ABSTRACT This paper presents CLIP Driven Few-shot Panoptic Segmentation (CLIP-FPS), a novel few-shot panoptic segmentation model that leverages the knowledge of Contrastive Language-Image Pre-training (CLIP) model. The proposed method builds upon a center indexing attention mechanism to facilitate knowledge transfer, which entails representing objects in an image as centers along with their pixel offsets. The model comprises a decoder responsible for generating object center-offset groups and a self-attention module tasked with producing a feature attention map. Subsequently, the object centers index the map to acquire the corresponding embeddings, paving the way for matrix multiplication and SoftMax operation to facilitate text embedding matching and the computation of the final panoptic segmentation masks. Quantitative evaluation on datasets such as COCO and Cityscapes shows that our method outperforms existing panoptic segmentation techniques in terms of Panoptic Quality (PQ) metrics.

INDEX TERMS Panoptic segmentation, CLIP, cityscapes, convolutional neural network, image processing.

I. INTRODUCTION
Panoptic segmentation is a comprehensive technique in the field of computer vision, which amalgamates object detection, instance segmentation, and semantic segmentation in order to allocate a unique semantic label or instance ID to individual pixels in an image [1]. Conventional panoptic segmentation methods mainly rely on fixed labels, usually ranging from tens to hundreds, depending on the number of categories in the dataset. Nevertheless, the natural language employed to depict images in a photograph frequently comprises thousands of nouns [2]. Consequently, a more advanced model is essential to accurately correlate objects within images to their respective natural language descriptions.

In recent years, Contrastive Language-Image Pre-training (CLIP) [3], has emerged as a high-performance vision-language pretraining model, showcasing notable enhancements in many multi-modal tasks. These tasks include image-text retrieval [3], video-text retrieval [4], [5], and text referred semantic segmentation [6]. Basically, CLIP extracts textual features and visual features from an extensive dataset, comprising over 400 million text-image pairs. Utilizing contrastive learning, CLIP embeds similar objects closely within the feature space, ensuring the cosine distances of embeddings for related objects are near, while maintaining distinct distances for dissimilar objects. Correspondingly, CLIP strives to align text embeddings and image embeddings to the nearest domain, thus resulting in a minimal cosine distance between embeddings extracted from the text encoder and image encoder of CLIP for equivalent objects.

In this paper, we introduce a novel method called CLIP-driven Few-shot Panoptic Segmentation, hereafter abbreviated as CLIP-FPS. This method utilizes the powerful capabilities of the CLIP model for few-shot panoptic segmentation, so that panoptic segmentation of images can be performed based on natural language style text instructions. Previous attempts have been made to transfer the knowledge of CLIP for image segmentation tasks, such as Wang et al.’s CRIS [7], which focuses on referring image segmentation. Nevertheless, it is limited in its capacity to only segment a single object in an image, leading to inadequate segmentation accuracy. Boyi et al. [8] propose another approach, centered on using CLIP for few-shot semantic segmentation.
FIGURE 1. Illustration of the proposed CLIP-FPS architecture, which can be divided into four components: (a) image and text feature extraction, (b) panoptic segmentation centers and offsets, (c) center indexing attention, and (d) image-text embeddings matching. Specifically, the image and text feature extraction part extracts image features as well as text features; the panoptic segmentation centers and offsets part performs the segmentation procedure; the center indexing attention part indexes the centers to compute dot product attention; and the image-text embeddings matching part matches the embedding of the image and the text to make the final prediction.

However, this method encounters difficulties in distinguishing between objects with identical class labels within the image. To the best of our knowledge, no prior research has combined panoptic segmentation with CLIP, making this an exceptionally challenging task.

The crux of this problem lies in obtaining suitable feature embeddings for each object in the image. CLIP models typically generate a single embedding for the entire image, which necessitates the creation of a suitable architecture for extracting distinct feature embeddings for different objects. This architecture must simultaneously yield high-quality panoptic segmentation results and maintain accurate image-text multimodal correspondence.

