(a) Manual method (b) Proposed method

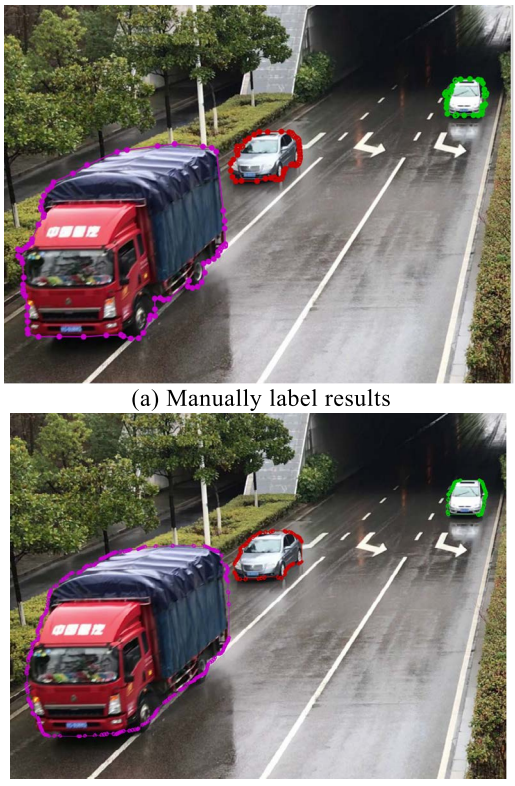
FIGURE 11. Comparative experiment of a single-object.

Further, the polygon around the object is to distinguish the object from the background as much as possible at the pixel level. The object is inside the closed area, and the background is outside the closed area. Therefore, the more accurate keypoints on the polygon, the higher the quality of the semantic segmentation dataset.

As shown in Fig.11, there are few keypoints marked in Fig.11(a). The number of keypoints marked in Fig.11(b) is more, and the keypoints are all in accurate positions. Therefore, the results generated by the proposed method can better distinguish the object from the background, and its quality is higher than that of the manual method in the single-object comparison experiment.

In order to further discover the data augmentation effect of the proposed algorithm in multi-object, we carried out multi-object qualitative comparison experiments in three scenarios: single-object, multi-object and ultra-multi-object.

Fig.12 is the data augmentation comparison experiment for multiple objects, Fig.12(a) is the result of manual labeling, and Fig.12(b) is the result of the proposed method. The labeling effect of Fig.12(b) and Fig.12(a) is close. Further, we conduct speed comparison experiments. For Fig.12, manual labeling takes 42 seconds to label, while the proposed algorithm only takes 1.2 seconds. Finally, it is proved that the proposed data augmentation algorithm has good performance in multi-object scenarios, and the proposed method is faster than manual labeling.



(a) Manually label results

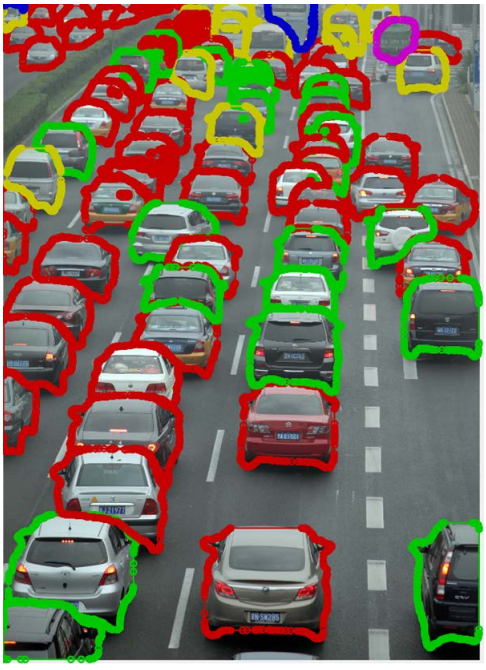
(b) Data augmentation results of the proposed

FIGURE 12. Multi-object data augmentation contrast experiment.

In order to better evaluate the performance of the proposed algorithm, we next conduct related experiments in the ultra-multi-object scenario.

