



A sketch semantic segmentation method using novel local feature aggregation and segment-level self-attention

Lei Wang¹ · Shihui Zhang^{1,2} · Wei Wang¹ · Weibo Zhao¹

Received: 28 June 2022 / Accepted: 21 March 2023 / Published online: 8 April 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

Sketch semantic segmentation presents great challenges, since sketches have simpler appearances and more levels of abstraction than natural images. To overcome these challenges, we propose a sketch semantic segmentation method. Concretely, we treat a sketch as a 2D point set and exploit the structures of strokes and the spatial position relationship among 2D points to develop a novel local feature aggregation module. The novel local feature aggregation module encodes informative local features, which are highly useful to analyze semantics. And we define “stroke distance” to balance the two-dimensional spatial distributions of sketches and the internal structures of strokes. Simultaneously, we design a segment-level self-attention module to establish and enhance the relationship between segments by encoding the contents and positions of segment features. Further, based on the above two modules, we construct a similar encoder–decoder structure with two sub-branches, which retains the features of the significant points and integrates the features of several intermediate stages by utilizing a global multi-scale mechanism. Finally, the two outputs of the two sub-branches are fused to obtain the final sketch semantic segmentation result. Extensive experiments on SPG and SketchSeg-150K show that our method achieves state-of-the-art results.

Keywords Sketch semantic segmentation · Novel local feature aggregation · Segment-level self-attention · Encoder–decoder structure · Global multi-scale mechanism

1 Introduction

Freehand sketching is one of the most intuitive and convenient communication ways, which has been popular since ancient times. In recent years, people has given rise to the creations of sketches by this way, with the popularity of touch-screen devices (e.g. tablets and smartphones) and the emergence of drawing programs. Obviously different from natural images, sketches are composed of several strokes drawn by humans instead of being captured by cameras. Accordingly, the contents of sketches have the sparsity and present multiple levels of abstraction and different drawing styles. At present, the studies related to sketches mainly include sketch recognition [1–3], sketch-based image retrieval [4–6], sketch-based 3D object retrieval [7, 8], sketch semantic segmentation [9, 10], sketch caption [11] and sketch synthesis [12, 13]. Among them, sketch semantic segmentation is the understanding for a sketch from the fine-grained perspective and part perspective, which is of great help to analyze the content of the sketch

✉ Shihui Zhang
sshzz@ysu.edu.cn

Lei Wang
ysu_wanglei6@stumail.ysu.edu.cn

Wei Wang
973667927@qq.com

Weibo Zhao
zhaowb@stumail.ysu.edu.cn

¹ School of Information Science and Engineering, Yanshan University, West section, Hebei Street, Qinhuangdao 066004, Hebei Province, China

² Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province, Yanshan University, West section, Hebei Street, Qinhuangdao 066004, Hebei Province, China

and plays an important role in the realization of other sketch tasks. Sketch semantic segmentation facilitates the discovery of the abstract drawing mechanism and the development of human vision perception. Besides, sketch semantic segmentation can automatically and efficiently label part or whole semantics to provide users with a friendly human-computer interaction manner in the computer-aided design system [14, 15]. As a preprocessing step, sketch semantic segmentation can also assist users to accomplish fine-grained retrieval and to improve the performance of sketch tasks. Figure 1 shows the application of sketch semantic segmentation to fine-grained image or 3D retrieval.

Early sketch semantic segmentation mainly depends on hand-crafted features and traditional complicated models such as radial basis functions [16], graph models [17] and conditional random field (CRF) [18]. These methods handle with large varieties in the appearances of sketches difficultly, with the limited representative power and high computational cost. Consequently, researchers begin to introduce deep neural network into sketch semantic segmentation task. According to the specific formats of sketch inputs, these methods are divided into three categories including sequence-based methods, image-based methods and point-based methods. The sequence-based methods [19–21] regard a sketch as a stroke sequence to encode the sketch based on a sequence model such as a recurrent neural network (RNN). The image-based methods [22, 23] directly transform a sketch semantic segmentation issue into an image semantic segmentation issue, which is solved by establishing convolution neural network (CNN)

combined with sketch characteristics. This kind of methods treat a sketch as a static image, so they discard stroke information. The point-based methods [10, 24, 25] regard a sketch as a set of points and generate a point-wise feature map for semantic segmentation by continuously aggregating local features and establishing an encoder–decoder structure based on these features. For convenience, we call the component that can continuously aggregate local features as a local feature aggregation module. Obviously, the local feature aggregation module is significant to solve the sketch semantic segmentation issue. However, the common local feature aggregation modules perceive the surrounding regions of the sampling points based on Euclidean distance, which ignores the internal structures of strokes and the continuity of strokes. Therefore, we design a novel local feature aggregation (NLFA) module to perceive the surrounding points by defining “stroke distance”. In addition, most common local feature aggregation modules lose the relative position information when searching the surrounding points of the sampling points, which obviously violates the important idea of extracting local features by CNN. Accordingly, we capture the relative position information and stroke information to strengthen the related semantics. Compared with the common local feature aggregation modules, the encoded information in the proposed module has richer contents, which are obviously helpful to understand semantics.

Generally, the relationship between segments can describe a sketch partly. However, common sketch semantic segmentation methods pay little attention to the relationship between segments. Thus, we design a segment-

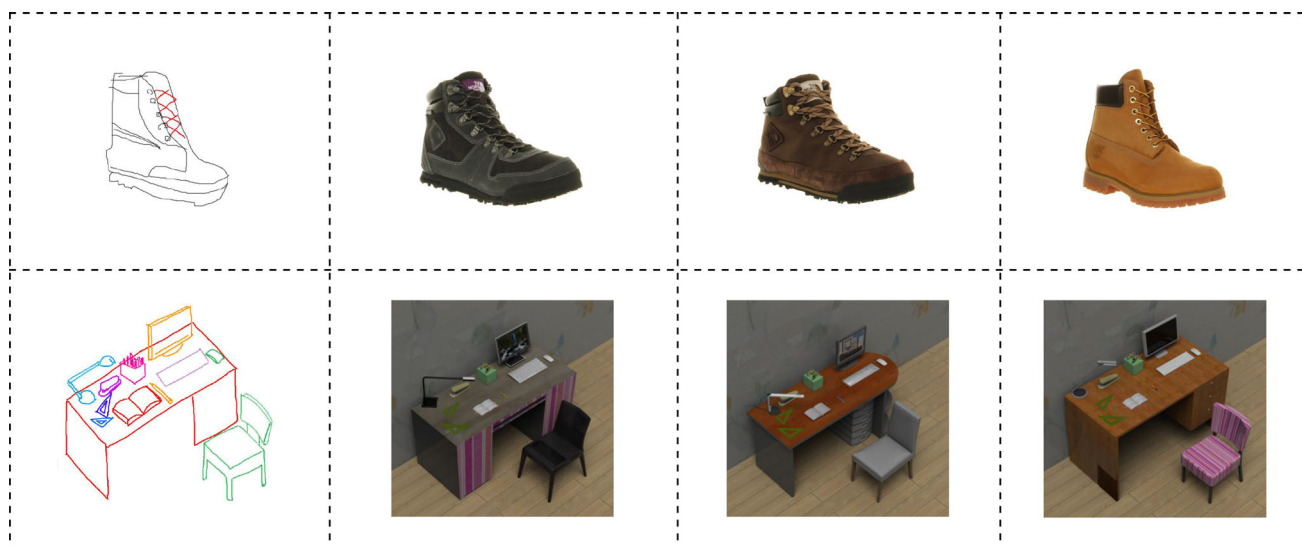


Fig. 1 The applications of sketch semantic segmentation. Humans can retrieve some images and 3D objects according to the labeling details or parts. One possible solution is that we use sketch semantic

segmentation method to obtain the important concerned semantics and then improve the ranking of images with the concerned semantics in the retrieval results

level self-attention (SLSA) module based on the idea of multi-head self-attention by encoding the contents and positions of segment features. The SLSA module can discover the relationship between segments automatically for capturing the internal structure of a sketch.

Further, we propose a sketch semantic segmentation method using the NLFA module and the SLSA module. Specifically, the method is based on a similar encoder–decoder structure with two sub-branches named point-level sub-branch and segment-level sub-branch. In the encoding part, the NLFA module is exploited to extract informative local features. In the decoding part, a global multi-scale mechanism is exploited to aggregate the local features and several intermediate-stage features as global features. What's more, we only perform a dimension reduction operation to label each point in the point-level sub-branch while we establish and enhance the segment-level relationship in the segment-level sub-branch. Then, the point-level sub-branch and the segment-level sub-branch consume global features to generate two sketch semantic segmentation maps. Finally, the two maps are fused to obtain the final segmentation result.

In short, the main contributions of this paper could be summarized as the following threefold. (1) We develop a NLFA module to capture local features of the sampling points adequately. Compared with existing local feature aggregation modules, it is a unique local module that fully considers the position information of the sampling points and the internal structures of strokes. (2) We design a SLSA module to establish and enhance the relationship between segments. Both the local and global position information of each segment are encoded as the position embedding vectors to describe the internal structures of sketches more precisely. (3) We construct a similar encoder–decoder structure with two sub-branches and propose a sketch semantic segmentation method based on the structure. The proposed method realizes a global multi-scale mechanism for the first time, which fully integrates the sketch features of several intermediate stages. Most importantly, the proposed method labels the semantics of each point from a point-level perspective and a segment-level perspective simultaneously, which is rarely involved in the previous sketch semantic segmentation methods.

The rest of this article is organized as follows: The related work is discussed in Sect. 2. The proposed sketch semantic segmentation method is described in Sect. 3; Sect. 4 states the experimental operations and results; Sect. 5 summarizes the full texts and future prospects.

All the significant abbreviations and the corresponding full names used in this work are listed in Table 1.

Table 1 The abbreviations and the corresponding full names

Abbreviation	Full name
NLFA	Novel local feature aggregation
SLSA	Segment-level self-attention
CRF	Conditional random field
RNN	Recurrent neural network
CNN	Convolution neural network
MLP	Multi-layer perceptron
FPS	Farthest point sampling
FLOPs	Floating point of operations

2 Related work

Sketch semantic segmentation is essentially to classify each point of a sketch. Labeling each point as different semantics brings benefits to many applications such as sketch-based image retrieval [4–6], sketch caption [11] and sketch scene segmentation [9]. Many researchers focus on this research. Sun et al. [17] propose a novel sketch segmentation method by using the low-level perception based on the proximity and the high-level knowledge based on the past experience. Huang et al. [26] develop a data-driven approach by designing a mixed integer programming and using 3D template models. Schneider et al. [18] utilize a heuristic method to establish the graph structures of different strokes and calculate a semantic labeling map based on CRF.