In the proposed CLIP-FPS, a novel center indexing attention mechanism has been developed to address the challenges raised earlier. Building upon the principles of Panoptic DeepLab [9], the mechanism enables the representation of objects in an image as centers with corresponding pixel offsets, which point towards these centers. The CLIP image encoder serves as the backbone for this approach, facilitating the regression of object centers and their associated pixel offsets. Figure 1 demonstrates the design of the mechanism, which integrates the image encoder and text encoder elements of the CLIP model for effective knowledge transfer. Decoders are then implemented to produce object center-offset groups, while a self-attention module generates a feature attention map. Subsequently, this feature attention map is indexed by the object centers to obtain their respective embeddings. Matrix multiplication, followed by a SoftMax operation, is employed to align the text embedding with the object embeddings. Ultimately, the panoptic segmentation masks are computed based on the centers and offsets, effectively highlighting objects referenced in the text within the resulting images.

In this study, we conducted the quantitative evaluation to measure the performance of our proposed method CLIP-FPS. To facilitate this, we designed a dataset processing method that provides panoptic segmentation datasets such as COCO [10] and Cityscapes [11] with natural language descriptions. The new natural language text descriptions allow us to make better use of labels of these datasets and then engage in few-shot fine-tuning. We compared the performance of our proposed method to the foremost panoptic segmentation techniques, and the results showed that our method outperformed them in terms of Panoptic Quality (PQ) metrics [1].

The primary contributions of this paper include:
- The proposal of a novel center-indexing attention mechanism that transfers the knowledge of the CLIP model to the task of panoptic segmentation.
- The introduction of a dataset processing approach that converts panoptic segmentation datasets into image-text multimodal datasets, and the public accessibility of the generated dataset.
- The achievement of state-of-the-art performance through the joint training of the CLIP backbone and the panoptic segmentation head using our proposed CLIP-FPS method.

II. RELATED WORKS
A. OVERVIEW OF PANOPTIC SEGMENTATION
Panoptic segmentation, as described by the literature [1], is a method that combines the tasks of semantic segmentation and instance segmentation. These tasks each exhibit distinct characteristics and methods, which contribute to the overall efficacy of panoptic segmentation. Semantic segmentation primarily focuses on enhancing the perception of obscured features. Strategies for improvement include
spatial pyramid pooling at various grid scales, as demonstrated by Zhao et al. [12], which results in increased precision. Additionally, Chen et al. [13] and Dai et al. [14] expand the perceptual fields for convolutional kernels by incorporating dilated convolution and deformable kernels. Instance segmentation, on the other hand, extends the capabilities of Mask R-CNN [15] to achieve better performance. Bolya et al. [16] proposed YOLACT, a method that refines object masks by cropping and assembling them, thereby improving the Intersection over Union (IoU) performance. Huang et al. [17] suggested the inclusion of an extra mask IoU for loss calculation, which further enhances the IoU performance.

The primary goal of panoptic segmentation is to address both semantic and instance segmentation tasks using a unified neural network. Most cutting-edge panoptic segmentation techniques employ a two-stage network, as they necessitate an additional stage for generating proposals based on Mask R-CNN [15]. Examples of these two-stage networks include TASCNet [18], AUNet [19], Panoptic FPN [20], UPSNet [21], and AdaptIS [22], which create instance masks by modifying Mask R-CNN and incorporating lightweight stuff segmentation branches for the remaining regions. These networks also utilize post-processing techniques to resolve mask overlaps. In contrast, some panoptic segmentation methods are proposal-free, such as DeeperLab [23], SSAP [24], and Panoptic DeepLab [9]. Notably, the majority of existing segmentation models are limited by a fixed label set, as determined by the labels present in the datasets [8]. To address this limitation, researchers have begun to explore few-shot segmentation approaches, which provide a small number of annotated examples of novel classes during testing.