The data augmentation comparison experiment in ultra-multi-object scenario is shown in Fig.13, which contains 77 vehicle objects and 6 types of vehicles. The data augmentation comparison experiment of ultra-multi-object is shown in Fig.13, which contains 77 vehicle objects and 6 kinds of vehicles. And there are large objects, small objects and severely occluded objects in Fig.13, which is a complex ultra-multi-object environment. As shown in Fig.13, Fig.13(a) is the result of manual labeling. Fig.13(b) is the result of the proposed algorithm. Through careful qualitative comparison,

(a) Manually label results



(b) Data augmentation results of the proposed method

FIGURE 13. Data augmentation comparative experiment in ultra-multi-object scenario.

the effect of the proposed method is close to that of manual labeling. Further, in terms of labeling time, manual labeling takes 10 minutes and 23 seconds, while the algorithm only needs 43 seconds. Finally, in the ultra-multi-object scenario, it is demonstrated that the proposed data augmentation algorithm has great performance, and the speed of the proposed method is much faster than manual labeling.

To summarize Part C, we qualitatively compare the effects of the proposed algorithm and manual labeling in single-object, multi-object, and ultra-multi-object scenarios. The results show that the proposed data augmentation algorithm can well replace manual labeling and can complete the data labeling task. At the same time, we find that the proposed method is much faster than the manual method in terms of speed.

D. SPEED COMPARISON EXPERIMENT OF VEHICLE IMAGE SEMANTIC SEGMENTATION DATA AUGMENTATION

In order to verify that the proposed algorithm has the advantages of high speed, and to accurately measure the labeling speed of the proposed algorithm in different scenarios, a speed comparison experiment was carried out.

First, we collect 20 hours and 40 minutes of vehicle videos in different scenes, and extract 10,880 unlabeled vehicle images from the vehicle videos. And the resolutions of these unlabeled vehicle images are all 1920*1080. Then, the 10,880 images are divided into four categories: no object, single-object, multi-object, and ultra-multi-object.

To explain further, no object is the absence of a vehicle in the image. Single-object means that there is only one vehicle in the image. Multi-object means that there are no more than 30 vehicles in the image. Ultra-multi-object means that the number of vehicles in the picture is more than 30. Since there are no vehicles in the no-object images, only single-object, multi-object, and ultra–multi-object need to be labeled. Mixed categories are unclassified, unlabeled vehicle images, which contain 4 classes of no-object, single-object, multi-object, and ultra-multi-object. In the mixed category, the number of no-object is more than half. Although images of no-object do not need to be labeled, they still take up time for manual labeling or algorithm recognition. Therefore, the mixed category not only considers the time for multi-object labeling, but also considers the time required to identify no-object pictures.

As shown in table 3, we calculated the average time and average speed of manual labeling and the proposed algorithm labeling, respectively. Average time is the average time to label a group of images. The average speed is the average speed for only labeling one image. In the manual labeling experiment, we selected 30 people to label images of three categories, all of whom are very skilled in labeling methods for semantic segmentation. Further, the average time and average speed of manual labeling in the three categories are obtained by statistics. In the labeling experiments of the proposed method, we use the proposed algorithm to label images with 30 sets of experiments for each category. Further, the average time and average speed of the proposed method are obtained by calculation. In the mixed category, if there is no vehicle in the picture, it goes directly to the next picture for labeling. The mixed category takes into account the time it takes to recognize no-object pictures.

As shown in table 3, in the single-object category, the speed of the proposed algorithm is 94.91% faster than that

TABLE 3. Data augmentation speed comparison experiments.

Categories	Amount of samples	method	Average time	Average speed/S
Single-object	3720	Manual labeling	11h 22min 54s	11.01
		Proposed algorithm	34min 37s	0.56
Multi-object	642	Manual labeling	37h 50min 50s	212.87
		Proposed algorithm	1h 17min 47s	7.27
Ultra-multi-object	356	Manual labeling	43h 7min 31s	436.1
		Proposed algorithm	2h 31min 11s	25.48
Mixed category	10880	Manual labeling	95h 46min 39s	31.69
		Proposed algorithm algorithm	4h 30min 27s	1.49

of manual labeling. In the multi-object category, the speed of the proposed method is 96.58% faster than that of manual labeling. In the Ultra-multi-object category, the speed of the proposed method is 94.16% faster than that of manual labeling. In the mixed category, the speed of the proposed method is 95.3% faster than that of manual labeling.