At present, most scholars study sketch semantic segmentation based on deep learning. Wu et al. [19] transform the sketch semantic segmentation issue into a sequence-to-sequence generation issue and convert the sequence of strokes into the corresponding semantic labels based on RNN. Li et al. [22] design an hourglass-shaped CNN combined with the post-processing steps of multi-label graph cuts to improve the segmentation results. Among the deep learning methods, the point-based methods are becoming more and more popular because of their low computational cost and high segmentation accuracy. Wang et al. [24] directly take the sampling points as the input and propose a multi-column point convolutional neural network. The network uses multiple columns with different convolution kernel sizes to better capture the sketch structure. Based on graph convolution neural network, Yang et al. [10] obtain a semantic segmentation map by constructing a static graph convolution unit, a dynamic graph convolution unit and a mix pooling module. Our method is a sketch semantic segmentation method with points as the input using NLFA and SLSA. Compared with the above point-based methods, the proposed method can

achieve more abundant sketch features and take into account the two-dimensional spatial distributions of sketches and the internal structures of strokes.

Sketch semantic segmentation methods are closely related to image semantic segmentation methods and partly follow the development of image semantic segmentation methods. At present, most of image semantic segmentation methods are based on deep neural network. Long et al. [27] transfer a classification network into an end-to-end full convolution network to generate a pixel-wise output for semantic segmentation. Vijay et al. [28] propose a deep convolution encoder–decoder network SegNet. The innovation of SegNet is that the decoder uses pooling indices to deal with nonlinear up-sampling operations. Chen et al. [29] develop a DeepLab system by using an atrous algorithm and a fully connected CRF to solve two technical problems of image labeling including signal down-sampling and spatial invariance. Afterward, they also develop some variants of DeepLab including deeplabv2 [30], deeplabv3 [31] and deeplabv3+ [32]. Ronneberger et al. [33] design a more elegant u-shaped framework U-Net for biomedical image segmentation. The framework captures the context by a contracting path and realizes precise localization with a symmetric extending path. Similar to the DeepLab system, some scholars improve U-Net and propose attention U-Net [34], U-Net++ [35] and R2U-Net [36]. Zhang et al. [37] investigate a densely connected neural architecture search framework, which can directly search the optimal structure to represent the multi-scale visual information over a large-scale target dataset. Although the above methods have achieved the impressive performance in the field of image semantic segmentation, for the task of sketch semantic segmentation, these methods generally have the disadvantages of more parameters and higher computational cost. Moreover, for the sparsity of sketches, a large number of blank spaces can be ignored [24], so it is not effective to directly generalize image semantic segmentation methods to sketch field.

3 Methodology

3.1 Overview of the proposed method

Given the sampling points of a sketch, sketch semantic segmentation aims to classify these sampling points so that the points of the same type correspond to the same semantic part of the sketch. To accomplish this task, we propose a sketch semantic segmentation method using a NLFA module and a SLSA module. Figure 2 shows the pipeline of the proposed method with a similar encoder–decoder structure. A sketch is converted into a point set using the farthest point sampling method. The encoding

part is used to extract deep features of the point set, and the decoding part generates a semantic segmentation map according to these features. Specifically, the NLFA module determines the surrounding regions of the sampling points using Euclidean distance or “stroke distance” randomly, so that the proposed method can better adapt to the two-dimensional spatial distributions of sketches and the internal structures of strokes. Then, the NLFA module extracts rich local features such as absolute position information, relative position information and stroke information, which give better play to the performance of point-based modality. In general, the encoding part is composed of four NLFA modules which correspond to the first four stages of the similar encoder–decoder structure while the decoding part is composed of five stages. The outputs of four intermediate stages are concatenated to feed into two different sub-branches for the final semantic segmentation result. There are two loss functions at the ends of the sub-branches totally, which are combined together in the training process. The joint loss is expressed in Eq. (1).

$$L = L_{\text{point-level}} + L_{\text{segment-level}} \quad (1)$$

$L_{\text{point-level}}$ is a cross-entropy loss for learning point-level semantics at the point-level sub-branch. The inputs of the loss are a predicted semantic segmentation map and the ground-truth semantic segmentation map with $[B, N, C]$ dimension, where B is batch size, N is the number of sampling points, and C is the number of semantic labels. $L_{\text{point-level}}$ is expressed in Eq. (2).

$$L_{\text{point-level}} = - \sum_{i=1}^B \sum_{j=1}^N \sum_{k=1}^C y_{i,j,k} \log(\hat{y}_{i,j,k}) \quad (2)$$

where $y_{i,j}$ is a one-hot vector. $y_{i,j,k}$ is equal to 1 when the label of $y_{i,j}$ is k while all other elements of $y_{i,j}$ are 0. $L_{\text{segment-level}}$ is also a cross-entropy loss to learn the semantics of each segment. Its input is the output of the SLSA module to enhance and learn the relationship between segments. Similar to the point-level loss, $L_{\text{segment-level}}$ is defined in Eq. (3).

$$L_{\text{segment-level}} = - \sum_{i=1}^B \sum_{j=1}^S \sum_{k=1}^C y_{i,j,k} \log(\hat{y}_{i,j,k}) \quad (3)$$

where S represents the number of segments. From the above discussion, it can be seen that the proposed method is mainly realized by a NLFA module, a SLSA module and a similar encoder–decoder structure based on the two modules. We will detail these two modules and the structure in the following texts.

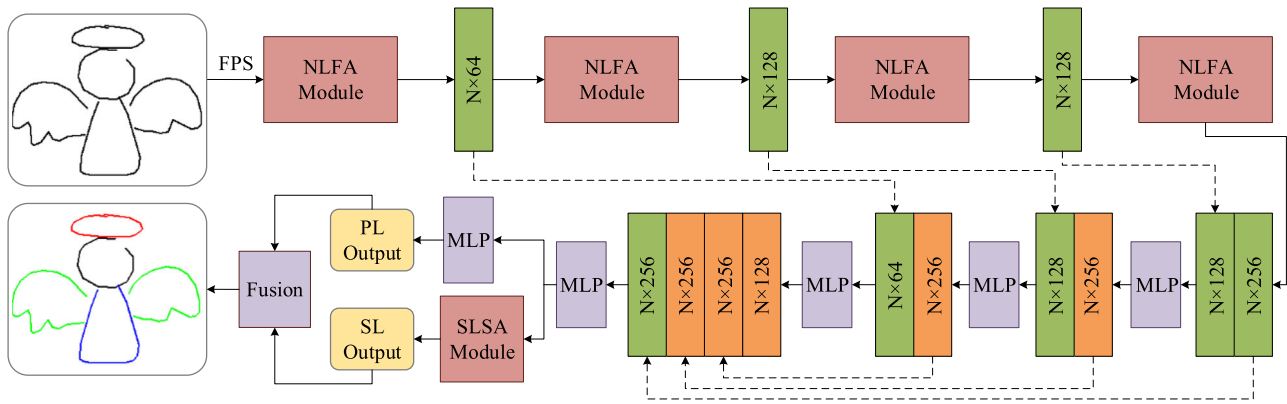


Fig. 2 The pipeline of the propose method. The green rectangles are the encoded features, and the orange rectangles are the decoded features. The NLFA module indicates a novel local feature aggregation module, and the SLSA module indicates a segment-level self-attention module. MLP is a multi-layer perception, and the dotted

lines are concatenation operations. PL output is a point-level sketch semantic segmentation map, and SL output is a segment-level sketch semantic segmentation map. FPS is the farthest point sampling method

3.2 Novel local feature aggregation module

Local feature aggregation module is an important component of a sketch semantic segmentation method with points as the input. The internal process of NLFA module is shown in Fig. 3. For convenience, we ignore the parameter batch size B . It can be seen from Fig. 3 that the input point p_i uses a perception procedure P to perceive the surrounding regions of p_i using different dilated rates r . Based on the position relationship between the surrounding points and the input point p_i , a feature calculation procedure C is used to encode the sketch features with wealthy information. Finally, the encoding procedure is completed by multi-layer perceptron and max-pooling operation. Thus, a NLFA module mainly includes a region perception

procedure P and a feature calculation procedure C. The NLFA module acts on all center points and encodes local features by acquiring several neighboring points around the center points.

3.2.1 Region perception procedure

The region perception procedure plays a crucial role when the sampling points perceive their surrounding regions in a local feature aggregation module. The common region perception procedure based on Euclidean distance combined with K -nearest neighbor algorithm is shown in Fig. 4a.

In Fig. 4a, $p_i(x_i, y_i)$ is a center point, and $p_{ij}(x_{ij}, y_{ij})$ represents any surrounding point of p_i ; then, the

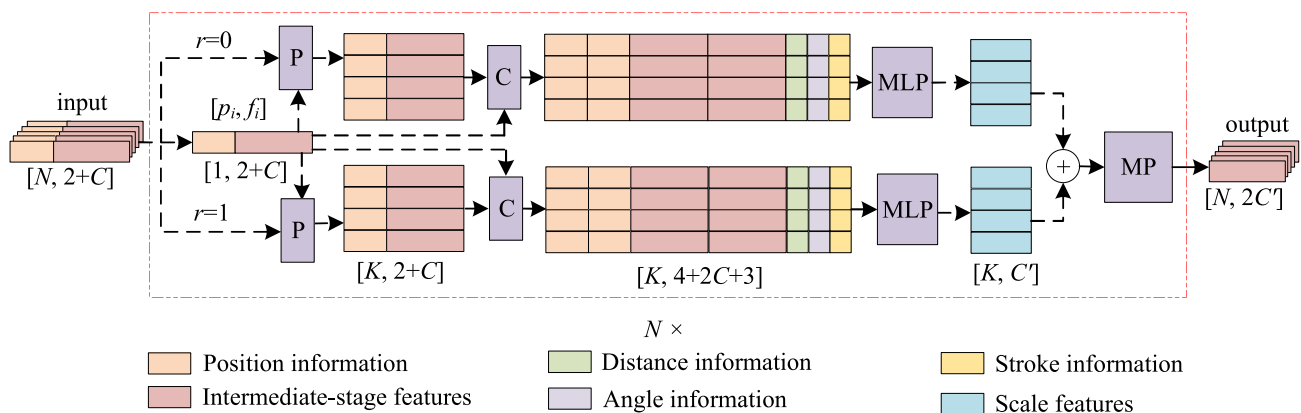


Fig. 3 The internal process of encoding sketch features by a NLFA module. P represents a perception procedure for points, which perceives the surrounding points of the sampling point p_i by using Euclidean distance or “stroke distance” combined with the K -nearest neighbor algorithm. C represents a feature calculation procedure, which is utilized to collect and calculate distance information, angle

information, stroke information, etc. MLP stands for a multi-layer perceptron. MP stands for a max-pooling operation. \oplus is a concatenation operation. r is a dilated rate to realize a multi-scale technology, where r is set to 0 and 1. $N \times$ indicates that the process in the red dotted box is performed N times due to N points

