A Semantic Segmentation Network Simulating the Ventral and Dorsal Pathways of the Cerebral Visual Cortex

YUN WU, ZIMENG HUANG, HUIYUN LONG, GUANGQIAN KONG, XUN DUAN, AND JIANYONG JIANG
School of Computer Science and Technology, Guizhou University, Guiyang 550025, China
Corresponding author: Zimeng Huang (zimeng_huang@qq.com)

ABSTRACT
Aiming at the problem of spatial information loss in the semantic segmentation process, we propose a semantic segmentation network, termed the ventral and dorsal network (VDNet), which simulates the ventral and dorsal pathways of the cerebral visual cortex. The ventral pathway network focuses on extracting semantic information, and the dorsal pathway network focuses on extracting spatial information. We use the semantic enhancement module (SEM) in the ventral pathway network to fuse information of different scales to enhance the extraction of semantic information, and we use the spatial attention module (SAM) in the dorsal pathway network to assign weights to different locations in space to enhance the extraction of spatial information. By fusing the information of the two pathways, the final semantic segmentation result is obtained. Since the dorsal pathway network is used to specifically enhance the extraction of spatial information, the problem of spatial information loss during the segmentation process is effectively improved, and higher segmentation accuracy can be achieved by using only a small backbone network. On the CamVid, Cityscapes and PASCAL VOC 2012 datasets, we achieve the mean intersection over union (mIoU) of 82.1%, 77.8%, and 81.0%, respectively, which verifies the effectiveness of the proposed method.

INDEX TERMS Cerebral visual cortex, convolutional neural network, dorsal pathway, semantic segmentation, ventral pathway.

I. INTRODUCTION
Semantic segmentation refers to classifying the objects in an image at the pixel level to segment different objects. Semantic segmentation is one of the basic tasks of computer vision, and it has a wide range of applications in medical image processing, remote sensing classification, defect detection, automatic driving, video monitoring, etc.

Deep convolutional neural networks (DCNNs) [1] have powerful feature extraction capabilities. DCNNs gradually acquire rich semantic information through the superposition of convolutional layers and a continuous downsampling process. This is very useful for image classification tasks, and the abilities of current classification models [2] based on DCNNs have even surpassed those of humans. However, the semantic segmentation task not only needs to obtain the semantic information of each pixel but also needs to obtain the spatial information such as the position and shape of the objects. DCNNs will inevitably lose spatial information during the downsampling process, which will affect the segmentation accuracy.

To obtain higher segmentation accuracy, how to maintain spatial information while obtaining semantic information becomes an important consideration. There are currently three main ways to maintain spatial information:

1. Limiting the number of downsampling operations. Compared with the common model with up to 32 downsampling operations, DeepLabV3 [3] limits the downsampling to 16 operations, and ENet [4] limits the downsampling to 8 operations. This method is beneficial for preserving spatial information, but it also leads to the inability to obtain a larger receptive field, thereby reducing the richness of semantic
information. Although DeepLabV3 and ENet both use the atrous convolution to compensate for the lack of large receptive fields, due to the sparsity of the atrous convolution kernel, its effect is limited. 

(2) U-shaped structure. The FCN [5] is a pioneering work applying deep neural networks to semantic segmentation tasks. In the decoding stage, FCN-8s gradually merges 16 times and 8 times downsampled feature maps in the encoding stage to retrieve lost spatial information. U-Net [6] has made further improvements on the basis of the FCN. The network is designed as a symmetrical and complete U-shaped structure. In each stage of the decoder, the feature maps from the corresponding stages of the encoder are integrated. ExFuse [7] improves the U-shaped structure and proposes the semantic embedding branch (SEB) to involve more semantic information from high-level features to guide the feature fusion. 

(3) Multipath structure. BiSeNet [8] uses an independent, shorter path to obtain 8 times downsampled feature maps to supplement the missing spatial information. ICNet [9] uses three processing paths to input the original resolution, 1/2 resolution and 1/4 resolution images, respectively, and obtain feature maps with 1/8, 1/16, and 1/32 resolution, respectively, which can also supplement the spatial information. However, using the U-shaped structure or multipath structure alone has a limited ability to recover spatial information.

From a biological point of view, according to the research of Mishkin et al. [10], the visual cortex of monkeys is divided into two pathways. One leads to the ventral side of the brain, termed the ventral pathway, which is mainly responsible for object recognition; the other leads to the dorsal side of the brain, termed the dorsal pathway, which is mainly responsible for sensing the spatial positions of objects. We found that the functions of these two pathways correspond to the semantic information and spatial information needed in the semantic segmentation task, respectively.

Therefore, we are inspired by the ventral and dorsal processing pathways of the cerebral visual cortex, design a novel semantic segmentation model with a two-pathway structure and call the model the ventral and dorsal network (VDNet). The experimental results show that the model can effectively improve the problem of spatial information loss while acquiring rich semantic information to obtain more precise segmentation results than other methods.

Our contributions mainly include the following aspects:

(1) We design two pathways, one long and one short, to simulate the ventral and dorsal pathways of the cerebral visual cortex, respectively, to extract semantic information and spatial information and combine the U-shaped structure to obtain richer spatial information. Finally, the two pathways of information are fully fused to obtain high-precision segmentation results.

(2) We propose the semantic enhancement module (SEM), which enhances the extraction of semantic information from the ventral pathway by combining information of different scales.

(3) We propose the spatial attention module (SAM), which enhances the extraction of spatial information from the dorsal pathway by assigning weights to different positions in space.