B. ATTENTION

Attention mechanisms have gained significant traction in the realm of deep learning models, now recognized as a powerful tool for various natural language processing tasks. They have the ability to capture contextual keywords [25], [26] and have proven to be invaluable in numerous applications. The Transformer [26] model proposed in the attention mechanism paper has been developed extensively. Based on the Transformer model, many well-known and powerful models have been developed in the fields of natural language processing and computer vision, such as BERT [27], DALL-E [28], GPT series [29], [30], etc. CLIP [3] model is also one of them, leading the text-visual multimodal fusion. In the field of multimodal tasks, word attention is employed to adjust the importance of different areas in images, which in turn aids in generating image captions [31], answering questions about images [32], and reference image segmentation [33]. Attention mechanisms are implemented for visual grounding, as the varied attention of queries, images, and objects are individually computed and cyclically accumulated [34]. Self-attention [26], a technique where words can attend to all other words within an input sequence, has proven to be a useful method for learning relationships and enhancing machine translation performance. This approach has also been applied in video processing tasks to capture long-term dependencies across temporal frames [35].

In this paper, we propose a novel cross-modal self-attention module that connects image attention derived from object center features with language features. Our method uses the core design of the attention mechanism to extract object-by-object visual features from the CLIP output visual features of the whole image, which facilitates subsequent segmentation tasks and visual text matching. This innovative approach aims to further enhance the performance of multimodal tasks and deepen our understanding of attention mechanisms in natural language processing.

C. CLIP

Contrastive Language Image Pretraining, has emerged as a significant domain of research in recent times. With rapid advancements and impressive accomplishments in various multimodal downstream tasks, it has garnered considerable attention. Generally, this field encompasses visual question answering [36], image captioning [5], and image-text retrieval. Radford et al. [3] introduced CLIP as a means to illustrate that traditional non-text associated recognition tasks could substantially benefit from the combined training of image-text multimodal processing. Benefiting from large-scale image and text pairs collected from the Internet, CLIP has achieved remarkable success in aligning two modality representations within the embedding space. The model employs contrastive learning, using a large-scale language model and a visual feature encoder, to capture compelling visual concepts for zero-shot image classification.

In recent years, numerous tasks [8], [37], [38], [39] have been developed to transfer the knowledge gained from the CLIP model to downstream applications, yielding promising results in areas such as video captioning, video text search, and image generation. ViLd [38] presents an advanced few-shot object detection method that leverages CLIP, while MDETR [40] proposes an end-to-end approach for modulating transformer-based baseline detectors with text features derived from state-of-the-art language models. Similar to CLIP, these studies demonstrate that the robustness and generality of object detection models can be significantly enhanced with language support. Inspired by these methods, our work represents the first attempt to combine a panoptic segmentation model with a large-scale language model, paving the way for further advancements in this field.

III. METHODOLOGY

This section details the different modules of the proposed model. First, the encoding modules are employed to extract image and text features. Then, the center and offset decoding modules are used to predict the panoptic segmentation results. Next, the attention mechanism is employed to calculate the visual feature of each object. Finally, the cosine distance between the text and the visual features is calculated to match the text to the object in the image.
A. IMAGE AND TEXT FEATURE EXTRACTION

The proposed method utilizes the knowledge transfer from the CLIP model to extract image and text feature embeddings for further processing. CLIP can employ a ResNet50 [41] and a Transformer [26] to extract visual and text embeddings, respectively. By training an unsupervised contrastive learning model on 400 million image-text pairs, CLIP embeds similar objects close in the feature space and aligns text embeddings with image embeddings. The cosine distances of similar textual descriptions are close, while the cosine distances of different textual descriptions are further apart. Additionally, the embeddings extracted from the text and image encoders have a near cosine distance. Thus, this approach is ideal for the panoptic segmentation task, as the image and text encoders are straightforwardly employed without the necessity of the image-text contrastive comparison module. Note that the ResNet version CLIP Image Encoder is employed in the proposed architecture because the mainstream segmentation networks [12], [13], [14], [42] employ the ResNet series modules as their backbones. By employing the ResNet version CLIP, the proposed method can take advantage of the existing high-performance modules for the semantic segmentation subtask.