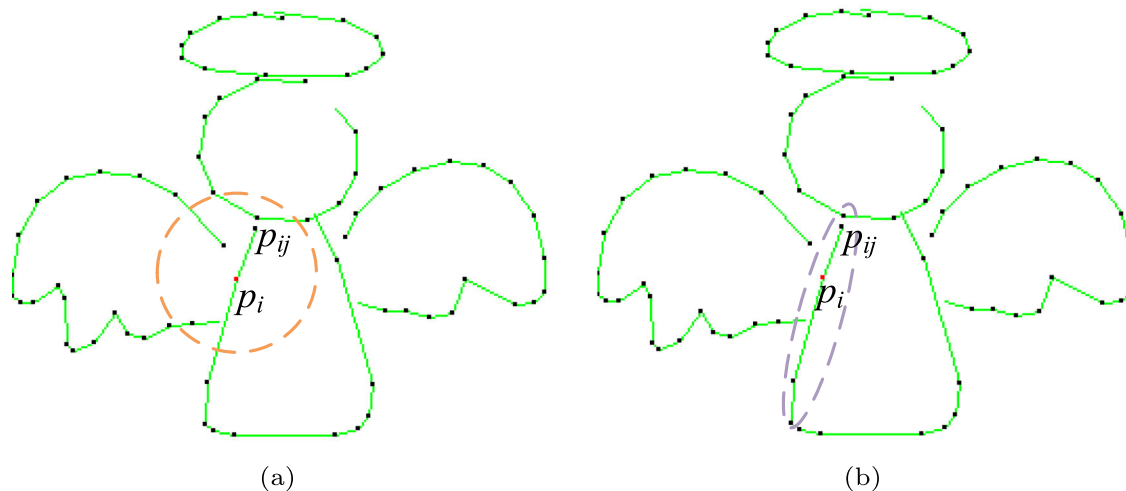


Fig. 4 The comparison between the common region perception procedure and the proposed region perception procedure. The black and red points are the sampling points, and the dotted lines are the perception regions. **a** shows a perception region using the common procedure, while **b** shows a perception region using the proposed

procedure, where $k = 4$. The common procedure perceives the surrounding region according to the two-dimensional spatial distribution around the center point. The proposed procedure perceives the surrounding region along the corresponding stroke of the center point

surrounding region $P_{ik}=\{p_{i1}, p_{i2}, \dots, p_{ik}\}$ of p_i is calculated by Eq. (4).

$$P_{ik} = \operatorname{argmin}_{p_{i1}, p_{i2}, \dots, p_{ik}} \sqrt{(x_i - x_{ij})^2 + (y_i - y_{ij})^2} \quad (4)$$

where argmin represents gaining the k points corresponding to the k shortest distances. Although the surrounding regions obtained by the above perception procedure conform to the two-dimensional spatial distributions of sketches, it ignores the stroke structures of sketches. Therefore, we propose to obtain the surrounding regions based on “stroke distance” combined with the K -nearest neighbor algorithm. Specifically, “stroke distance” is defined in Eq. (5).

$$P_{ik} = \operatorname{argmin}_{p_{i1}, p_{i2}, \dots, p_{ik}} \begin{cases} \sqrt{(x_i - x_{ij})^2 + (y_i - y_{ij})^2}, s_i = s_{ij} \\ m \times \sqrt{(x_i - x_{ij})^2 + (y_i - y_{ij})^2}, s_i \neq s_{ij} \end{cases} \quad (5)$$

where $p_i(x_i, y_i)$ is a center point, $p_{ij}(x_{ij}, y_{ij})$ is any point around p_i , s_i and s_{ij} represent the two strokes corresponding to p_i and p_{ij} , m is a scaling factor and m satisfies Eq. (6).

$$m \times \sqrt{(x_i - x_{ij})^2 + (y_i - y_{ij})^2} \geq \sqrt{w^2 + h^2} \quad (6)$$

where w and h are the width and height of the sketch. Equations (5) and (6) indicate that the “stroke distance” is equal to the traditional Euclidean distance when p_i and p_{ij} belong to the same stroke while the “stroke distance” is increased evidently when they do not belong to the same stroke. In other words, the NLFA module can perceive the

surrounding regions along the direction of a stroke, which makes our method consider the continuity of strokes and is conducive to perceive the stroke structure. The proposed perception procedure is shown in Fig. 4b. The NLFA module randomly selects either one of these two perception procedures, which leads to taking into account the two-dimensional spatial distributions and the continuity of strokes simultaneously.

To enlarge the perception region of each sampling point, we introduce multi-scale technology based on the idea of dilated convolution [38]. Assuming that any point p_i in the NLFA module has several perception regions with different dilated rates r , the perception region with r can be expressed as $\{p_{ij} \mid p_{ij} \in P_{ik \times (r+1)}, j \% (r+1) = 0\}$, that is, the perception region evenly obtains the first k points closest to the sampling point with the step of $r+1$. The realization of the above multi-scale perception regions is only suitable for testing, but the perception regions with different dilated rates r are more effective to obtain k surrounding points by a random selecting method during training. Figure 5 describes the perception regions using the common selecting method and the random selecting method.

In Fig. 5a, $\{p_{i1}, p_{i5}, p_{i9}\}$ are the selected points and the perception region is within the orange dotted line roughly. In Fig. 5b, three points are randomly selected from $\{p_{i1}, p_{i2}, \dots, p_{i9}\}$ and the perception region can cover the entire region within the purple dotted line roughly after multiple iterations.

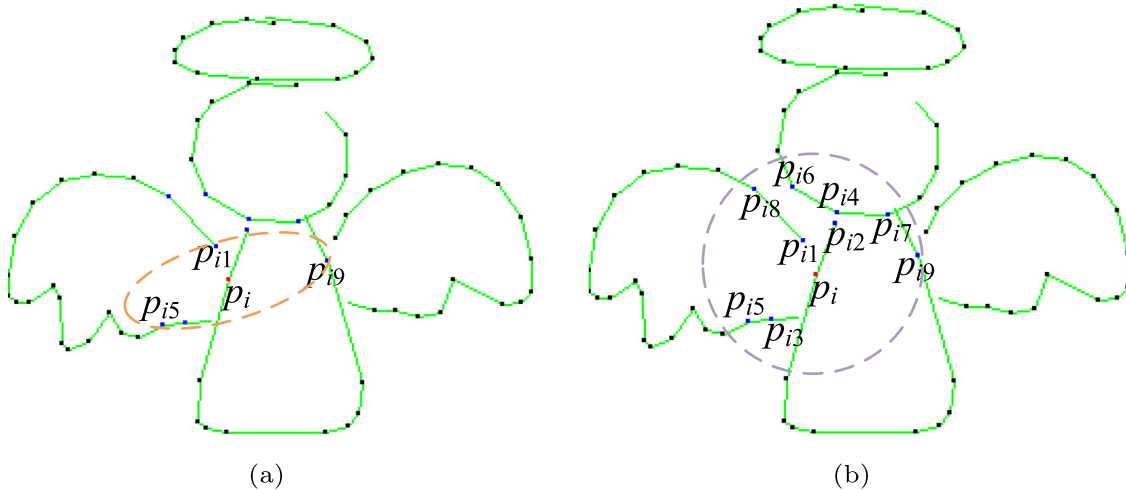


Fig. 5 The perception regions using different selecting methods. p_i is a center point, $\{p_{i1}, p_{i2}, \dots, p_{in}\}$ is the n points closest to p_i . For ease of understanding, n, k, r are set to 9, 3, 3 in order. **a** shows a perception

region using the common selecting method. **b** shows a perception region using the random selecting method. The perception region in **b** is much larger than the perception region in **a**

3.2.2 Feature calculation procedure

Feature calculation procedure is employed to handle with all kinds of sketch information. The common feature calculation procedures mainly collect the coordinates of points and the sketch features of the intermediate stages [24, 25]. However, these collected contents are not sufficient for a sketch, which greatly hinders the performance of sketch semantic segmentation. Obviously, the feature calculation procedure essentially plays the same role as the convolution kernel of CNN. However, due to the discreteness of the sampling points, the common feature calculation procedures discard some crucial elements when searching and determining the surrounding points. The distance information between the sampling points and their surrounding points, and the angle information are implicitly encoded using a traditional convolution kernel. Figure 6 illustrates the encoded contents. Thus, the feature calculation procedure of the NLFA module should encode the distance information and the angle information to ensure the integrity of the sketch features. Assuming that the absolute coordinate of the sampling point p_i is (x_i, y_i) , and the absolute coordinate of its any surrounding point p_{ij} is (x_{ij}, y_{ij}) . The distance $dist_{ij}$ between the sampling point p_i and p_{ij} can be expressed in Eq. (7).

$$dist_{ij} = \sqrt{(x_i - x_{ij})^2 + (y_i - y_{ij})^2} \tag{7}$$

Similarly, the angle information of p_i and p_{ij} is calculated by Eq. (8).

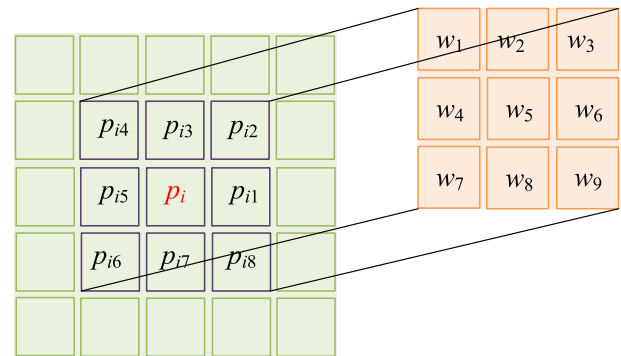


Fig. 6 The implicitly encoded contents using a traditional convolution kernel. The orange region is a traditional convolution kernel of 3×3 size, and $\{w_1, w_2, \dots, w_9\}$ are the weights of the kernel. The green region is a region of 5×5 size. It can be clearly observed that the distances between the center point p_i and $\{p_{i1}, p_{i2}, \dots, p_{i8}\}$ are $\{1, \sqrt{2}, 1, \sqrt{2}, 1, \sqrt{2}, 1, \sqrt{2}\}$, and the angles between p_i and $\{p_{i1}, p_{i2}, \dots, p_{i8}\}$ are $\{0^\circ, 45^\circ, \dots, 135^\circ\}$

$$\theta_{ij} = \arctan\left(\frac{y_{ij} - y_i}{x_{ij} - x_i}\right) \tag{8}$$

In Eqs. (7) and (8), the coordinates of all points are in the Cartesian coordinate system. The proposed feature calculation procedure ensures that the sketch features do not ignore the crucial context information, which is conducive to learn the complex geometric patterns. Further, the detailed feature calculation procedure of the NLFA module is expressed in Eq. (9).

$$F_{ij} = \delta\{\mathcal{M}\{p_i \oplus (p_{ij} - p_i) \oplus f_i \oplus (f_{ij} - f_i) \oplus dist_{ij} \oplus \theta_{ij} \oplus s_{ij}\}\} \tag{9}$$

where f_i and f_{ij} represent the intermediate-stage features, s_{ij} represents the stroke information, \oplus is a concatenation operation, \mathcal{M} is a multi-layer perceptron, and δ indicates the ReLU activation function. From Eq. (9), it can be concluded that the NLFA module spans rich context information. Compared with the common local feature aggregation modules, we supplement our module with the more informative contexts, involving the intermediate-stage features after normalization, distance information, angle information and stroke information according to the encoding idea of convolutional kernel, the basic properties of point format data, and the stroke structure of a sketch. And we define “stroke distance”, which enables the obtained local regions to extend along the stroke directions and better perceive the stroke structures.