II. RELATED WORK

A. SIMULATION OF THE BRAIN’S VISUAL MECHANISM

The biological visual system includes retinas, lateral geniculate nuclei, visual cortices, etc. The visual cortices include the ventral pathway composed of functional areas such as V1, V2, V4, and IT and the dorsal pathway composed of functional areas such as V1, V2, and V5. Kubilius et al. [11] proposed a deep convolutional neural network CORNet-S that simulates the ventral pathway of the cerebral visual cortex and the recurrence of some of its functional areas for image classification. Shibuya and Hotta [12] proposed a neural network model called Feedback U-Net for semantic segmentation. They recurrently calculate data on the entire network and use convolutional LSTM (long short-term memory) to fuse the data of two adjacent recurrences. The experimental results of the two models prove that the model is effective at simulating the visual mechanism of the brain.

B. SEMANTIC INFORMATION EXTRACTION

Semantic information is one of the two types of basic information required by semantic segmentation tasks, and it means the correct class to which each pixel in the image belongs. Current semantic segmentation models mostly use transfer learning methods, which use classification models that have been trained on large-scale datasets such as ImageNet [13] as pretrained models and use these pretrained models as backbone networks after removing the last fully connected layers. This method can enable better initialization of network parameters, which is very helpful for the extraction of semantic information. Generally, the better the classification performance of the backbone network that is used, the higher the accuracy of semantic segmentation will be. Commonly used backbone networks include VGG [14], ResNet [2], Xception [15], ResNeXt [16], EfficientNet [17], etc. These models have been proven to achieve good results on image classification tasks, and many semantic segmentation models [18]–[31] that use them as backbone networks have also obtained good results.

C. SPATIAL INFORMATION EXTRACTION

Spatial information, which refers to the position and shape of each object in the image, is another type of basic information required by semantic segmentation tasks. Convolutional neural networks will gradually lose spatial information in the downsampling process. At present, the following methods are the main methods used to retrieve the missing spatial information: (1) Limiting the number of downsampling operations, such as ENet, DeepLabV3, etc. However, this approach is not conducive to the extraction of semantic information. (2) Using the U-shaped structure to integrate the high-resolution feature maps, such as FCN-8s, U-Net, ExFuse, etc.
(3) Using additional pathways to extract spatial information, such as BiSeNet, ICNet, etc. However, only using a U-shaped structure or additional paths has a limited ability to restore spatial information.

**D. ATTENTION MECHANISM**

As we all know, instead of paying equal attention to the entire field of vision, human eyes always focus on a certain part when observing things, which helps us quickly obtain more important visual information. Hu et al. [32] proposed the squeeze-and-excitation (SE) block, which is a channel attention mechanism. The weight of each channel is obtained through global average pooling and the nonlinear calculation of the feature map, thus paying more attention to the more important channels. Woo et al. [33] proposed a two-dimensional spatial attention method, which assigns weights to each spatial position of the input feature map by calculating a two-dimensional feature map so that more attention can be paid to important positions in space. Wang et al. [34] proposed an integrated attention mechanism to distribute the probability weights of the key features, thereby enhancing the selection of features in the model. Cai and Wei [35] proposed a cross-attention mechanism that assigns horizontal weight coefficients to each row of features through a horizontal attention mechanism and assigns vertical weight coefficients to each column of features through a vertical attention mechanism. In addition, the differences between features are enlarged through the weight multiplication and maximum matching strategies. You et al. [36] constructed a graph convolutional network (GCN) under the attention mechanism such that the features that have high-probability weights are related to each other. The application of these attention mechanisms has brought significant performance improvements to the models.

The above studies clearly show that semantic information and spatial information are two types of basic information indispensable in the semantic segmentation process, and it is difficult for existing semantic segmentation models to take both into account. The model we propose simulates the visual mechanism of the brain using two pathways to extract semantic information and spatial information, respectively, thereby reducing the information loss. In addition, we applied the channel attention mechanism and the spatial attention mechanism in the two pathways, respectively, to strengthen the extraction of semantic information and spatial information. Experimental results show that the model proposed in this paper can extract both semantic information and spatial information.

**III. VENTRAL AND DORSAL NETWORK**

**A. NETWORK ARCHITECTURE**

The structure of VDNet, which is an encoder-decoder structure, is shown in Fig. 1. The encoder contains two pathways. We use the names of the two pathways in the cerebral visual cortex and call them the ventral pathway and the dorsal pathway. The two pathways input images independently. The ventral pathway has a longer path, which is mainly responsible for extracting semantic information and uses the SEM to enhance the extraction of information. The dorsal pathway has a shorter path and retains feature maps with a larger resolution, is mainly responsible for extracting spatial information, and uses the SAM to enhance the extraction of information. The decoder gradually merges the feature maps of the ventral pathway and the dorsal pathway from top to bottom to obtain the final segmentation result.

**B. VENTRAL PATHWAY**

The ventral pathway is composed of a pretrained backbone network and the SEM. VP-1, VP-2, VP-3, and VP-4 are different stages of the backbone network, and the 4 times, 8 times, 16 times, and 32 times downsampled feature maps are obtained, respectively. The ventral pathway is longer, and rich semantic information can be gradually obtained through more convolution and downsampling operations.