E. ACCURACY COMPARISON TESTS OF DIFFERENT METHODS

In part E, first, a qualitative verification test of the quality of data augmentation result is carried out. Second, a comparative experiment with other data augmentation algorithms was carried out. These two sets of experiments verify that the proposed algorithm has the advantage of high accuracy.

A direct way to verify the performance of data augmentation is to test the results of data augmentation in other networks. If the accuracy of the test network is improved after data augmentation, it means that the proposed data augmentation algorithm performs well. At the same time, it can also prove that the new dataset generated by the proposed data augmentation algorithm has great performance.

In qualitative verification experiments of the quality of data augmentation results, E-net is chosen as the test network.

Then, 800 pieces of data are manually labeled as the data set before data augmentation, which is called the original data, which is called the original data. Further, we perform data augmentation on the original data using the proposed algorithm and generate 2200 new labeled data. The original 800 pieces of data and these 2200 pieces of labeled data are merged to form a new dataset, called data-augmented data. Finally, the dataset is divided into training set, validation set and test set according to 8:1:1. The original data and the data-augmented data were sent to E-net for training, and the accuracy of the two data sets after training was compared. Table 4 shows the accuracy comparison experiment before and after data augmentation.

TABLE 4. Validation testing of the quality of data-augmented results.

Dataset	Amount of sample	A_{P50}	A_{P75}
Original data	800	0.832	0.36
Data-augmented data	3000	**0.924**	**0.41**

As shown in Table 4, after data augmentation, AP50 is increased from 0.832 to 0.924, and AP75 is increased from 0.36 to 0.41. Validation test results show that the proposed data augmentation algorithm can generate high-quality labeled datasets, which can improve the accuracy of semantic segmentation algorithms. This proves that the proposed algorithm has the advantage of high accuracy.

In order to verify that the proposed algorithm has the advantage of high precision compared with other methods, the proposed algorithm is compared with other data augmentation algorithms, which are DCGAN, WGAN, and traditional data augmentation methods. Further explanation, the traditional data augmentation algorithm performs data augmentation by cropping, flipping, changing the brightness, and panning and zooming the labeled data set. Traditional data augmentation methods create new data by simply transforming the original data, which preserves the characteristics of the original data to the greatest extent. DCGAN and WGAN are two deep learning-based data augmentation algorithms that generate new images through a generative adversarial network, and then manually label the new images as a result of data augmentation. In terms of accuracy, the comparison

TABLE 5. Accuracy comparison test of different data augmentation methods.

Method	A_{P50}	A_{P75}	A_R
Traditional Method	0.862	0.31	0.49
DCGAN	0.904	0.41	0.50
WCGAN	0.912	0.41	0.52
Proposed algorithm	**0.924**	**0.41**	**0.54**

results of the four data augmentation methods are shown in Table 5.

Where A_R is the average recall in the range of IoU from 0.5 to 0.95. The proposed algorithm is the data augmentation algorithm for semantic segmentation proposed in this paper. As shown in Table 5, the A_{P50} of the proposed method is 0.924. The A_{P75} of the proposed method is 0.4, and the A_R of the proposed method is 0.54. The proposed algorithm achieves the highest performance in three evaluation metrics. Therefore, compared with other data augmentation methods, this comparative test verifies that the proposed algorithm has the advantage of high accuracy.

VI. CONCLUSION

Manually labeling vehicle semantic segmentation datasets is slow, and training a well-performing data augmentation algorithm for vehicle image semantic segmentation requires a large amount of training data. Aiming at these problems, a fast and efficient data augmentation algorithm for vehicle image semantic segmentation based on LQE head is proposed. The proposed algorithm can simultaneously generate vehicle images and corresponding labels. Furthermore, the size of the label file is too huge, which can cause the calculation speed of the algorithm to decrease. Aiming at this problem, a lightweight algorithm for the label file based on BAS-DP is proposed. The lightweight algorithm can enormously reduce the size of the label file. The experimental results demonstrate that the proposed data augmentation algorithm can generate a high-quality vehicle semantic segmentation dataset with only a small amount of training data. In addition, the proposed algorithm performs well in the following three scenarios: single-object, multi-object and ultra-multi-object. Compared with other data enhancement algorithms, the proposed algorithm has three advantages, which are higher accuracy, faster speed, and less training data required.