3.3 Segment-level self-attention module

Supposing the inputs of the SLSA module are the sketch features with $[B, N, D]$ dimension, where B is batch size, N represents the number of the sampling points and D indicates the number of channels. We divide several points on the same stroke into one segment. Each sketch is divided into S segments, the initial feature dimension of each segment is $[B, S, D \times N/S]$. The new segment features with the dimension $[B, S, D']$ can be computed by feeding the initial segment features into a multi-layer perceptron. We aim to learn the relationship between segments for describing the internal structure of a sketch. As we all know, multi-head self-attention mechanism can be applied to calculate the relationship between any two objects and is widely utilized in the fields of natural language processing and computer vision. The impressive success of multi-head self-attention mechanism is originated from the correlation between any two objects and the long dependence between them. The multi-head self-attention mechanism makes neural networks pay attention to the characteristics of multiple subspaces at the same time. Inspired by the above observations, based on the idea of multi-head self-attention, we design a SLSA module to learn and strengthen the internal structure of a sketch through taking into account segment features and sketch characteristics. The specific calculation method is shown in Fig. 7.

In Fig. 7, given the segment features $f_s[S, D']$, the query vector $v_q[H, D''/H, S]$ and the key vector $v_k[H, D''/H, S]$ are obtained by Eqs. (10) and (11).

$$v_q = \mathcal{T}(\mathcal{M}(f_s)) \quad (10)$$

$$v_k = \mathcal{T}(\mathcal{M}(f_s)) \quad (11)$$

where \mathcal{M} stands for a multi-layer perception, its output dimension is D'' , and \mathcal{T} stands for matrix deformation.

Both v_q and v_k have H subspaces. Simultaneously, the position information of segments is processed by Eqs. (12) and (13).

$$e_{\text{stroke}} = \mathcal{T}(\mathcal{S}(p_{\text{stroke}}, d_{\text{stroke}})) \quad (12)$$

$$e_{\text{sketch}} = \mathcal{T}(\mathcal{S}(p_{\text{sketch}}, d_{\text{sketch}})) \quad (13)$$

where \mathcal{S} is a query procedure of the related embedding dictionary. p_{stroke} and p_{sketch} are two one-hot vectors, which represent the position of one segment in the corresponding stroke and that in the corresponding sketch, respectively. The embedding dictionaries d_{stroke} and d_{sketch} are queried according to p_{stroke} and p_{sketch} , then a deformation operation is performed on the query results to obtain the position embedding vectors e_{stroke} and e_{sketch} . The position information of a segment in the corresponding stroke p_{stroke} is local relative to the position information of the stroke in the corresponding sketch p_{sketch} . Thus, the existence of p_{stroke} and p_{sketch} enables the relationship between segments to capture the local position information in the corresponding stroke and the global position information in the corresponding sketch, which makes the relationship more accurate. Furthermore, we need to encode the position features and content features of segments, respectively. The relevant equations are expressed in Eqs. (14) and (15).

$$f_p = (e_{\text{stroke}} + e_{\text{sketch}}) \otimes v_q \quad (14)$$

$$f_c = \mathcal{T}(v_q) \otimes v_k \quad (15)$$

where \otimes represents a matrix multiplication, and $+$ is an element-wise addition. Finally, the calculation process of the SLSA module is expressed in Eq. (16).

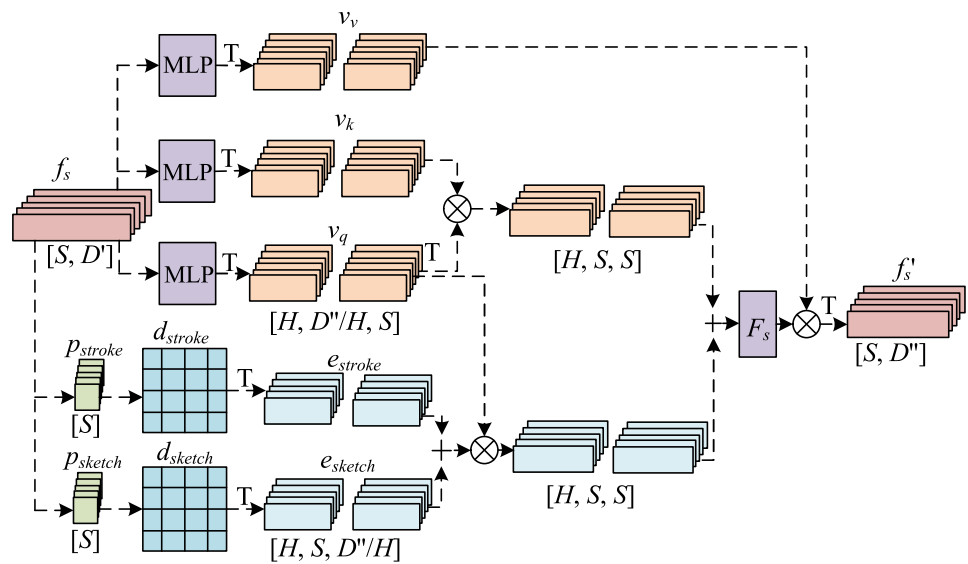
$$f'_s = \mathcal{T}(v_v \otimes \text{softmax}(f_p + f_c)) \quad (16)$$

where f'_s represents segment features after using the SLSA module. The SLSA module can focus on sketch features of multiple subspaces and calculate the correlation of any two segments through matrix operations and softmax function for establishing the relationship between segments. Then, the relationship is further enhanced by capturing crucial local and global position information of segments, which is better for learning the internal structure of a sketch more precisely. The SLSA module provides a basis for sketch semantic segmentation from a segment perspective.

3.4 Similar encoder–decoder structure

The proposed method is mainly realized by constructing a similar encoder–decoder structure. The structure has two significant characteristics. (1) It is a similar encoder–decoder structure rather than a real encoder–decoder structure. (2) It employs a global multi-scale mechanism, that is,

Fig. 7 The internal process of the SLSA module. For simplicity, we discard the parameter batch size B . MLP stands for a multi-layer perceptron, and F_s stands for a softmax function. T is a matrix deformation. d_{sketch} and d_{stroke} represent the embedding dictionaries



sketch features of several intermediate stages are aggregated to generate the final semantic segmentation map.

3.4.1 Similar encoder–decoder structure

Figure 2 illustrates this similar encoder–decoder structure, which is composed of nine stages. The first four stages are the encoding stages using the NLFA module, and the last five stages are the decoding stages using nonlinear transformation primarily. Compared with the common encoder–decoder structure, our structure does not perform the down-sampling and up-sampling operations on the sampling points. The number of the sampling points remains unchanged, which retains the features of the significant points and further ensures the performance of the proposed method. Similar to the common encoder–decoder structure, the number of feature channels in all stages changes from increasing to decreasing, which illustrates that we enrich deep features at the front stages and then label each point after decoding according to the deep features.

3.4.2 Global multi-scale mechanism

The NLFA modules of the encoding stages realize a multi-scale mechanism based on the idea of dilated convolution. However, the multi-scale mechanism is limited to local regions, and the global context information of different stages is ignored in the process of encoding and decoding features. From the global perspective, the lack of long dependence will lead to the feature differences of points from the same category. Consequently, we establish a global multi-scale mechanism to aggregate the outputs of four intermediate stages using skip connections. The four intermediate stages contain the last encoding stage and the

front three decoding stages, and the details are shown in Fig. 2. It can be observed from Fig. 2 that the features used to generate the final semantic segmentation map come from the features of four stages with different depths, which can describe multi-scale and global semantic information to enhance the capability for semantic analysis.

4 Experiments

4.1 Experimental settings

4.1.1 Dataset

We evaluate the proposed method on two widely used benchmarks named SPG [20] and SketchSeg-150K [21]. The sketches in SPG and SketchSeg-150K are created by unprofessional painters, which presents the practical significance. SPG dataset has a total of 16,000 sketches spanning 20 categories. SketchSeg-150K dataset has 150,000 sketches spanning 20 categories. Compared with SPG dataset, SketchSeg-150K has fewer semantic labels for each category.

4.1.2 Evaluation metrics

At present, almost all sketch semantic segmentation methods use P -metric and C -metric [26] as the standardized metrics. P -metric is a point-based metric that describes the proportion of the correctly marked points in sketches. C -metric is a component-based metric that describes the proportion of the correctly marked strokes in sketches. The two evaluation metrics are complementary and focus on strokes with different lengths.

4.1.3 Implementation details

As shown in Fig. 2, the model of the proposed method is mainly composed of four encoding stages and five decoding stages. In the encoding stages, there are four NLFA modules which utilize K -nearest neighbor algorithm and multi-layer perceptron. In the K -nearest neighbor algorithm, K is equal to 32. In the decoding stages, there are four multi-layer perceptrons and two sub-branches. The parameters of the above multi-layer perceptrons are listed in Table 2 where the two numbers in each bracket represent the input dimension and the output dimension.

Our method is implemented by Pytorch framework. We employ Adam algorithm to train the model of the proposed method for 100 epochs where the initial learning rate is 0.001 and the learning rate is updated to 0.65 times of the original learning rate every 10 epochs. The batch size is 20, every 2 points are divided into a segment on SPG, and every 4 points are divided into a segment on SketchSeg-150K. We compare the proposed method with the baseline methods on a server configured with GTX 1080 GPU and 32 G memory.