To enhance the semantic information extraction capability of the ventral pathway, we propose the SEM. The SEM is improved from the ASPP [19]. The ASPP uses convolution kernels with different atrous rates to capture information at different scales. The larger the atrous rate of the convolution kernel, the larger the receptive field. The small receptive field falling on the small targets can extract useful information. However, it cannot extract enough information when it falls on large targets. The large receptive field can extract more complete information from large targets. However, it may also cover multiple targets of different classes at the same time, resulting in the inability to extract useful information. In Fig. 2, we use boxes to represent the receptive fields with different sizes. The small solid box in Fig. 2a and the large solid box in Fig. 2b cover most of the area of a single object, so effective semantic information can be extracted. However, the dashed small box in Fig. 2a only covers a small part of the...
building, and the large dashed box in Fig. 2b covers multiple types of objects such as trees, pedestrians, buildings, sky, and poles. As a result, it is difficult to extract effective semantic information for both classes.

To avoid their own shortcomings as much as possible, it is necessary to make a trade-off between different sized receptive fields. Therefore, we introduce the channel attention mechanism into the ASPP. Through continuous learning, different weights are assigned to the feature maps from different receptive field convolutional layers so that the network can pay more attention to the important receptive fields and achieve the best information extraction effect.

The structure of the SEM is shown in Fig. 3. The SEM includes five parallel processing paths, including a $1 \times 1$ convolution, global average pooling, and three kinds of atrous convolutions with different atrous rates, to obtain different scales of receptive fields from local to global. “$1 \times 1$ Conv” represents the convolution operation with the $1 \times 1$ convolution kernel, and “$3 \times 3$ Conv” represents that with the $3 \times 3$ convolutional kernel. “BN” represents batch normalization [37]. “Rate” represents the atrous rate of convolution. “GAPooling” represents global average pooling. “Upsample” represents bilinear interpolation upsampling. “Concat” represents the concatenation operation in the channel dimension. We use the Swish [38] activation function. The structure of the CAM is shown in Fig. 4, where the “$\times$” symbol represents the dot product of two feature maps, and the “$+$” symbol represents the sum of two feature maps.

C. DORSAL PATHWAY

The dorsal pathway is composed of the pretrained ResNet18 and the SAM. We only use the part of ResNet18 that is cut to 8 times downsampling, and it is represented by DP-1 and DP-2. The dorsal pathway is shorter, which can retain richer spatial information, such as the boundaries of objects.

To enable the network to learn the importance of different locations in space and thereby enhance the representation of spatial information by the dorsal pathway, we improve the original spatial attention module [33] and propose our SAM.
As shown in Fig. 5, we visualize the intermediate feature maps before and after the SAM in the dorsal pathway.

Fig. 5b shows the ability of the dorsal pathway used to extract and retain spatial information such as object boundaries. The boundaries of objects such as cars, buildings, traffic lights, and poles are effectively extracted. From Fig. 5c, it can be seen that the boundary information has been enhanced, which reflects the SAM’s ability to strengthen spatial information.

The structure of the SAM is shown in Fig. 6, where “Mean” represents calculating the average value in the channel dimension to obtain the feature map with a single channel, and “Max” represents taking the maximum value in the channel dimension to obtain the feature map with a single channel. On the basis of the original spatial attention module, we added a branch that uses a $3 \times 3$ convolution to output a feature map with two channels to enhance the learning ability of the module. After connecting the feature maps of these three branches, a $3 \times 3$ convolution is used to output a feature map with a single channel. After being calculated by the sigmoid activation function, this feature map is multiplied and then added by the input feature map of the module to obtain the final output feature map of the module.

**D. DECODER**

We believe that the ability of the decoder to recover spatial information through learning is limited, so we focus on the fusion with high-resolution feature maps rich in spatial information. The decoder includes three fusion modules (FM) and four segmentation heads (SegHead). The feature maps of the ventral pathway and the dorsal pathway are gradually merged from top to bottom, thereby effectively fusing the semantic information and spatial information of the image. The structure of a FM is shown in Fig. 7. FM-1 has two inputs, FM-2 and FM-3 have three inputs. The SegHead follows FM-3. We use a simple design for the SegHead, and its structure with only two convolution layers is shown in Fig. 8. After the SegHead, the segmentation result of the input image size is obtained through 4 bilinear interpolation upsampling operations.
In addition to the SegHead after FM-3, we also add SegHeads after SEM, SAM, FM-1, and FM-2 to obtain segmented images to calculate the auxiliary loss.

### E. LOSS FUNCTION

Next is the design of the loss function. As shown in equation (1), we use the joint loss to train the network.

$$L = \lambda_1 L_1 + \lambda_2 L_2 + \lambda_3 L_3 + \lambda_4 L_4 + \lambda_5 L_5$$  

$L_1$, $L_2$, $L_3$, $L_4$, and $L_5$ are the respective losses calculated using the feature maps output by the SegHeads after FM-3, FM-2, FM-1, SEM, and SAM. As shown in equation (2), we use the focal loss [39].

$$FL(p, q) = - \sum_{i=1}^{N} p(x_i)(1 - q(x_i))^\beta \log(q(x_i))$$  

$N$ represents the total number of classes contained in the image. $p(x_i)$ represents whether a pixel is the $i$th class. If it is, $p(x_i)$ is equal to 1; otherwise, it is 0. $q(x_i)$ represents the probability that a certain pixel belongs to the $i$th class, which is predicted by the network. The value of $\beta$ is set to 2.

The later the network is, the richer the information contained, the greater the importance, and the greater the weight of the loss calculation. In the formula, $\lambda_1$, $\lambda_2$, $\lambda_3$, $\lambda_4$, and $\lambda_5$ are the weights of each loss, and we set them as 1, 0.8, 0.6, 0.4, and 0.2, respectively.