The input of the pipeline is an image, denoted as \( I \in \mathbb{R}^{H \times W \times 3} \), and a text containing natural language descriptions about the image, denoted as \( T \in \mathbb{R}^{L} \). The image \( I \) is then input into the CLIP ResNet50 Image Encoder, from which features are collected from stages 2, 3, and 4 of the ResNet50 and denoted as \( F_{S2}, F_{S3}, F_{S5} \). By upsampling and merging these features, we can obtain the semantic segmentation result and instance features. Based on the instance features, we can further produce the offset map and the center map. The center map predicts the probability of each pixel being an object center; The offset map predicts the relative location of each pixel to its center. The instance segmentation result can be obtained by calculating and shifting the centers and offsets. Finally, we can get the panoptic segmentation result by fusing the semantic and instance segmentation result.

B. PANOPTIC SEGMENTATION CENTERS AND OFFSETS

In order to generate panoptic segmentation masks from the visual and text features retrieved from the CLIP image and text encoders, we use two decoders: a semantic segmentation decoder and an instance segmentation decoder, as illustrated in the upper path of Figure 2. These decoders are employed to decode the image features. To accomplish this, the DeeperLab [23] model has been modified to incorporate the Atrous Spatial Pyramid Pooling (ASPP) [42] in order to generate the general feature \( F_{S5} \in \mathbb{R}^{H \times W \times C_5} \) from the last feature map \( F_{S4} \) of the image encoder by atrous convolution. Afterwards, the feature \( F_{S5} \) is upsampled and concatenated with the feature \( F_{S4} \), and then convolved with a \( 1 \times 1 \) convolutional layer to obtain upsampled feature \( F_{up4} \in \mathbb{R}^{H \times W \times C_5} \). Similarly, features \( F_{up4} \) and \( F_{S3} \) are fused to produce the upsampled feature \( F_{up3} \). This process is repeated until \( F_{S2} \), resulting in \( F_{up2} \in \mathbb{R}^{H \times W \times C_2} \). Finally, the feature \( F_{up2} \) is upsampled and argmaxed as the semantic segmentation mask \( M_{sem} \in \mathbb{R}^{H \times W \times N} \), with \( N \) denoting the number of classes of the panoptic segmentation dataset.

To generate the instance segmentation mask \( M_{ID} \), we design an instance decoder to produce centers and offsets, thereby calculating the object masks. As shown in the lower path of Figure 2, the basic architecture of the instance decoder is the same as the semantic segmentation decoder. However, the decoders contain different model parameters, so that they can generate the two predictions: a center map \( M_{center} \) and an offset map \( M_{offset} \). The center map \( M_{center} \in \mathbb{R}^{H \times W \times N} \) regresses the object mass centers of the instance masks, The offset map \( M_{offset} \in \mathbb{R}^{H \times W \times N} \) regresses the relative offset coordinates from each pixel to its corresponding center.
By filtering the centers from the center map \( M_{center} \) and calculating the corresponding offset pixels from the offset map \( M_{offset} \), it’s easy to obtain the instance segmentation mask \( M_{ID} \). Note that values in the center map \( M_{center} \) indicate the probability that a particular pixel is the center of a given mask.

To obtain the center predictions, a keypoint-based non-maximum suppression (NMS) [43] is applied on the center map \( M_{center} \). Those predictions with a confidence score below a threshold of 0.1 are then filtered out, leaving the top-K (K = 100) highest confidence scores as the eligible centers, denoted as \( \{ C_k : (i_k, j_k) \} \in \mathbb{R}^{K \times 2} \), where \( (i_k, j_k) \) refer to the coordinates of the center points in the center map. To determine the mask ID for each pixel, each pixel value of the offset map \( M_{offset} \) is added with its coordinate value \((i, j)\), and then calculate the distances of these values with the K center coordinate value. Pick up the closest center \( C_k \), thus the mask ID of that pixel is the index of center \( C_k \). This process can be succinctly expressed as calculating the mask ID map \( M_{ID} \) at each pixel \((i, j)\):

\[
M_{ID} (i, j) = \arg \min_k \| (i, j) + M_{offset} (i, j) - C_k \|^2 \tag{1}
\]

By applying this method, we are able to obtain the semantic segmentation mask \( M_{sem} \), the object centers map \( M_{center} \), the offset value map \( M_{offset} \), and the instance segmentation mask \( M_{ID} \), whose id falls in \([0, K]\) with \( K \) referring to the number of detected objects. The id value of the semantic segmentation id map \( M_{sem} \) falls in \([0, N]\) with \( N \) referring to the number of the stuff classes. When fused with the \( M_{sem} \), the instance segmentation ID map can be converted into the panoptic segmentation ID map \( M_{pano} \), whose id falls in \([0, N + K]\). Note that the instance segmentation IDs are added with \( N \) during the fusing process.