4.2 Comparative results and analysis

4.2.1 Segmentation accuracy and analysis

To evaluate the effectiveness of the proposed method, we conduct extensive experiments to compare the proposed method with the existing representative baseline methods. The baseline methods mainly fall into three categories. (1) The sequence-based methods include SPGSeg [20] and SketchSegNet+ [21]. (2) The image-based methods include FastSeg [22] and U-Net [34]. (3) The point-based methods include SketchGNN [10] and PointMLP [39]. U-Net and PointMLP are an image semantic segmentation method and a point cloud semantic segmentation method, respectively. Given their adaptability to sketch semantic segmentation, they are added to the comparative experiment. The original SketchGNN is trained for each category separately. For fairness, we use sketches with all categories to train SketchGNN together, which is same as other

baseline methods in the comparative experiment. The experimental results of all the methods on SPG are shown in Table 3. As can be observed, the proposed method achieves the average accuracy of 90.6% in P -metric, and the average accuracy of 86.6% in C -metric, which outperforms other baseline methods. Specifically, the proposed method exceeds the second best PointMLP by 3.8% in P -metric and 4.8% in C -metric, respectively. In addition to the average accuracy, Table 3 also shows the segmentation accuracy of each category. It can be observed that the proposed method can achieve the highest accuracy in almost all categories, which shows the high applicability to most categories of sketches. Besides, the proposed method has great advantages in the categories of alarm clock, ant, backpack, calculator crab and ice cream.

The experimental results of the baseline methods on SketchSeg-150K are shown in Table 4. It should be noted that the segmentation accuracy of SPGSeg is not shown in Table 4 because we did not find the original experimental results and its source code. As listed in Table 4, the proposed method obtains the average accuracy of 96.3% in P -metric and the average accuracy of 95.0% in C -metric, which outperforms all baseline methods. In terms of the average accuracy, the proposed method exceeds the second best FastSeg by 1.3% in P -metric, and 3.0% in C -metric, respectively. Compared with other baseline methods, the performance of the proposed method in C -metric is obviously superior to that in P -metric. This superiority is mainly rooted in the fact that the integration of strokes and the internal structures of sketches are adequately considered in the NLFA module and the SLSA module, respectively. From the above discussion, it can be concluded that the proposed method has great advantages no matter in P -metric or C -metric, which fully verifies the effectiveness.

4.2.2 Parameter size and calculation complexity

The parameter size and calculation complexity are also the important indicators to reflect the advantage and disadvantage of a method. The parameter sizes and calculation complexity (FLOPs) using different semantic segmentation methods are shown in Table 5. According to Table 5, the

Table 2 The parameters of all multi-layer perceptrons in the proposed method

	The encoding part		The decoding part	
	The first layer	The second layer	The first layer	The second layer
The first MLP	[7, 32]	[32, 32]	[384, 256]	[256, 256]
The second MLP	[135, 64]	[64, 64]	[384, 256]	[256, 256]
The third MLP	[263, 64]	[64, 64]	[320, 128]	[128, 128]
The forth MLP	[263, 128]	[128, 128]	[896, 128]	[128, 128]

Table 3 Semantic segmentation accuracy (%) on SPG dataset in *P*-metric (*P*) and *C*-metric (*C*)

Category	SPGSeg		SketchSegNet+		FastSeg		U-Net		SketchGNN		PointMLP		Our method	
	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>
Airplane	82.9	70.9	75.0	68.0	85.3	75.2	77.5	67.3	81.6	75.7	82.1	76.2	86.4	81.5
Clock	84.8	81.0	66.0	64.0	84.6	72.3	80.2	72.3	85.9	84.6	86.6	82.6	90.3	86.3
Ambulance	80.7	68.1	71.0	69.0	85.8	75.3	81.1	75.6	88.8	83.3	83.0	78.5	87.2	81.9
Ant	66.4	56.6	72.0	70.0	68.9	66.4	69.9	66.8	77.3	75.9	75.7	70.3	85.1	80.8
Apple	89.9	71.8	88.0	82.0	91.4	82.3	75.9	63.3	89.2	81.4	88.4	71.1	92.7	87.0
Backpack	75.2	63.7	62.0	60.0	73.3	59.8	65.5	44.1	73.8	64.7	72.7	63.1	82.3	73.3
Basket	84.8	83.2	73.0	75.0	86.6	82.2	82.1	71.8	79.5	82.8	83.5	80.3	87.9	85.5
Butterfly	89.0	83.6	86.0	78.0	92.7	79.3	87.5	79.9	93.5	89.6	91.2	85.8	95.2	91.9
Cactus	77.5	72.3	83.0	80.0	73.3	68.6	81.0	77.8	83.0	82.8	84.8	80.5	89.9	86.1
Calculator	91.1	89.9	91.0	92.0	97.4	93.0	89.3	86.8	91.4	89.7	94.1	93.0	97.8	96.7
Campfire	92.3	91.4	83.0	78.0	95.6	92.9	82.5	75.0	90.8	84.6	89.7	86.7	94.7	91.7
Candle	88.3	71.8	81.0	74.0	90.8	80.1	89.5	86.5	96.9	94.3	94.1	86.0	96.4	93.9
Coffee cup	92.0	87.2	72.0	72.0	90.9	87.0	82.2	75.1	88.7	86.5	93.3	92.0	92.2	89.1
Crab	77.9	70.5	77.0	70.0	75.9	55.4	83.0	77.3	83.7	80.1	84.6	77.0	88.6	83.1
Duck	86.9	75.4	74.0	69.0	88.9	75.1	81.7	77.5	92.2	87.9	91.8	90.2	90.2	85.8
Face	88.0	80.1	75.0	66.0	88.1	80.4	81.9	78.0	90.3	84.2	89.1	86.2	89.9	87.0
Ice cream	85.4	79.3	80.0	75.0	87.5	80.1	83.3	79.7	84.0	81.9	85.7	81.7	93.4	90.8
Pig	81.9	75.4	70.0	69.0	81.1	73.9	85.1	81.6	82.6	77.1	83.7	82.7	88.2	83.5
Pineapple	89.8	90.2	77.0	75.0	91.9	82.3	79.0	68.9	82.9	82.5	89.4	84.3	92.1	87.2
Suitcase	92.7	90.7	75.0	69.0	94.8	86.7	76.0	68.0	94.1	90.8	92.6	89.4	91.9	88.7
Average	84.9	77.6	77.0	73.0	86.2	77.4	80.7	73.7	86.5	83.0	86.8	81.8	90.6	86.6

Bold values indicate the highest accuracy in *P*-metric or *C*-metric

proposed method has a parameter size of 0.7 M, and its computational complexity is about 0.3 G in FLOPs, which fully demonstrates that the proposed method has the advantages of the high average accuracy, low calculation complexity and small parameter size. The sketch applications mainly occur on portable devices such as mobile phones and tablets. The parameter size less than 1 M is very friendly to be applied on the devices.

4.3 Qualitative results and analysis

We will display intuitively some visual semantic segmentation results in the qualitative experiment. Three open source methods (U-Net, PointMLP, SketchGNN) and the proposed method are taken part in the experiment. The specific segmentation results are shown in Fig. 8. Compared with other methods, the semantic segmentation results of our method are closer to the level of humans in most cases. Results of the traditional image semantic segmentation method U-Net are unsatisfactory, because stroke information is not considered in the encoding process. PointMLP ignores the relationship between segments, resulting in the inferior performance. The original SketchGNN is trained for each category. Thus, SketchGNN

loses desired results in this qualitative experiment. Our method captures rich sketch information and then determines the semantic segmentation results of sketches from a point-level perspective and a segment-level perspective, which brings the great performance improvement.

4.4 Ablation analysis

The proposed method mainly involves two crucial modules named a NLFA module and a SLSA module. Thus, we evaluate the impact of the NLFA module and the SLSA module on sketch semantic segmentation results. Both SPG and SketchSeg-150K are important semantic segmentation datasets. There are 3–7 labels per category in SPG, while 2–4 labels per category in SketchSeg-150K. Therefore, the evaluation of the proposed method on SPG is representative and presents more challenges. In ablation analysis, we conduct relevant experiments on SPG dataset.

4.4.1 The study of the novel local feature aggregation module

The NLFA module encodes a large number of useful information related to sketch characteristics. Theoretically,

Table 4 Semantic segmentation accuracy (%) on SketchSeg-150K dataset in *P*-metric (*P*) and *C*-metric (*C*)

Category	SketchSegNet+		FastSeg		U-Net		SketchGNN		PointMLP		Our method	
	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>
Angel	89.0	86.0	98.0	96.0	94.8	91.9	82.5	82.7	94.6	93.4	98.4	97.4
Bird	98.0	97.0	82.0	70.0	86.5	64.7	74.1	59.3	88.6	78.4	98.3	97.7
Bowtie	99.0	100.0	100.0	100.0	98.3	99.3	99.6	98.9	98.8	99.9	100.0	100.0
Butterfly	95.0	95.0	98.0	96.0	93.3	88.5	43.5	39.2	96.5	94.4	99.8	99.7
Candle	83.0	69.0	96.0	78.0	58.4	53.0	69.2	53.5	75.0	69.6	85.9	79.8
Cup	77.0	74.0	91.0	92.0	84.1	82.0	95.3	94.6	89.9	89.4	95.4	94.6
Door	99.0	99.0	100.0	100.0	97.6	90.4	99.6	99.2	99.9	99.9	100.0	100.0
Dumbbell	99.0	99.0	98.0	98.0	98.8	97.4	99.0	98.4	98.7	99.3	99.8	99.8
Envelope	100.0	99.0	100.0	100.0	89.7	56.4	91.1	92.3	99.1	100.0	100.0	100.0
Face	94.0	91.0	98.0	95.0	95.0	72.8	98.4	94.4	98.2	94.7	99.4	98.3
Ice	72.0	69.0	100.0	100.0	81.6	81.0	99.0	98.4	94.2	95.2	98.3	98.8
Lamp	95.0	94.0	78.0	78.0	79.8	72.8	55.1	46.9	85.9	79.8	95.2	93.2
Lighter	99.0	98.0	99.0	96.0	89.0	76.7	77.4	64.6	96.2	91.5	99.6	99.6
Marker	61.0	55.0	90.0	80.0	54.5	39.6	51.6	55.2	55.8	49.6	80.4	71.9
Mushroom	70.0	66.0	98.0	94.0	79.4	64.0	91.3	90.8	92.9	89.2	93.3	91.1
Pear	99.0	98.0	97.0	94.0	96.9	87.7	22.2	39.2	99.1	98.3	100.0	100.0
Plane	86.0	85.0	100.0	99.0	90.4	82.4	98.1	97.7	93.4	91.0	99.2	99.1
Spoon	85.0	81.0	80.0	79.0	67.3	45.3	46.1	49.4	79.8	68.0	86.4	84.3
Traffic	96.0	96.0	89.0	93.0	74.7	76.0	49.8	74.5	90.9	90.6	96.6	96.7
Van	87.0	84.0	99.0	99.0	91.5	75.3	94.8	97.1	94.3	91.8	98.8	98.4
Average	89.0	87.0	95.0	92.0	85.5	75.0	78.8	78.0	91.2	88.4	96.3	95.0

Bold values indicate the highest accuracy in *P*-metric or *C*-metric

Table 5 The parameter sizes and computational complexity using different semantic segmentation methods

	SPGSeg	FastSeg	U-Net	SketchGNN	PointMLP	Our method
Parameters (M)	23.4	40.9	17.3	0.4	16.8	0.7
FLOPs (M)	–	–	40522.7	43.6	5413.6	321.56

there are three kinds of information that are important factors for the proposed method to achieve high segmentation accuracy, including distance information, angle information and stroke information. In order to evaluate the important impact of these information on the performance, we build three kinds of local feature aggregation modules for the proposed method. Besides, to verify the benefits of “stroke distance”, we also build two kinds of local feature aggregation modules. Five methods using different local feature aggregation modules are as follows:

- (1) *NLFA w/o distance* This method is based on the local feature aggregation module that encodes all information in Eq. (9) except the distances between the sampling points and the surrounding points.
- (2) *NLFA w/o angle* This method is based on the local feature aggregation module that encodes all

information in Eq. (9) except the angles between the sampling points and the surrounding points.