The proposed algorithm also has a few limitations. Firstly, it belongs to supervised learning, so the proposed algorithm still requires a limited amount of labeled data for training. In the future, we will train an unsupervised data augmentation algorithm with unlabeled data. Secondly, the proposed data augmentation algorithm has been only applied to vehicle objects and has not been applied to other types of objects. For example, animal objects are different from vehicle objects in terms of appearance, so the performance of the proposed algorithm applied to animal objects is unknown. In the future, we will try to apply this data augmentation algorithm to animal objects and other types of objects.

ACKNOWLEDGMENT

(Fan Wang and Zhenyu Wang contributed equally to this work.)

REFERENCES

[1] H. Tayara, K. G. Soo, and K. T. Chong, "Vehicle detection and counting in high-resolution aerial images using convolutional regression neural network," *IEEE Access*, vol. 6, pp. 2220–2230, 2018.

[2] R. Gala, S. Verma, U. Kumar, and H. Ojha, "A survey of intelligent traffic light control systems," *Int. J. Comput. Appl.*, vol. 180, no. 21, pp. 31–36, Feb. 2018.

[3] J. Shin and M. Sunwoo, "Vehicle speed prediction using a Markov chain with speed constraints," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 3201–3211, Sep. 2019.

[4] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.

[5] E. Odat, J. S. Shamma, and C. Claudel, "Vehicle classification and speed estimation using combined passive Infrared/Ultrasonic sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1593–1606, May 2018.

[6] J. Zhang, X. Yin, J. Luan, and T. Liu, "An improved vehicle panoramic image generation algorithm," *Multimedia Tools Appl.*, vol. 78, no. 19, pp. 27663–27682, Oct. 2019.

[7] Y. Huang, X. Cao, Q. Wang, B. Zhang, X. Zhen, and X. Li, "Long-short-term features for dynamic scene classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 4, pp. 1038–1047, Apr. 2019.

[8] C. S. Lai, Z. Mo, T. Wang, H. Yuan, W. W. Y. Ng, and L. L. Lai, "Load forecasting based on deep neural network and historical data augmentation," *IET Gener., Transmiss. Distrib.*, vol. 14, no. 24, pp. 5927–5934, Dec. 2020.

[9] Y. Sun, J. Zhang, G. Li, Y. Wang, J. Sun, and C. Jiang, "Optimized neural network using beetle antennae search for predicting the unconfined compressive strength of jet grouting coalcretes," *Int. J. Numer. Anal. Methods Geomech.*, vol. 43, no. 4, pp. 801–813, Mar. 2019.

[10] L. Zhao and G. Shi, "A trajectory clustering method based on douglas-peucker compression and density for marine traffic pattern recognition," *Ocean Eng.*, vol. 172, pp. 456–467, Jan. 2019.

[11] L. Wu, X. Zhang, K. Wang, X. Chen, and X. Chen, "Improved high-density myoelectric pattern recognition control against electrode shift using data augmentation and dilated convolutional neural network," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 12, pp. 2637–2646, Dec. 2020.

[12] P. Tang, H. Wang, and S. Kwong, "G-MS2F: GoogLeNet based multi-stage feature fusion of deep CNN for scene recognition," *Neurocomputing*, vol. 225, pp. 188–197, Feb. 2017.

[13] 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. 2017.

[14] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.

[15] X. Zhang, Z. Wang, D. Liu, Q. Lin, and Q. Ling, "Deep adversarial data augmentation for extremely low data regimes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 15–28, Jan. 2021.

[16] K. Chen, X. Zhou, W. Xiang, and Q. Zhou, "Data augmentation using GAN for multi-domain network-based human tracking," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2018, pp. 1–4.

[17] J. Yebes, L. Bergasa, and M. García-Garrido, "Visual object recognition with 3D-aware features in KITTI urban scenes," *Sensors*, vol. 15, no. 4, pp. 9228–9250, Apr. 2015.

[18] Z. Hu, W. Fang, T. Gou, W. Wu, J. Hu, S. Zhou, and Y. Mu, "A novel method based on a mask R-CNN model for processing dPCR images," *Anal. Methods*, vol. 11, no. 27, pp. 3410–3418, Jul. 2019.