### C. CENTER INDEXING ATTENTION MECHANISM

The key problem now is how to generate visual embeddings of each object in the image given the object centers and corresponding segmentation masks. To address this problem, we propose a center indexing attention mechanism. It starts with applying \( 1 \times 1 \) convolutions and Multi-Head Self-Attention (MHSA) to the general feature \( F_{S5} \). MHSA consists of three linear mapping layers that transform \( F_{S5} \) into intermediate representations of queries \( Q \in \mathbb{R}^{P \times d} \), keys \( Y \in \mathbb{R}^{P \times d} \), and values \( V \in \mathbb{R}^{P \times d} \). The attention feature map \( F_{attn} \in \mathbb{R}^{P \times d} \) is given by

\[
F_{attn} = \text{softmax} \left( \frac{QY^T}{\sqrt{d}} \right) V \tag{2}
\]

where \( P = \frac{H}{64} \times \frac{W}{64} \) denotes the pixel number of feature \( F_{S5} \), and \( d \) denotes the dimension of the feature embedding. By reshaping \( F_{attn} \) into a size of \( \frac{H}{64} \times \frac{W}{64} \) and upsampling it via the “nearest” method, we obtain the new attention feature map \( F_{attn} \in \mathbb{R}^{H \times W \times d} \).

After that, it’s easy to obtain the feature embeddings for the objects in the image. By indexing the attention feature map \( F_{attn} \) with the filtered object centers \( \{ C_k : (i_k, j_k) \} \in \mathbb{R}^{K \times 2} \), the feature embeddings of the objects show up as \( F_{center} \in \mathbb{R}^{K \times d} \). Note that the feature embeddings \( F_{center} \) only contain instance segmentation masks. Furthermore, a linear layer and a global pooling module are employed on \( F_{attn} \) to produce \( N \)-class semantic segmentation attention feature embeddings \( F_{sem} \in \mathbb{R}^{N \times d} \). For convenience, the embeddings \( F_{center} \) and \( F_{sem} \) are concatenated into unified panoptic segmentation feature embeddings \( F_{pano} \in \mathbb{R}^{(K+N) \times d} \).

### D. IMAGE AND TEXT EMBEDDINGS MATCHING

With the availability of the text feature embeddings \( F_{text} \in \mathbb{R}^{L \times d} \) and the panoptic segmentation feature embeddings \( F_{pano} \in \mathbb{R}^{(K+N) \times d} \), the matching between these image and text embeddings is required. To indicate which panoptic segmentation mask refers to which natural language text, their embeddings can be correlated by the inner product to obtain the visual-language matching matrix

\[
M_{match} = F_{pano} \cdot F_{text}^T \tag{3}
\]

where \( M_{match} \in \mathbb{R}^{(K+N) \times L} \). By using SoftMax and argmax operations on the matching matrix \( M_{match} \), we can easily obtain the best language referring panoptic segmentation results.

### E. JOINTLY TRAINING CLIP AND PANOPTIC SEGMENTATION MODEL

As the proposed network is a combination of CLIP backbone and panoptic segmentation network, we need to jointly train the whole network. The proposed method is trained with a ResNet-50 backbone and a ViT-B/32 text encoder for 200 epochs with a base learning rate of 0.005 and weight decay of \( 5e^{-4} \). For the CLIP image and text encoder modules, the learning rate is reduced by multiplying 0.1 with the standard learning rate, because only fine-tune is required for the CLIP encoders in this panoptic segmentation task.