- (3) *NLFA w/o stroke* This method is based on the local feature aggregation module that encodes all information in Eq. (9) except the corresponding stroke ID.
- (4) *NLFA w/o SD* This method is based on the local feature aggregation module that encodes all information in Eq. (9) and only utilizes Euclidean distance.
- (5) *NLFA* The proposed method is based on the local feature aggregation module that encodes all information in Eq. (9).

The above methods will use Euclidean distance or “stroke distance” randomly unless noted otherwise. Table 6 shows semantic segmentation results of these five methods on SPG.

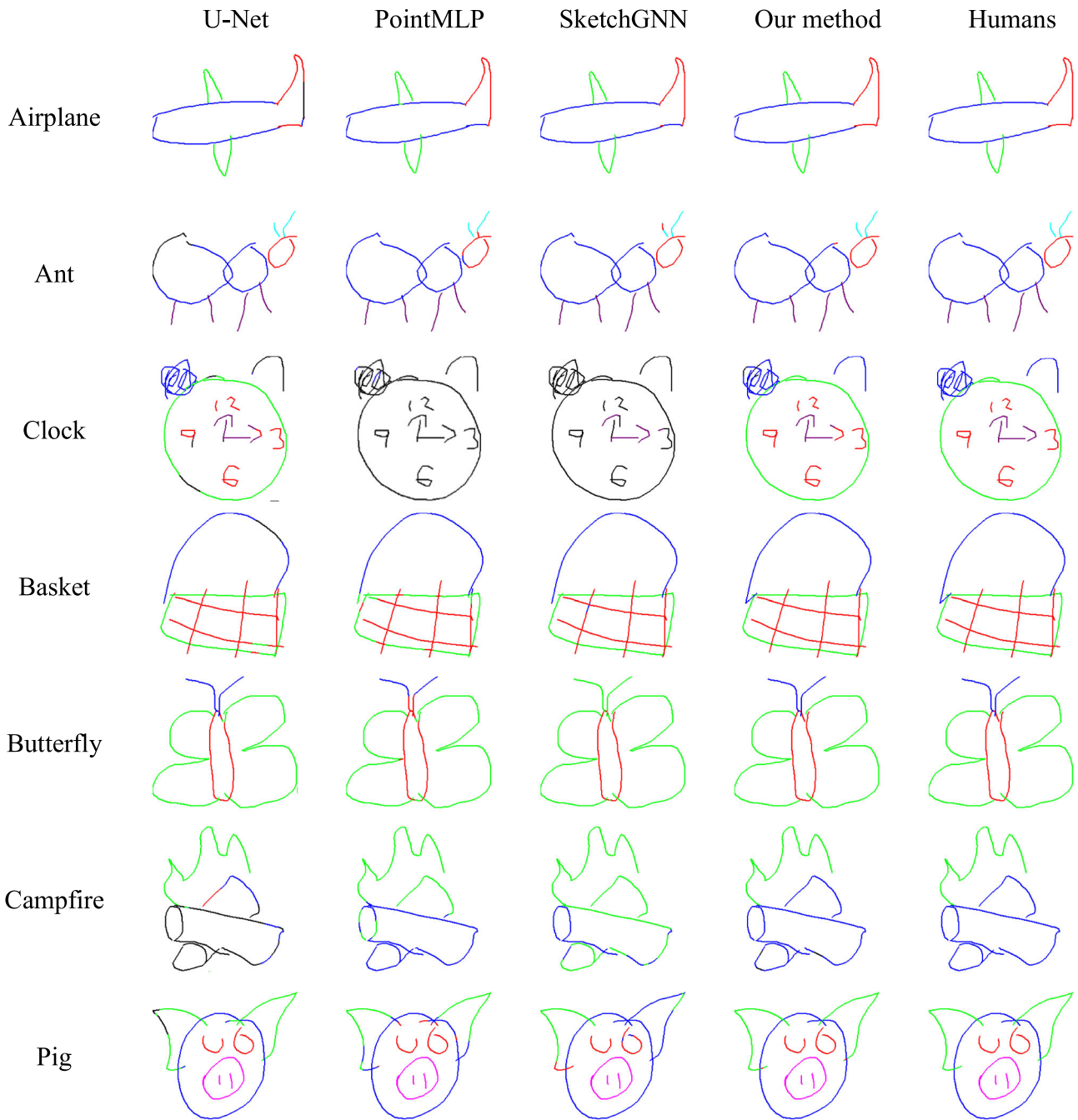


Fig. 8 Some visual sketch semantic segmentation results. Different colors represent different semantics

As demonstrated in Table 6, distance information, angle information and stroke information improve the average accuracy of the proposed method by 0.6%, 0.9% and 4.3% in *P*-metric, respectively, while they improve the average accuracy by 0.8%, 1.5% and 6.1% in *C*-metric, respectively. It can be concluded that these three kinds of information are of great help to enhance the performance, and stroke information plays the most important role in the sketch semantic segmentation task. Stroke information

represents the internal structure of a sketch partly, which has a great impact on semantic analysis, so the performance gap between with and without stroke information can be as high as 6.1% in *C*-metric. In addition, according to Table 6, “stroke distance” improves the average accuracy by 0.9% in *P*-metric, and 1.3% in *C*-metric, respectively. Results illustrate that the NLFA module based on “stroke distance” is more effective for the performance enhancement.

Table 6 Semantic segmentation accuracy (%) of the five methods on SPG dataset in *P*-metric (*P*) and *C*-metric (*C*)

Category	NLFA w/o distance		NLFA w/o angle		NLFA w/o stroke		NLFA w/o SD		NLFA	
	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>
Airplane	86.3	80.1	88.3	82.7	82.2	74.1	83.7	78.0	86.4	81.5
Clock	88.1	81.5	89.4	83.7	87.8	81.7	87.2	81.8	90.3	86.3
Ambulance	86.3	80.3	86.5	81.2	82.2	76.1	85.0	78.0	87.2	81.9
Ant	83.7	79.8	85.8	80.6	76.1	68.5	82.9	76.8	85.1	80.8
Apple	91.7	85.8	90.6	81.6	88.5	77.1	92.3	84.6	92.7	87.0
Backpack	84.0	75.7	78.5	69.5	72.0	57.4	81.0	70.7	82.3	73.3
Basket	85.9	81.8	85.0	80.6	79.4	71.1	86.7	83.5	87.9	85.5
Butterfly	94.9	92.5	94.5	91.8	91.1	86.8	94.7	91.6	95.2	91.9
Cactus	90.3	87.2	90.8	87.6	85.5	78.9	90.2	87.1	89.9	86.1
Calculator	97.4	96.7	96.9	95.2	94.2	92.9	97.2	96.1	97.8	96.7
Campfire	94.1	92.1	93.8	91.4	88.4	83.6	94.8	93.2	94.7	91.7
Candle	95.9	93.7	95.9	93.3	95.5	93.2	95.4	92.9	96.4	93.9
Coffee cup	90.9	88.2	92.2	88.3	90.1	87.6	91.8	88.9	92.2	89.1
Crab	87.8	82.3	88.6	83.4	84.0	79.0	89.7	83.9	88.6	83.1
Duck	89.0	84.4	90.2	86.5	89.4	86.2	89.4	85.2	90.2	85.8
Face	90.5	87.5	89.0	84.5	89.3	85.2	89.3	84.7	89.9	87.0
Ice cream	93.5	91.2	92.9	90.3	87.1	81.9	92.4	90.4	93.4	90.8
Pig	88.0	84.1	86.4	82.8	85.1	82.1	88.2	85.5	88.2	83.5
Pineapple	90.4	83.5	88.9	81.3	88.0	80.8	91.7	85.8	92.1	87.2
Suitcase	92.5	88.8	89.5	85.4	90.4	86.9	91.5	88.0	91.9	88.7
Average	90.0	85.8	89.7	85.1	86.3	80.5	89.7	85.3	90.6	86.6

Bold values indicate the highest accuracy in *P*-metric or *C*-metric

4.4.2 The study of the segment-level self-attention module

The SLSA module is combined with a cross-entropy loss $L_{\text{segment-level}}$ to establish and strengthen the relationship between segments, so as to improve the understanding for sketch semantics. In order to evaluate the importance of the proposed module, we will test the proposed method without and with the SLSA module, respectively. Table 7 shows the semantic segmentation results on SPG dataset.

As shown in Table 7, the SLSA module can enhance the performance of sketch semantic segmentation. The reason for this result is that the SLSA module establishes the segment-to-segment relationship and provides a new segment-level perspective to label each point in a sketch. In Table 7, the SLSA module improves the average accuracy by 1.2% in *P*-metric, and 1.8% in *C*-metric, respectively. The positive effect of the SLSA module in *C*-metric is better than that in *P*-metric slightly.

4.4.3 The impact of the number of points in each segment on the performance

The number of points in each segment can affect the performance of the proposed method. Therefore, we study the cases that each segment contains 2, 4, 8, 16 and 32 points,

respectively. Figure 9 shows the average accuracy using 2, 4, 8, 16 and 32 points on SPG and SketchSeg-150K. It can be observed that the impact of the number of points is limited when the number of points is between 2 and 16, but the average accuracy will decline sharply when the number of points reaches 32. This result illustrates that more points in each segment cannot obtain the better performance. The reason for this result is that the points from different strokes will divide into the same segment when each segment contains too many points, which has an adverse effect on establishing the relationship between segments. In fact, the segments composed of 2 points on SPG can obtain the highest average accuracy of 90.6% in *P*-metric and 86.6% in *C*-metric, respectively, while the segments composed of 4 points on SketchSeg-150K can obtain the highest average accuracy of 96.3% in *P*-metric and 95.0% in *C*-metric, respectively.