Using the joint loss function can shorten the gradient transfer distance, avoid the vanishing gradient problem, speed up the convergence speed of the model, and improve the final segmentation accuracy.

### IV. EXPERIMENTAL EVALUATION

#### A. DATASETS AND METRICS

We train and test our model on three challenging semantic segmentation datasets: CamVid [40], [41], Cityscapes [42], and PASCAL VOC 2012 [43].

The CamVid dataset is a road scene semantic segmentation dataset with 701 images in total, including 367 images in the training set, 101 images in the validation set, and 233 images in the test set. We use 11 segmentation classes including road, pedestrian, car, building, etc. The remaining classes are all classified into the 12th class and are ignored when calculating the segmentation accuracy. The resolution of the images is 720 pixels × 960 pixels. We use the training set and validation set to train our model and the test set for testing.

The Cityscapes dataset is a semantic segmentation dataset with 5000 finely annotated images, including 2975 images in the training set, 500 images in the validation set, and 1525 images in the test set. We use 19 segmentation classes including roads, buildings, sky, person, traffic light, etc. The remaining classes are all classified into the 20th class and are ignored when calculating the segmentation accuracy. The resolution of the images is 1024 pixels × 2048 pixels. We use the training set to train the model, the validation set to verify the model, and the test set to test the model. We upload the prediction results on the test set to the evaluation server of Cityscapes to obtain the segmentation accuracy.

The PASCAL VOC 2012 dataset involves 20 object classes and one background class. The original training set has 1464 images. The augmented training set with 10582 images is provided by [44]. The validation and test sets have 1449 and 1456 images, respectively. We employ the augmented training set and the validation set for training, and the test set for testing.

We use random horizontal flips and randomly scale the input images in training to perform data augmentation. We use 7 scales {0.5, 0.75, 1, 1.25, 1.5, 1.75, and 2} for all datasets. In addition, we employ multiscale and random horizontal flip testing to improve the segmentation accuracy. We use 6 scales {0.75, 1, 1.25, 1.5, 1.75, and 2}.

We use the pixel accuracy (pixAcc) and the mean intersection over union (mIoU) as the evaluation metrics. The mIoU metric is a more stringent metric since it penalizes false positive predictions. Therefore, we use mIoU as the main evaluation metric. The calculation method of pixAcc is shown in equation (3), and the calculation method of the mIoU is shown in equation (4).

$$\text{pixAcc} = \frac{\sum_{i=1}^{N} x_{ii}}{\sum_{i=1}^{N} T_i}$$  

$$\text{mIoU} = \left( \frac{\sum_{i=1}^{N} x_{ii}}{T_i + \sum_{j=1}^{N} (x_{ji} + x_{ij})} \right) / N$$  

$N$ represents the number of classes contained in the image; $T_i$ represents the total number of pixels of the $i$th class; $x_{ii}$ represents that the actual class is $i$ and the predicted class is also $i$, that is, the total number of pixels that are predicted correctly; and $x_{ij}$ represents that the actual class is $i$ and the predicted class is $j$, that is, the total number of pixels with incorrect predictions.

#### B. IMPLEMENTATION DETAILS

We use the deep learning framework Pytorch to train and test the model on a NVIDIA GeForce RTX3090 card and a NVIDIA GeForce GTX1080Ti card. We use the Adam as the optimizer with a weight decay of 1 × 10⁻⁴. We combine “warmup” and “cosine annealing [45]” strategies, and the learning rate is updated according to equation (5).

$$lr = \begin{cases} 
\frac{lr_{\text{max}} \text{ iters}}{\text{iters}_{\text{warm}}}, & T < T_{\text{warm}} \\
\frac{lr_{\text{max}} + (lr_{\text{min}} - lr_{\text{max}})}{2} \left(1 + \cos \left( \frac{T}{T_{\text{max}}} \pi \right) \right), & T \geq T_{\text{warm}} 
\end{cases}$$  

$lr_{\text{max}}$ represents the maximum value of the learning rate; $lr_{\text{min}}$ represents the minimum value of the learning rate and is set to 1/100 of $lr_{\text{max}}$; iters represents the current number of iterations; $T$ represents the current number of training rounds; $T_{\text{warm}}$ represents the number of “warmup” epochs and is set to 5, and iters_{warm} represents the corresponding number of iterations.

On the CamVid dataset, we use the original image with a resolution of 720 pixels × 960 pixels and conduct 200 epochs...
of training. We use ResNet18, ResNet50 and ResNeXt50 as the respective backbone networks. The batch size of training is set to 4. The maximum value of the learning rate is set to $4 \times 10^{-5}$.

On the Cityscapes dataset, we use ResNeXt50 as the backbone network. We first scale the image to a resolution of 512 pixels $\times$ 1024 pixels and conduct 200 epochs of training. The batch size of training is set to 12. The maximum value of the learning rate is set to $6 \times 10^{-5}$. Then we use the original image with a resolution of 1024 pixels $\times$ 2048 pixels and conduct 100 epochs of training. The batch size of training is set to 3. The maximum value of the learning rate is set to $6 \times 10^{-6}$.

On the PASCAL VOC 2012 dataset, we resize the original images to a resolution of 513 pixels $\times$ 513 pixels and conduct 200 epochs of training. We use ResNeXt50 as the backbone network. The batch size of training is set to 8. The maximum value of the learning rate is set to $5 \times 10^{-6}$.