For the loss function, we calculate the sum and average over the losses from different parts of the network. These losses include semantic segmentation loss, center loss, offset loss, and the CLIP matching matrix loss:

The semantic segmentation loss \( L_1 \) employs the cross-entropy loss over the semantic segmentation map \( M_{sem} \) and the semantic segmentation target label \( T_{sem} \):

\[
L_1 = \text{CrossEntropy} \left( M_{sem}, T_{sem} \right) \tag{4}
\]

The center loss \( L_2 \) employs the Mean-Square (MSE) loss over the object centers map \( M_{center} \) and the corresponding center target label \( T_{center} \):

\[
L_2 = \text{MSE} \left( M_{center}, T_{center} \right) \tag{5}
\]

The offset loss \( L_3 \) employs the L1 loss over the object pixel offset map \( M_{offset} \) and the corresponding offset target label \( T_{offset} \):

\[
L_3 = | M_{offset} - T_{offset} | \tag{6}
\]
The CLIP matching matrix loss \( L_4 \) employs the cross-entropy loss over the visual-language matching matrix map \( M_{match} \) and the corresponding object text pair target \( T_{text} \):

\[
L_4 = \text{CrossEntropy}(M_{match}, T_{text})
\]

Eventually, we can obtain the final loss \( Loss \) by averaging the above losses

\[
Loss = 0.25 \times (L_1 + L_2 + L_3 + L_4)
\]

IV. EXPERIMENTAL RESULTS

This section begins by introducing the standard metrics used to evaluate the experiments, followed by a brief description of the dataset used. Quantitative comparison and evaluation of the proposed architectural components are then presented, with the concluding visualization of panoptic segmentation evaluation for the dataset

A. EXPERIMENTAL DATASETS AND EVALUATION METRICS

The experiments of this paper used the public panoptic segmentation datasets COCO [10] and Cityscapes [11] as testbeds. The COCO dataset contained 118,000 images for training purposes, as well as 2,000 images for validation and 5,000 for testing, categorizing scenes into 28 ‘stuff’ and 37 ‘things’ classes. Meanwhile, the Cityscapes dataset was an urban street scenes dataset, focusing on a semantic understanding of common driving scenarios, with scenes recorded in different seasons and from over 50 European cities. This dataset was particularly challenging due to its diversity as well as the large number of dynamic objects, such as pedestrians and cyclists. These dynamic objects often could be close to each other or partially occluded. Such close distance makes the task of panoptic segmentation, particularly with regard to the ‘thing’ classes, more challenging.

We evaluate the effectiveness of the proposed CLIP-FPS via comparison with other well-known architectures, using the Panoptic Quality (PQ) metric as the evaluation criterion [1]. PQ is a composite score combining the segmentation coverage rate and the instance-level segmentation accuracy, which is expressed as:

\[
PQ = \frac{\sum_{(p,g) \in TP} \text{IoU} (p, g)}{|TP| + \frac{1}{2} |FP| + \frac{1}{2} |FN|}
\]

where \( p \) and \( g \) refer to the predicted mask region and the ground-truth mask region, respectively. \( TP, FP, FN \), and \( \text{IoU} \) are true positives, false positives, false negatives, and intersection-over-union, respectively. The performance of the predicted mask region is computed as \( \text{IoU} = TP / (TP + FP + FN) \). We also report Semantic Quality (SQ) and Recognition Quality (RQ) metrics, calculated as

\[
SQ = \frac{\sum_{(p,g) \in TP} \text{IoU} (p, g)}{|TP|}
\]

\[
RQ = \frac{|TP|}{|TP| + \frac{1}{2} |FP| + \frac{1}{2} |FN|}
\]