4.4.4 The impact of the number of sampling points on the performance

The number of sampling points is an important hyper-parameter, which can directly affect the final segmentation result. Theoretically, the reasonable number of sampling points can retain the crucial details of a sketch and keep

Table 7 Semantic segmentation accuracy (%) without and with the SLSA module on SPG dataset in *P*-metric (*P*) and *C*-metric (*C*)

Category	Without SLSA module		With SLSA module	
	<i>P</i>	<i>C</i>	<i>P</i>	<i>C</i>
Airplane	84.4	77.1	86.4	81.5
Alarm clock	89.2	85.7	90.3	86.3
Ambulance	86.2	81.0	87.2	81.9
Ant	80.6	76.1	85.1	80.8
Apple	92.0	85.0	92.7	87.0
Backpack	81.6	70.3	82.3	73.3
Basket	84.7	80.4	87.9	85.5
Butterfly	94.4	90.9	95.2	91.9
Cactus	90.7	87.7	89.9	86.1
Calculator	97.5	96.5	97.8	96.7
Campfire	93.7	90.4	94.7	91.7
Candle	97.0	95.6	96.4	93.9
Coffee cup	92.4	89.4	92.2	89.1
Crab	86.5	79.7	88.6	83.1
Duck	90.0	85.5	90.2	85.8
Face	89.3	85.4	89.9	87.0
Ice cream	92.6	88.6	93.4	90.8
Pig	86.1	82.7	88.2	83.5
Pineapple	91.0	84.1	92.1	87.2
Suitcase	88.3	84.0	91.9	88.7
Average	89.4	84.8	90.6	86.6

Bold values indicate the highest accuracy in *P*-metric or *C*-metric

low computational complexity. Based on this principle, we choose 256 sampling points. Table 8 shows the impact of the number of sampling points on the segmentation accuracy. Generally speaking, within a certain range, the greater *N* value is set, the more points can be captured by the receptive field, so the proposed method presents the higher segmentation accuracy. This is the reason why the performance of sketch semantic segmentation will

Table 8 The impact of the number of sampling points on the segmentation accuracy (%)

	<i>N</i> = 64	<i>N</i> = 128	<i>N</i> = 256	<i>N</i> = 512
<i>P</i> -metric	88.6	89.8	90.6	89.2
<i>C</i> -metric	84.0	85.4	86.6	84.8

gradually improve as *N* increases in Table 8 when *N* is less than 512. When *N*=512, due to the excessive number of points, the region obtained under the fixed receptive field is relatively small compared with the whole sketch, so the captured information tends to be local and the global perception is unsatisfactory. Due to the lack of global perception, the performance of sketch semantic segmentation is degraded. In fact, when we increase the receptive field and set the *K* value to 48, *P*-metric and *C*-metric is 90.5% and 86.9%, respectively, which is basically the same as the performance using *N* = 256. Therefore, we do not continue to increase the number of *N* points, because too large *N* value will lead to a decrease in computing efficiency and segmentation accuracy simultaneously.

4.5 A simple application

Sketch semantic segmentation is the understanding about sketch semantics. Therefore, theoretically, sketch semantic segmentation can improve the performance of sketch recognition. In this section, we show the application of sketch semantic segmentation to the sketch recognition task. The sketch recognition network we used is Sketch-a-Net [40], which is representative and widely used in the sketch field. The sketch recognition method combined with the proposed sketch semantic segmentation method can be divided into four stages. (1) In the training phase, Sketch-a-Net is used to train on the SPG dataset [20] to obtain a

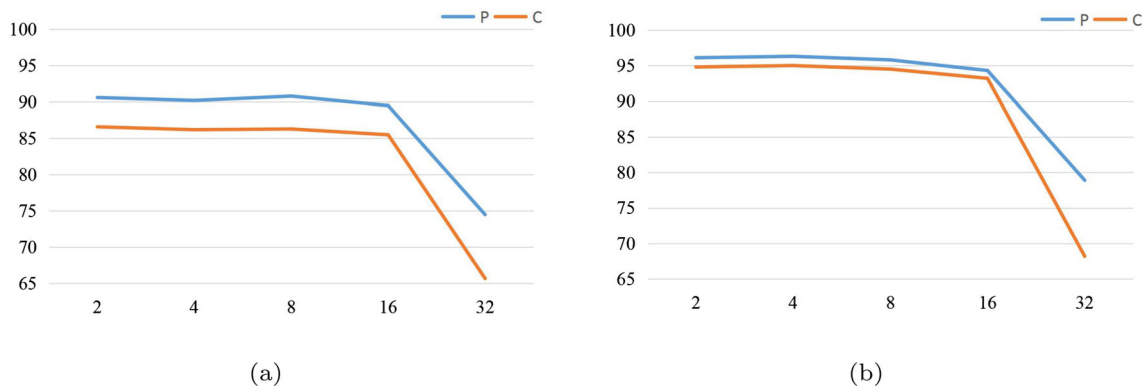


Fig. 9 The impact of the number of points in each segment on the average accuracy. The horizontal axes represent the number of points, and the vertical axes represent the average accuracy (%). **a** is the average accuracy on SPG, and **b** is the average accuracy on SketchSeg-150K

Table 9 Comparison of the accuracy (%) of two sketch recognition methods

Method name	Accuracy (%)
Sketch-a-Net	85.7
Sketch-a-Net + SG	94.1

standard sketch recognition model. (2) In the segmentation stage, the proposed sketch semantic segmentation method is used to obtain the semantic parts of the input sketch. Certainly, the noise semantics will be removed. (3) The stage of reassigning values for the neurons in the last fully connected layer. Each neuron corresponds to a sketch class, and we take the neuron with the largest score. If the sketch class corresponding to the neuron does not contain any segmented semantic part, the score of the neuron will be reduced to the lowest score. Repeat step (3) until the neuron with the largest score contains a segmented semantic part. (4) In the prediction phase, a sketch class is obtained through the neuron with the largest score in the last fully connected layer. We show the recognition accuracy of Sketch-a-Net and Sketch-a-Net combined with the proposed sketch semantic segmentation method (Sketch-a-Net + SG) in Table 9. It can be seen from Table 9 that the performance of Sketch-a-Net is significantly improved. Specifically, the recognition accuracy is improved by 8.4%. In fact, the performance improvement comes from the revision of predicted sketch classes by sketch semantics. The original Sketch-a-Net does not check the semantics of the candidate sketch classes, while Sketch-a-Net + SG automatically eliminates the sketch classes with incorrect semantics, thus avoiding unnecessary errors. Although Sketch-a-Net + SG is more complex than the original Sketch-a-Net, it can achieve significant performance improvement. Figure 10 shows the qualitative results of

Sketch-a-Net and Sketch-a-Net + SG. In Fig. 10, the first line shows the wrong results of Sketch-a-Net, while the second line shows the wrong results of Sketch-a-Net + SG. Obviously, compared with Sketch-a-Net, the wrong results of Sketch-a-Net + SG is easier to understand because it is difficult for humans to recognize sketches of the second line.

5 Discussion

In this paper, we propose a sketch semantic segmentation method using novel local feature aggregation and segment-level self-attention, which has great advantages in segmentation accuracy, parameter size and computational complexity. The proposed method captures the internal structures of sketches by encoding informative point-level features, perceiving the stroke structures, and building the relationship between segments, thus obtaining the state-of-the-art semantic segmentation accuracy. The input of the proposed method is a point set and a segment set. The MLP for processing these data is much less than the convolutional neural network for processing images in terms of parameter size and computational complexity. This advantage is more conducive to the application of the proposed method to portable devices. Besides, the proposed method aims at tackling sketch semantic segmentation task. Although it cannot directly be used for sketch instance-level or sketch scene segmentation tasks, it can be extended for sketch instance-level or sketch scene segmentation tasks by adding a post-processing step based on clustering algorithm or metric learning [41]. Simultaneously, our approach exists some limitations. (1) We divide n points into a segment. The value of n is determined by a verification set, and some segments may not conform to the stroke structure. (2) The proposed method still has unsatisfactory segmentation results on some sketches as shown

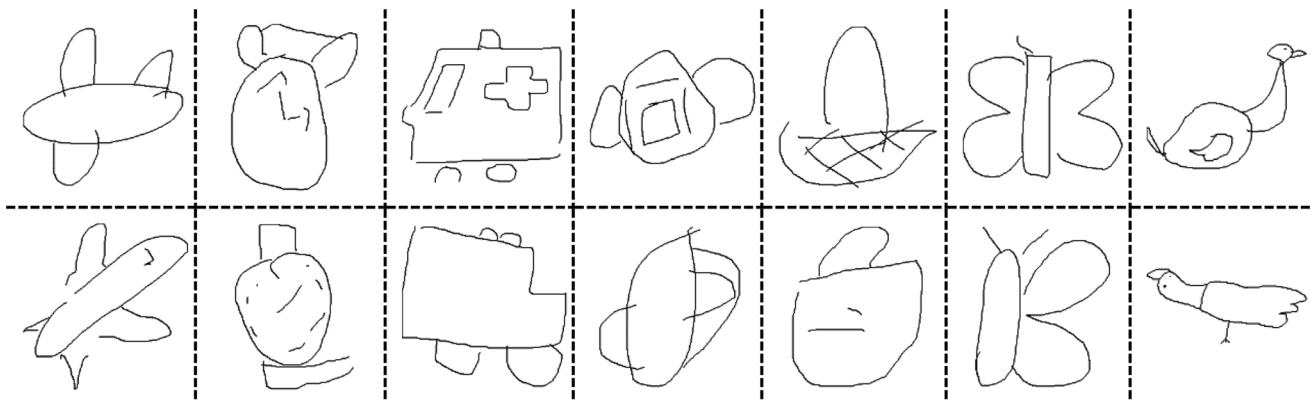
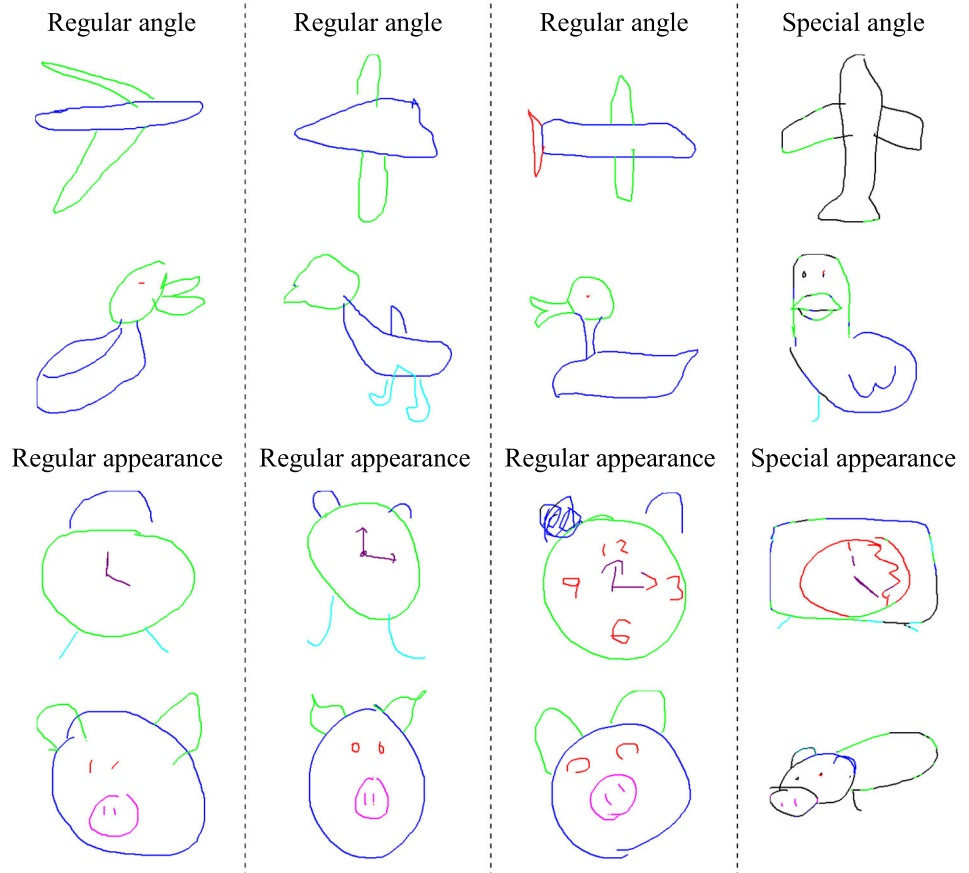