Table 1 shows the size and number of channels of the output feature maps of each module of the network when the backbone network of the ventral pathway is ResNet18 and the CamVid dataset is used. In addition, the size of the output feature maps of the SegHeads used to obtain the auxiliary loss that is not listed in the table is 720 pixels $\times$ 960 pixels, and the number of channels is 12.

### Table 1. The size and number of channels of the output feature map of each module of the network.

| Module | Height $\times$ Width | Channels |
|--------|-----------------------|----------|
| VP-1   | 180 $\times$ 240      | 64       |
| VP-2   | 90 $\times$ 120       | 128      |
| VP-3   | 45 $\times$ 60        | 256      |
| VP-4   | 23 $\times$ 30        | 512      |
| SEM    | 23 $\times$ 30        | 256      |
| DP-1   | 180 $\times$ 240      | 64       |
| DP-2   | 90 $\times$ 120       | 128      |
| SAM    | 90 $\times$ 120       | 128      |
| FM-1   | 45 $\times$ 60        | 256      |
| FM-2   | 90 $\times$ 120       | 256      |
| FM-3   | 90 $\times$ 120       | 256      |
| SegHead| 720 $\times$ 960      | 12       |

### C. ABLATION STUDY

To prove the effectiveness of our design, we use ResNet18 as the backbone network and use CamVid for the following ablation experiments. We use the CamVid training set and validation set to train the model and use the CamVid test set to test the model.

1. We remove the dorsal pathway, and only retain the ventral pathway. As shown in Table 2, the pixAcc is 92.2%, which decreased by 2.6%; and the mIoU is 69.8%, which decreased by 6.5%. The table shows that the ability of the dorsal pathway to extract spatial information is helpful to improve the overall network performance.

2. We remove the SAM, and the FM-2 is changed to fuse the feature maps from VP-1, VP-2, and DP-2. As shown in Table 2, the pixAcc is 92.9%, which decreased by 1.9%; and the mIoU is 71.2%, which decreased by 5.1%. The table shows that the SAM is useful to enhance the spatial information in the dorsal path.

3. We remove the SEM, and FM-1 is changed to fuse the feature maps from VP-3 and VP-3. As shown in Table 2, the pixAcc is 91.8%, which decreased by 3.0%; and the mIoU is 72.6%, which decreased by 5.3%. The table shows that the enhancement of semantic information by the SEM is beneficial to improving the final segmentation accuracy.

4. We replace the SEM with the ASPP. As shown in Table 2, the pixAcc is 92.5%, which decreased by 2.3%; and the mIoU is 72.6%, which decreased by 3.7%. The table shows that our improvement to the ASPP is effective. Assigning the learned weights to different sized receptive fields can indeed achieve better information extraction results.

### D. EXPERIMENTAL RESULTS AND COMPARISON

Table 3 shows the segmentation accuracy of VDNet on the CamVid test set, including the intersection over union (IoU) of each class, the pixAcc, and the mean intersection over union (mIoU) of 11 classes. “Backbone” indicates the backbone network pretrained on the ImageNet dataset.

It can be seen that VDNet has high segmentation accuracy for large-scale objects such as the sky, roads, sidewalks, and cars, but it has low segmentation accuracy for small objects such as signs and poles. Compared with ResNet18, using ResNet50 and ResNeXt50 improves the mIoU by 1.97% and 2.67%, respectively.

As shown in Table 4, we compare our models with some other models, including ENet [4], DFANet B [20], DFANet A [20], SegNet [21], RTA [22], Dilation8 [23], ICNet [9], BiSeNet [8], PSPNet [24], DenseDecoder [25], SwiftNet [26] and VideoGCRF [27]. “∗” indicates the models are pretrained on Cityscapes. “∗” means that the metric was not given by the paper. Our model achieves the pixAcc of 95.7% and the mIoU of 82.1%, which is better than those of the compared models, indicating that our method is effective.

As shown in Table 4, we compare our models with some other models, including ENet [4], DFANet B [20], DFANet A [20], SegNet [21], RTA [22], Dilation8 [23], ICNet [9], BiSeNet [8], PSPNet [24], DenseDecoder [25], SwiftNet [26] and VideoGCRF [27]. “∗” indicates the models are pretrained on Cityscapes. “∗” means that the metric was not given by the paper. Our model achieves the pixAcc of 95.7% and the mIoU of 82.1%, which is better than those of the compared models, indicating that our method is effective.

Table 5 shows the segmentation accuracy of VDNet on the Cityscapes test set, including the IoU of each class and the mIoU of 19 classes. Our test results can be accessed on the evaluation server of Cityscapes (https://www.cityscapes-dataset.com/anonymous-results/?id=3d390bb00184b8aa30680644f8719fa870e9e078dcd946d3b53443138aad0e7d).
TABLE 3. Segmentation accuracy of VDNet on the CamVid test set.

| Backbone  | Bicyclist | Building | Car | Pole | Fence | Pedestrian | Road | Sidewalk | Sign | Sky | Tree | pixAcc (%) | mIoU (%) |
|-----------|-----------|----------|-----|------|-------|------------|------|----------|------|-----|------|------------|----------|
| ResNet18  | 69.05     | 90.38    | 93.63 | 43.11 | 64.79  | 74.61      | 96.54 | 90.32    | 42.74 | 93.59 | 80.65 | 94.81      | 76.31    |
| ResNet50  | 75.09     | 90.99    | 94.86 | 49.56 | 68.49  | 77.21      | 96.77 | 90.46    | 42.06 | 93.88 | 81.74 | 95.14      | 78.28    |
| ResNeXt50 | 73.94     | 91.52    | 94.98 | 48.93 | 69.61  | 77.57      | 96.69 | 90.37    | 48.62 | 94.02 | 82.51 | 95.29      | 78.98    |

FIGURE 9. Visualization examples of applying VDNet to the CamVid test set.