B. CONVERT PANOPTIC SEGMENTATION DATASETS INTO MULTIMODAL DATASETS

As discussed in Section III-E, we present a dataset process method to convert panoptic segmentation datasets, such as COCO and Cityscapes, into image-text multimodal datasets. The primary concept, illustrated in Figure 3, involves designing a text prompt for each object present in the dataset image. Initially, the input image is uniformly partitioned into a \( 3 \times 3 \) grid, encompassing top left, top, top right, left, central, right, bottom left, bottom, and bottom right sections. Subsequently, objects situated in distinct areas are allocated specific prompt labels, such as “the person at the top right of the image.” Two essential pieces of information are included in the prompt: the category name and the object location, both of which can be acquired from the original labels of the panoptic segmentation dataset. The category name pertains to the semantic labels assigned to each object, while the object location is computed based on the instance masks. By averaging the pixel locations of the object, we obtain the object center location. The location prompt is then assigned based on the area in which each object center resides. This approach allows the efficient conversion of panoptic segmentation datasets into image-text multimodal datasets, facilitating further research and development in this domain. The multimodal dataset generated through the aforementioned processing can be obtained via the link\(^1\) to assist researchers in conducting further training and scientific research.

\(^1\)https://drive.google.com/drive/folders/11OJQxxuTfuY_KPcqsIghNRF1NHyK

FIGURE 3. The illustration of the text prompt generation process on the COCO dataset. The input image is evenly divided into 3 x 3 areas. The objects in the different area will be assigned different prompt labels according to their category and center location.
C. PANOPTIC SEGMENTATION COMPARISON ON COCO DATASET

In this subsection, we evaluate the panoptic segmentation performance of our proposed method, CLIP-FPS, using the well-known COCO dataset. To provide a fair comparison with existing mainstream methods, we have selected several prominent works, including AUNet [19], Panoptic-FPN [20], AdaptIS [22], UPSNet [21], Detectron2 [43], SSAP [24], and Panoptic DeepLab [9]. It is important to note that the experiment settings are intentionally kept basic, with data augmentation techniques such as flipping, multiscale input, or extra coarse labeled data being disabled. This approach allows for a direct and unbiased assessment of the bare models’ performance.

As shown in TABLE 1, our proposed method, CLIP-FPS, demonstrates superior performance on the COCO validation dataset in comparison to the selected mainstream methods. Specifically, CLIP-FPS outperforms the state-of-the-art method, Detectron2, with a PQ score that is 0.3% higher. Furthermore, our method achieves the highest SQ score, registering an improvement of 0.3% over UPSNet, and a competitive RQ of 45.5%, which is only 0.7% lower than Detectron2’s RQ of 46.2%. The effectiveness of the CLIP-FPS is further corroborated through the visualization results presented in Figure 4, which depict the panoptic segmentation on the COCO validation dataset.

D. PANOPTIC SEGMENTATION COMPARISON ON CITYSCAPES DATASET

In this subsection, we assess the panoptic segmentation performance of the proposed CLIP-FPS method on the Cityscapes dataset by conducting a direct comparison with other notable panoptic segmentation approaches, such as TASCNet [18] and Seamless [44]. These results, presented in TABLE 2, are derived from a base model configuration of each respective network, without the use of any augmentation techniques such as flipping, multiscale input, or extra coarse labeled data.

TABLE 2 displays the outcomes on the Cityscapes validation dataset, where our proposed CLIP-FPS method achieves
the highest performance, outperforming the current state-of-the-art Panoptic DeepLab by 1.2% for the PQ metric and 1.3% for mIoU. Moreover, our method exhibits superior results across all four metrics, including PQ, SQ, RQ, and mIoU. This not only signifies a robust segmentation accuracy for SQ but also indicates a high recall rate for RQ on the Cityscapes dataset. Figure 5 provides a visual representation of our CLIP-FPS approach on the Cityscapes validation dataset, further showcasing the impressive performance of our method in the context of few-shot panoptic segmentation.

**E. MULTIMODAL PERFORMANCE AND RESULTS**

In addition to the quantitative assessment of panoptic segmentation efficacy, this study also presents a qualitative analysis concerning image-text multimodal performance. The successful integration of the robust CLIP backbone and the versatile panoptic segmentation framework can be attributed to the meticulous design process. This allows for not only the accurate association of textual descriptions with their respective objects in the image, but also the precise masking of the highlighted objects. Consequently, the proposed network is capable of identifying and emphasizing the contours of desired objects when provided with a natural language description of the input image’s content.