[19] Q. Wang, W. Min, Q. Han, Q. Liu, C. Zha, H. Zhao, and Z. Wei, "Inter-domain adaptation label for data augmentation in vehicle re-identification," *IEEE Trans. Multimedia*, vol. 24, pp. 1031–1041, 2022.

[20] Y. Bang, Y. Lee, and B. Kang, "Image-to-image translation-based data augmentation for robust EV charging inlet detection," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 3726–3733, Apr. 2022.

[21] W. Kim, W.-S. Jung, and H. K. Choi, "Lightweight driver monitoring system based on multi-task mobilenets," *Sensors*, vol. 19, no. 14, p. 3200, Jul. 2019.

[22] Z. Wang, J. Hu, G. Min, Z. Zhao, and J. Wang, "Data-augmentation-based cellular traffic prediction in edge-computing-enabled smart city," *IEEE Trans. Ind. Informat.*, vol. 17, no. 6, pp. 4179–4187, Jun. 2021.

[23] J. Li, D. Wang, S. Li, M. Zhang, C. Song, and X. Chen, "Deep learning based adaptive sequential data augmentation technique for the optical network traffic synthesis," *Opt. Exp.*, vol. 27, no. 13, p. 18831, Jun. 2019.

[24] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep CNNs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 9, pp. 2917–2931, Sep. 2020.

[25] Y. Fu, X. Li, and Y. Ye, "A multi-task learning model with adversarial data augmentation for classification of fine-grained images," *Neurocomputing*, vol. 377, pp. 122–129, Feb. 2020.
[26] J. K. Dumagpi and Y.-J. Jeong, "Evaluating GAN-based image augmentation for threat detection in large-scale Xray security images," *Appl. Sci.*, vol. 11, no. 1, p. 36, Dec. 2020.
[27] X. Ke, J. Zou, and Y. Niu, "End-to-end automatic image annotation based on deep CNN and multi-label data augmentation," *IEEE Trans. Multimedia*, vol. 21, no. 8, pp. 2093–2106, Aug. 2019.
[28] L. Li, L. I. Yan-Hui, Y. I. N. Lin-Lin, Y. A. N. G. Hui-Hua, F. E. N. G. Yan-Chun, Y. I. N. Li-Hui, and H. U. Chang-Qin, "Data augmentation of Raman spectral and its application research based on DCGAN," *Spectrosc. Spectral Anal.*, vol. 41, no. 2, p. 400, Feb. 2021.
[29] W. Fang, F. Zhang, V. S. Sheng, and Y. Ding, "A method for improving CNN-based image recognition using DCGAN," *Comput., Mater. Continua*, vol. 57, no. 1, pp. 167–178, 2018.
[30] C. Zhou, J. Zhang, and J. Liu, "Lp-WGAN: Using Lp-norm normalization to stabilize Wasserstein generative adversarial networks," *Knowl.-Based Syst.*, vol. 161, pp. 415–424, Dec. 2018.
[31] M. Zhu, B. Zang, L. Ding, T. Lei, Z. Feng, and J. Fan, "LIME-based data selection method for SAR images generation using GAN," *Remote Sens.*, vol. 14, no. 1, p. 204, Jan. 2022.
[32] Y. Zhang, S. Han, Z. Zhang, J. Wang, and H. Bi, "CF-GAN: Cross-domain feature fusion generative adversarial network for text-to-image synthesis," *Vis. Comput.*, Feb. 2022, doi: 10.1007/s00371-022-02404-6.
[33] K. Wei, T. Li, F. Huang, J. Chen, and Z. He, "Cancer classification with data augmentation based on generative adversarial networks," *Frontiers Comput. Sci.*, vol. 16, no. 2, pp. 1–11, Apr. 2022.

FAN WANG received the M.Sc. degree from the School of Electrical and Automation Engineering, Nanjing Normal University, Nanjing, China, in 2021. He is currently pursuing the Ph.D. degree with the School of Information and Management, Wuhan University, China. His major research interests include computer vision, machine learning, health misinformation, and human activity recognition.

ZHENYU WANG received the M.Sc. degree from the School of Electrical and Automation Engineering, Nanjing Normal University, Nanjing, China, in 2021. He is currently a Research and Development Engineer with NetEase, Hangzhou, China. His major research interests include computer vision, machine learning, and human activity recognition.