Fig. 10 The qualitative results of Sketch-a-Net and Sketch-a-Net + SG. The sketches of columns 1 to 7 are as follows: airplane, alarm clock, ambulance, backpack, basket, butterfly and duck, respectively

Fig. 11 Some unsatisfactory results on special sketches. Sketches with special angles and special appearances show unsatisfactory segmentation results



in Fig. 11, including sketches with special angles, sketches with large differences in appearance from the same category. We deem that the above issues can be solved by better representations [42] and GAN-based methods [43].

6 Conclusion

We propose a sketch semantic segmentation method, which is realized by a NLFA module, a SLSA module and a similar encoder–decoder structure based on these two modules. The NLFA module uses Euclidean distance or “stroke distance” randomly when the sampling points search the surrounding regions, which makes the perception regions conform to the two-dimensional spatial distributions and the stroke structures of sketches. And the NLFA module encodes rich information such as distance information, angle information and stroke information, which enhances the semantic discrimination ability of sketch features. Simultaneously, the SLSA module is designed to enhance the relationship between segments and to learn the internal structures of sketches. Finally, the similar encoder–decoder structure integrates the global features of several intermediate stages to enrich semantic

contents. Extensive experiments show that the proposed method achieves state-of-the-art performance in both *P*-metric and *C*-metric. In the future, we will explore the unsupervised sketch semantic segmentation method, which is more in line with the actual application when a sketch lacks a large number of semantic labels.

Acknowledgements We are very grateful to the editor and reviewers for their time and efforts while reviewing this manuscript. Besides, we also appreciate the support of the Central Government Guided Local Funds for Science and Technology Development (No. 216Z0301G), the National Natural Science Foundation of China (No. 61379065) and the Natural Science Foundation of Hebei Province in China (No. F2019203285).

Data availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest No potential conflict of interest was reported by the authors.

References

- Li L, Zou CQ, Zheng YY et al (2021) Sketch-R2CNN: an RNN-rasterization-CNN architecture for vector sketch recognition. *IEEE Trans Vis Comput Graph* 27(9):3745–3754
- Wan J, Zhang KH, Li HD et al (2021) Angular-driven feedback restoration networks for imperfect sketch recognition. *IEEE Trans Image Process* 30:5085–5095
- Lin H, Fu Y, Jiang Y G et al (2020) Sketch-BERT: learning sketch bidirectional encoder representation from transformers by self-supervised learning of sketch gestalt. In: *IEEE conference on computer vision and pattern recognition*. IEEE Computer Society, pp 6757–6766
- Zhang XL, Shen ML, Xue M et al (2022) A deformable CNN-based triplet model for fine-grained sketch-based image retrieval. *Pattern Recognit* 125:108508
- Chen YD, Zhang ZL, Wang YF et al (2022) AE-Net: fine-grained sketch-based image retrieval via attention-enhanced network. *Pattern Recognit* 122:108291
- Bhunia AK, Chowdhury PN, Sain A et al (2021) More photos are all you need: semi-supervised learning for fine-grained sketch based image retrieval. In: *IEEE computer society conference on computer vision and pattern recognition*. IEEE Computer Society, pp 4245–4254
- Gryaditskaya YL, Song JF, Yang YX et al (2021) Toward fine-grained sketch-based 3d shape retrieval. *IEEE Trans Image Process* 30:8595–8606
- He X, Zhou Y, Zhou Z et al (2018) Triplet-center loss for multi-view 3d object retrieval. In: *IEEE conference on computer vision and pattern recognition*. IEEE Computer Society, pp 1945–1954
- Ge C, Sun HF, Song YZ et al (2022) Exploring local detail perception for scene sketch semantic segmentation. *IEEE Trans Image Process* 31:1447–1461
- Yang LM, Zhuang JJ, Fu HB et al (2021) SketchGNN: semantic sketch segmentation with graph neural networks. *ACM Trans Graph* 40(3):1–13
- Sarvadevatla RK, Dwivedi I, Biswas A et al (2017) Sketch-parse: towards rich descriptions for poorly drawn sketches using multi-task hierarchical deep networks. In: *ACM international conference on multimedia*. Association for Computing Machinery, pp 10–18
- Zhu MR, Li J, Wang NN et al (2021) Learning deep patch representation for probabilistic graphical model-based face sketch synthesis. *Int J Comput Vision* 129(6):1820–1836
- Willis KD, Jayaraman PK, Lambourne JG et al (2021) Engineering sketch generation for computer-aided design. In: *IEEE computer society conference on computer vision and pattern recognition workshops*. IEEE Computer Society, pp 2105–2114
- Xu BX, Chang W, Sheffer A et al (2014) True2Form: 3D curve networks from 2D sketches via selective regularization. *ACM Trans Graph* 33(4):1–13
- Xu K, Chen K et al (2013) Sketch2scene: sketch-based co-retrieval and co-placement of 3d models. *ACM Trans Graph* 32(4):123:1–123:15
- Pu JT, Gur D (2009) Automated freehand sketch segmentation using radial basis functions. *CAD Comput Aided Des* 41(12):857–864
- Sun ZB, Wang CH, Zhang LQ et al (2012) Free hand-drawn sketch segmentation. *European conference on computer vision*. Springer, New York, pp 626–639
- Schneider RG, Tuytelaars T (2016) Example-based sketch segmentation and labeling using CRFs. *ACM Trans Graph* 35(5):1–9
- Wu XY, Qi YG, Liu J et al (2018) Sketchsegnet: aRNN model for labeling sketch strokes. In: *IEEE international workshop on machine learning for signal processing*. IEEE Computer Society, pp 1–6
- Li K, Pang KY, Song YZ et al (2019) Towards deep universal sketch perceptual grouper. *IEEE Trans Image Process* 28(7):3219–3231
- Qi YG, Tan ZH (2019) SketchSegNet+: an end-to-end learning of RNN for multi-class sketch semantic segmentation. *IEEE Access* 7:102717–102726
- Li L, Fu HB, Tai CL (2019) Fast sketch segmentation and labeling with deep learning. *IEEE Comput Graph Appl* 39(2):38–51
- Zhu XY, Xiao Y, Zheng Y (2020) 2D freehand sketch labeling using CNN and CRF. *Multimed Tools Appl* 79(3):1–18
- Wang F, Lin SJ, Li HH et al (2020) Multi-column point-CNN for sketch segmentation. *Neurocomputing* 392:50–59
- Wang F, Lin S, Wu H et al (2019) SPFusionNet: sketch segmentation using multi-modal data fusion. In: *IEEE international conference on multimedia and expo*. IEEE Computer Society, pp 1654–1659
- Huang Z, Fu HB, Lau RWH et al (2014) Data-driven segmentation and labeling of freehand sketches. *ACM Trans Graph* 33(6):1–10
- Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. *IEEE Trans Pattern Anal Mach Intell* 39(4):640–651
- Badrinarayanan V, Kendall A, Cipolla R (2017) SegNet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans Pattern Anal Mach Intell* 39(12):2481–2495
- Chen LC, Papandreou G, Kokkinos I et al (2015) Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: *International conference on learning representations, ICLR*
- Chen LC, Papandreou G, Kokkinos I et al (2018) DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans Pattern Anal Mach Intell* 40(4):834–848
- Chen LC, Papandreou G, Schroff F et al (2017) Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*
- Chen LC, Zhu Y, Papandreou G et al (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. *European conference on computer vision*. Springer, New York, pp 833–851
- Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. *International conference on medical image computing and computer-assisted intervention*. Springer Verlag, New York, pp 234–241
- Oktay O, Schlemper J, Folgoc L L et al (2018) Attention U-Net: learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*
- Zhou Z, Siddiquee M, Tajbakhsh N et al (2018) U-Net++: a nested U-Net architecture for medical image segmentation. *Lect Notes Comput Sci* 11045:3–11
- Alom MZ, Hasan M, Yakopcic C et al (2018) Recurrent residual convolutional neural network based on U-Net (R2U-Net) for medical image segmentation. *arXiv preprint arXiv:1802.06955*
- Zhang X, Xu HM, Mo H et al (2021) DCNAs: Densely connected neural architecture search for semantic image segmentation. In: *IEEE conference on computer vision and pattern recognition*. IEEE Computer Society, pp 13951–13962
- Yu F, Koltun V (2016) Multi-scale context aggregation by dilated convolutions. In: *International conference on learning representations, ICLR*
- Ma X, Qin C, You H X et al (2022) Rethinking network design and local geometry in point cloud: a simple residual MLP framework. *arXiv preprint arXiv:2202.07123*

40. Yu Q, Yang Y, Liu F et al (2017) Sketch-a-Net: a deep neural network that beats humans. *Int J Comput Vision* 122(3):411–425
41. Weinberger KQ, Saul LK (2009) Distance metric learning for large margin nearest neighbor classification. *J Mach Learn Res* 10:207–244
42. Stefano Z, Shabab B, Stefan H et al (2022) PolyWorld: polygonal building extraction with graph neural networks in satellite images. In: *IEEE/CVF conference on computer vision and pattern recognition*, IEEE
43. Odena A, Olah C, Shlens J (2017) Conditional image synthesis with auxiliary classifier gans. In: *International conference on machine learning*. IMLS, pp 4043–4055

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.