As shown in Table 6, we compare our model with some other models, including DeepLab [18], Dilation10 [23], LRR [28], DFANet B [20], DeepLabv2 [19], DFANet A [20], FRRN [46], RefineNet [29], SwiftNet [26], DeepLabv3 [3], and DeepLabv3+ [47]. Here we have an additional evaluation metric called IoU categories, which is given by the evaluation server of Cityscapes. It is the mIoU calculated by summing 19 classes into 7 major categories. The pixAcc is not given here because the evaluation server of Cityscapes does not include this metric. ‘’-‘’ means that the metric was not given by the paper. Our model achieves 90.6% IoU categories and 77.8% mIoU on the Cityscapes test set.

Table 7 shows the segmentation accuracy of VDNet on the PASCAL VOC 2012 test set, including the IoU of each class and the mIoU of 21 classes. Our test results can be accessed on the evaluation server of PASCAL VOC 2012 (http://host.robots.ox.ac.uk:8080/anonymous/WNPHHF.html).

As shown in Table 8, we compare the segmentation accuracy of FCN-8s [5], ESPNetv2 [48], DeepLab [18],
TABLE 4. Experimental results and comparison on the CamVid test set.

| Model          | Backbone | pixAcc (%) | mIoU (%) |
|---------------|----------|------------|----------|
| ENet          | no       | -          | 51.3     |
| DFANet B      | Xception B | -          | 59.3     |
| DFANet A      | Xception A | -          | 64.7     |
| SegNet        | VGG16    | 90.4       | 60.1     |
| RTA           | VGG16    | 90.4       | 62.5     |
| Dilation8     | VGG16    | -          | 65.3     |
| ICNet         | ResNet50 | -          | 67.1     |
| BiSeNet       | ResNet18 | -          | 68.7     |
| PSPNet        | ResNet50 | -          | 69.1     |
| DenseDecoder  | ResNetX101 | -           | 70.9     |
| SwiftNet      | ResNet18 | -          | 72.6     |
| VideoGCNF*    | ResNet101 | -           | 75.2     |
| VDNet*        | ResNetX50 | 95.3       | 79.0     |
| VDNet*        | ResNetX50 | 95.7       | 82.1     |

TABLE 5. Segmentation accuracy of VDNet on the cityscapes test set.

| Road          | Sidewalk | Building | Wall | Fence   | Terrain |
|---------------|----------|----------|------|---------|---------|
| 98.4          | 85.1     | 93.0     | 56.1 | 58.3    |         |
| Pole          | Traffic light | Traffic sign | Vegetation | 93.2 | 74.6 |
| 66.5          | 74.6     | 78.4     | 95.6 | 69.9    |         |
| Sky           | Person   | Rider    | Car  | Truck   |         |
| 95.5          | 85.6     | 69.2     | 95.6 | 69.9    |         |
| Bus           | Train    | Motorcycle | Bicycle | mIoU (%) |
| 76.0          | 71.1     | 66.3     | 75.1 | 77.8    |         |

TABLE 6. Experimental results and comparison on the cityscapes test set.

| Model          | Backbone | IoU Categories (%) | mIoU (%) |
|---------------|----------|--------------------|----------|
| DeepLab       | VGG16    | -                  | 63.1     |
| Dilation10    | VGG16    | 86.5               | 67.1     |
| LRR           | VGG16    | 88.2               | 69.7     |
| DFANet B      | Xception B | -                 | 67.1     |
| DFANet A      | Xception A | -                 | 71.3     |
| FRRN          | no       | -                  | 71.8     |
| RefineNet     | ResNet101 | -                | 73.6     |
| SwiftNet      | ResNet18  | -                  | 75.5     |
| DeepLabv3     | ResNet101 | -                | 81.3     |
| DeepLabv3+    | Xception  | -                  | 82.1     |
| VDNet*        | ResNetX50 | 90.6             | 77.8     |

TABLE 7. Segmentation accuracy of VDNet on the PASCAL VOC 2012 test set.

| Background | Aeroplane | Bicycle | Bird | Boat |
|------------|-----------|---------|------|------|
| 95.8       | 93.0      | 70.9    | 94.6 | 69.7 |
| Bottle     | Bus       | Car     | Cat  | Chair|
| 76.3       | 91.2      | 91.2    | 92.0 | 35.5 |
| Cow        | Dining table | Dog    | Horse | Motorbike |
| 88.1       | 71.8      | 87.7    | 91.6 | 89.7 |
| Person     | Potted plant | Sheep | Sofa | Train|
| 88.6       | 69.7      | 93.1    | 53.1 | 83.7 |

CRF-RNN [30], BoxSup [31], Dilation [23], DeepLabv2 [19], DeepLabv3 [3], and DeepLabv3+ [47]. Only the mIoU is used because the PASCAL VOC 2012 evaluation server only contains this metric. Our model achieves 81.0% mIoU on the PASCAL VOC 2012 test set.

E. QUALITATIVE RESULTS

Fig. 9 visualizes part of the segmentation results on the CamVid test set. It can be seen that the overall segmentation effect of VDNet is better, and the segmentation results for most objects such as pedestrians, cars, roads, sidewalks and even large poles are relatively fine. However, the segmentation of small objects such as small poles is still lacking.