The intricate procedure of object-text multiplication is grounded in the multiplication matrix outlined in Section III-E. In this context, \( M_{\text{match}} \in \mathbb{R}^{(K+N) \times L} \) represents the object-text multiplication matrix, where \( K \) signifies the
quantity of objects present in the image, $N$ corresponds to the number of stuff classes, and $L$ denotes the pieces number of the text descriptions. By employing the SoftMax operation on the multiplication matrix along dimension 0, the object ID that best fits the required text descriptions can be effortlessly obtained. Subsequently, the object ID can be used to index the panoptic segmentation prediction outcomes $M_{pano}$, which consists of the mask map of the required description text when the pixels in $M_{pano}$ are equal to the object ID. This methodology serves as a reliable and effective approach to object-text matching, contributing to the overall accuracy and efficiency of the process.

In this study, we demonstrate the visualization of the multimodal performance of our proposed CLIP-FPS approach on the COCO validation dataset, as depicted in Figure 6. Upon examination, the initial two rows reveal the image-text prediction results based on our novel prompt-modified dataset on COCO. It can be observed that the network has the capability to accurately mask the specified objects in accordance with the location descriptions provided in the text prompt. Furthermore, the final two rows of the visualization indicate that the network’s performance is not restricted to the fine-tune dataset in terms of detecting the object with specific locations. This is evident when describing objects using natural language expressions, the network is still able to generate satisfactory segmentation outcomes. Additionally, our model can recognize the text prompts in natural language description, rather than strictly limited to the category labels provided by the dataset. For example, “athlete” and “girl” may be restricted to “person” within the dataset’s category labels. Leveraging the image-text alignment capability of CLIP, our model can precisely match the athlete and girls in the last two rows of Figure 6 to their corresponding objects and return the object masks accurately.
It has been demonstrated that the model outperforms existing panoptic segmentation techniques in various experiments. Although CLIP-FPS has shown promising results, further research and refinements could potentially lead to more advanced few-shot panoptic segmentation performance. Exploring alternative attention mechanisms and integration with other state-of-the-art models are potential directions for future work in this area.

**REFERENCES**


**TABLE 2. Panoptic segmentation performance comparison on the cityscapes validation dataset.**

<table>
<thead>
<tr>
<th>Methods</th>
<th>PQ (%)</th>
<th>SQ (%)</th>
<th>RQ (%)</th>
<th>mIoU (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TASCNet [18]</td>
<td>55.8</td>
<td>83.9</td>
<td>66.5</td>
<td>75.5</td>
</tr>
<tr>
<td>Seamless [40]</td>
<td>60.2</td>
<td>87.7</td>
<td>69.0</td>
<td>79.0</td>
</tr>
<tr>
<td>AUNet</td>
<td>59.1</td>
<td>85.8</td>
<td>68.9</td>
<td>77.2</td>
</tr>
<tr>
<td>Panoptic-FPN</td>
<td>58.1</td>
<td>85.0</td>
<td>68.4</td>
<td>76.5</td>
</tr>
<tr>
<td>AdaptIS</td>
<td>61.8</td>
<td>89.1</td>
<td>68.9</td>
<td>80.2</td>
</tr>
<tr>
<td>UPSNet</td>
<td>59.5</td>
<td>85.8</td>
<td>69.3</td>
<td>77.3</td>
</tr>
<tr>
<td>DeeperLab</td>
<td>56.6</td>
<td>83.6</td>
<td>67.7</td>
<td>75.3</td>
</tr>
<tr>
<td>SSAP</td>
<td>61.0</td>
<td>88.5</td>
<td>68.9</td>
<td>79.7</td>
</tr>
<tr>
<td>Panoptic DeepLab</td>
<td>62.4</td>
<td>89.4</td>
<td>69.8</td>
<td>80.4</td>
</tr>
<tr>
<td><strong>CLIP-FPS</strong></td>
<td><strong>63.7</strong></td>
<td><strong>90.7</strong></td>
<td><strong>70.23</strong></td>
<td><strong>81.7</strong></td>
</tr>
</tbody>
</table>