Fig. 10 shows the partial segmentation results of VDNet on the Cityscapes validation set and the comparison with the segmentation results of DeepLabv2. We use the white dashed
box to show where VDNet has achieved a notable improvement compared to DeepLabv2. Fig. 11 shows the partial segmentation results of VDNet on the PASCAL VOC 2012 validation set and the comparison with the segmentation
---

TABLE 8. Experimental results and comparison on the PASCAL VOC 2012 test set.

| Model       | Backbone   | mIoU (%) |
|-------------|------------|----------|
| FCN-8s      | VGG16      | 67.2     |
| ESPNetv2    | no         | 68.0     |
| DeepLab     | VGG16      | 71.6     |
| CRF-RNN     | VGG16      | 74.7     |
| BoxSup      | VGG16      | 75.1     |
| Dilation    | VGG16      | 75.3     |
| DeepLabv2   | ResNet101  | 79.7     |
| DeepLabv3   | ResNet101  | 85.7     |
| DeepLabv3+  | Xception   | 87.8     |
| VDNet       | ResNetX150 | 81.0     |

---

results of DeepLab. We use the white dashed box to show where VDNet has achieved a notable improvement compared to DeepLab.

V. DISCUSSION

According to research [10], the cerebral visual cortex contains a ventral pathway and a dorsal pathway, which are respectively responsible for object recognition and spatial position perception. This exactly corresponds to the semantic information and spatial information required by the semantic segmentation task. Therefore, we propose a novel semantic segmentation model that uses two branches to simulate the ventral and dorsal pathways of the cerebral visual cortex. We achieve 82.1%, 77.8%, and 81.0% mIoU on the CamVid, Cityscapes, and PASCAL VOC 2012 datasets, respectively, proving that our method is effective. In addition, in order to strengthen the feature extraction capabilities of the two pathways, we respectively propose the SEM and SAM. The SEM, which can extract semantic information of different scales more efficiently, improves upon the ASPP. The SAM improves upon the original spatial attention module and can therefore be more difficult with a small number. Small objects are more likely to lose information in the calculation and thereby effectively improving the segmentation accuracy. In future works, we will continue to seek inspiration from the visual mechanism of the brain; explore and simulate the biological structures such as binocular input, recurrence structures, and the interaction of the ventral and dorsal pathways; and apply them to computer vision tasks such as semantic segmentation.

VI. CONCLUSION

The novel semantic segmentation network we proposed simulates the ventral and dorsal pathways of the cerebral visual cortex and achieves good segmentation results on the CamVid, Cityscapes and PASCAL VOC 2012 datasets. Through experimental verification, our proposed SEM and SAM can respectively strengthen the extraction of semantic information and spatial information, thereby effectively improving the segmentation accuracy. In future works, we will continue to seek inspiration from the visual mechanism of the brain; explore and simulate the biological structures such as binocular input, recurrence structures, and the interaction of the ventral and dorsal pathways; and apply them to computer vision tasks such as semantic segmentation.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[2] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[3] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, “Rethinking atrous convolution for semantic image segmentation,” 2017, arXiv:1706.05587. [Online]. Available: http://arxiv.org/abs/1706.05587 and https://ui.adsabs.harvard.edu/abs/2017arXiv170605587C.

[4] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, “ENet: A deep neural network architecture for real-time semantic segmentation,” 2016, arXiv:1606.02147. [Online]. Available: http://arxiv.org/abs/1606.02147 and https://ui.adsabs.harvard.edu/abs/2016arXiv160602147P.

[5] E. Shelhamer, J. Long, and T. Darrell, “Fully convolutional networks for semantic segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 4, pp. 640–651, Apr. 2017, doi: 10.1109/TPAMI.2016.2572683.

[6] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent., Oct. 2015, pp. 234–241.

[7] Z. Zhang, X. Zhang, C. Peng, X. Xue, and J. Sun, “ExFuse: Enhancing feature fusion for semantic segmentation,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 273–288.

[8] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, “BiSeNet: Bilateral segmentation network for real-time semantic segmentation,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 334–349.

[9] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, “ICNet for real-time semantic segmentation on high-resolution images,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 418–434.

[10] M. Mishkin, L. G. Ungerleider, and K. A. MacKo, “Object vision and spatial vision: Two cortical pathways,” Trends Neurosci., vol. 6, pp. 414–417, Jan. 1983, doi: 10.1016/0166-2236(83)90190-X.

[11] J. Kubilius, M. Schrimpf, K. Kar, H. Hong, N. J. Majaj, R. Rajalingham, E. B. Issa, P. Bashivan, J. Prescott-Roy, K. Schmidt, A. Nayebi, D. Bear, D. L. K. Yamin, and J. J. DiCarlo, “Brain-like object recognition with high-performing shallow recurrent ANNs,” 2019, arXiv:1909.06161. [Online]. Available: http://arxiv.org/abs/1909.06161 and https://ui.adsabs.harvard.edu/abs/2019arXiv190906161K.

[12] E. Shibuya and K. Hotta, “Feedback U-net for cell image segmentation,” 2020, arXiv:2004.14581. [Online]. Available: http://arxiv.org/abs/2004.14581 and https://ui.adsabs.harvard.edu/abs/2020arXiv200404581S.

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.

[14] K. Simon and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, arXiv:1409.1556. [Online]. Available: http://arxiv.org/abs/1409.1556 and https://ui.adsabs.harvard.edu/abs/2014arXiv1409.1556S.
[15] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1800–1807, doi: 10.1109/CVPR.2017.195.

[16] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 5967–5995, doi: 10.1109/CVPR.2017.634.

[17] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in Proc. Int. Conf. Mach. Learn. (ICML), Jun. 2019, pp. 6105–6114.

[18] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Semantic image segmentation with deep convolutional nets and fully connected CRFs,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, Apr. 2018, doi: 10.1109/TPAMI.2017.2691814.

[19] H. Li, P. Xiong, H. Fan, and J. Sun, “DFANet: Deep feature aggregation for real-time semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 9514–9523, doi: 10.1109/CVPR.2019.00975.

[20] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A deep convolutional encoder-decoder architecture for image segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2646615.

[21] P.-Y. Huang, W.-T. Hsu, C.-Y. Chiu, T.-F. Wu, and M. Sun, “Efficient uncertainty estimation for semantic segmentation in videos,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Oct. 2018, pp. 536–552.

[22] F. Yu and V. Koltun, “Multi-scale context aggregation by dilated convolutions,” 2015, arXiv:1511.07122 [Online]. Available: http://arxiv.org/abs/1511.07122.

[23] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 6230–6239, doi: 10.1109/CVPR.2017.660.

[24] P. Bilinski and V. Prisacariu, “Dense decoder shortcut connections for single-pass semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2018, pp. 6596–6605, doi: 10.1109/CVPR.2018.00690.

[25] M. Oršić, I. Krešo, P. Bevandic, and S. Šegvic, “In defense of pre-trained ImageNet architectures for real-time semantic segmentation of road-driving images,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 12599–12608, doi: 10.1109/CVPR.2019.01289.

[26] S. W. Savare, C. C. Fowlkes, and I. Kokkinos, “Deep spatio-temporal random fields for efficient video segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2018, pp. 8915–8924, doi: 10.1109/CVPR.2018.00929.

[27] G. Ghiasi and C. C. Fowlkes, “Laplacian pyramid reconstruction and refinement for semantic segmentation,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2016, pp. 519–534.

[28] YUN WU was born in Jingdezhen, China, in 1973. He received the B.E. degree in computer science from the Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China, in 1999, and the M.E. and Ph.D. degrees in computer science from Guizhou University, Guiyang, China, in 2006 and 2009, respectively.

[29] J.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Oct. 2018, pp. 833–851.

[30] S. Mehta, M. Rastegari, L. Shapiro, and H. Hjärsheimer, “ESPNets2: A lightweight, power efficient, and general purpose convolutional neural network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 9182–9192, doi: 10.1109/CVPR.2019.00941.

[31] W. Cai and Z. Wei, “Remote sensing image classification based on a cross-attention mechanism and graph convolution,” IEEE Geosci. Remote Sens. Lett., early access, Oct. 1, 2020, doi: 10.1109/LGRS.2020.3026587.

[32] H. You, S. Tian, L. Yu, and Y. Lv, “Pixel-level remote sensing image recognition based on bidirectional word vectors,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 2, pp. 1261–1293, Feb. 2020, doi: 10.1109/TGRS.2019.2945591.

[33] J. Hu, S. Sun, D. Feng, and H. Ling, “An improved convolutional neural network for remote sensing image classification,” IEEE Geosci. Remote Sens. Lett., vol. 16, no. 1, pp. 52–56, Jan. 2019, doi: 10.1109/LGRS.2018.2887256.

[34] M. Oršić, I. Krešo, P. Bevandic, and S. Šegvic, “In defense of pre-trained ImageNet architectures for real-time semantic segmentation of road-driving images,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 12599–12608, doi: 10.1109/CVPR.2019.01289.

[35] F. Yu and V. Koltun, “Multi-scale context aggregation by dilated convolutions,” 2015, arXiv:1511.07122 [Online]. Available: http://arxiv.org/abs/1511.07122.

[36] Y. Wu, “Remote sensing image classification based on a cross-attention mechanism and graph convolution,” IEEE Geosci. Remote Sens. Lett., early access, Oct. 1, 2020, doi: 10.1109/LGRS.2020.3026587.
ZIMENG HUANG received the B.S. degree in pharmaceutics from the School of Pharmaceutical Science, Guizhou University, Guizhou, China, in 2017. He is currently pursuing the M.E. degree in computer science and technology with the School of Computer Science and Technology, Guizhou University, Guizhou. His research interests include deep learning, computer vision, image processing, and image semantic segmentation.

HUIYUN LONG received the Ph.D. degree in computer science from Guizhou University, Guiyang, China, in 2009. She is currently an Associate Professor and the Associate Dean of the School of Computer Science and Technology, Guizhou University. Her main research interests include process algebra, formalization of web services, and deep learning.

GUANGQIAN KONG received the B.S. degree from Central South University, Hunan, China, in 1996, and the M.S. and Ph.D. degrees from Guizhou University, Guizhou, China, in 2002 and 2009, respectively. He is currently an Associate Professor, a Graduate Supervisor, and a member of the China Computer Federation. His research interests include computer networks, big data, deep learning, and their applications.

XUN DUAN received the B.S. degree from the Chongqing University of Posts and Telecommunications in 1995, the Ph.D. degree in computer science from Guizhou University, in 2007. He has been an Associate Professor with the School of Computer Science and Technology, Guizhou University, since 2007. His current interests include big data, deep learning, and computer vision.

JIANYONG JIANG received the B.Admin. degree in information management and information system from the School of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou, China, in 2018. He is currently pursuing the M.E. degree in computer technology with the School of Computer Science and Technology, Guizhou University, Guizhou, China. His research interests include deep learning, computer vision, image recognition, and image semantic segmentation.