CRESO: CLIP-Based Referring Expression Segmentation for Object using Text Prompt

Subin Park
Department of Artifical Intelligence
Sejong University
Seoul, Republic of Korea
subinbing@sju.ac.kr

Zhegao Piao
Department of Computer Engineering
Sejong University
Seoul, Republic of Korea
piaozhegao5@sejong.ac.kr

Yeong Hyeon Gu*
Department of Artifical Intelligence
Sejong University
Seoul, Republic of Korea
yhgu@sejong.ac.kr
*Corresponding Author

***Abstract*— Text-based image segmentation is the task of segmenting specific objects in an image based on user-provided text prompts. To improve the performance of existing models, this paper emphasizes accurately embedding text conditions into image features and effectively capturing interactions between objects in complex visual scenes. In this paper, we propose the CRESO (CLIP-Based Referring Expression Segmentation for Objects) model using text prompt which builds upon the state-of-the-art CLIPSeg model. Our proposed CRESO model achieves superior accuracy in image segmentation by leveraging three key methods: the Feature-wise Linear Modulation (FiLM) Layer, Transformer Layer, and multi-scale feature fusion. Specifically, the FiLM Layer adjusts fine-grained image features using text embeddings to enable precise object segmentation based on text conditions, while the Transformer Layer utilizes a self-attention mechanism to learn contextual relationships between objects within an image, enhancing segmentation accuracy for complex and overlapping structures. Furthermore, multi-scale feature fusion enables the segmentation of objects with diverse sizes and intricate structures. Experimental results on the OCID-VLG dataset demonstrate that the CRESO model outperforms competitive baseline across diverse evaluation metrics and precision thresholds. This paper highlights the potential of the CRESO model for effectively addressing applications requiring accurate image segmentation guided by text prompts.**

Keywords—Multi-modal, Image Segmentation, Text Prompt, Referring Expression

I. INTRODUCTION

Image segmentation is a technology that identifies the location and shape of objects within a given image, serving as a fundamental component of image analysis and recognition [1, 2]. Among its subfields, text-based image segmentation, which focuses on segmenting specific objects based on textual input, has gained considerable attention in recent years. This approach enables the segmentation of objects in an image guided by user-provided prompts. It plays a pivotal role in improving the accuracy of obstacle recognition in critical domains such as medical diagnosis and autonomous driving [3-6]. As a result, text-based image segmentation has become an essential technology with applications spanning a wide range of industries.

Text-based image segmentation is a technology that segments objects within an image based on textual descriptions [7]. A notable example of a model in this domain is CLIP (Contrastive Language-Image Pre-Training), which is designed to learn from large-scale image-text pairs and identify the text most relevant to a given image [8]. A defining feature of CLIP is its ability to leverage the correlation between textual and visual modalities across diverse tasks without requiring task-specific pretraining, utilizing zero-shot learning [8]. Recently, numerous derivatives of the CLIP architecture have been introduced, each tailored to specific tasks. For example, CLIP-Decoder enhances multi-label classification by incorporating the attention-based architecture of the ML-Decoder, achieving state-of-the-art performance [9]. Similarly, ViLT-CLIP strikes a balance between supervised learning and generalization, excelling in applications such as video recognition and zero-shot learning [10]. In another adaptation, CLIPSyntel combines CLIP with a large-scale language model (LLM), demonstrating potential in specialized domains like medical disorder identification and context-aware question summarization [11]. Finally, CLIPSeg has been explicitly developed for text-prompt-driven image segmentation, enabling automated and precise object segmentation guided by textual inputs [2].

The CLIPSeg model is explicitly designed for text-based image segmentation tasks, aiming to segment specific objects within an image based on textual conditions. By leveraging the foundational capabilities of the CLIP model, CLIPSeg extends its functionality to effectively model the interaction between textual prompts and visual features. This enhanced understanding of cross-modal relationships enables the model to achieve precise and context-aware segmentation performance, making it well-suited for tasks requiring detailed object delineation guided by textual input.

CLIPSeg operates through three primary stages, each designed to facilitate precise text-based image segmentation. In the first stage, the input text and image data are transformed into their respective embeddings. The CLIP Text Transformer converts the text prompt into high-dimensional text embeddings, while the Visual Transformer processes the image through multiple Transformer blocks to produce a detailed image feature map. The second stage introduces the FiLM (Feature-wise Linear Modulation) layer, which modulates the image features based on the text embeddings. This mechanism selectively emphasizes or suppresses specific regions of the image according to the textual conditions, enabling a focused interpretation of the visual data. Additionally, the Projection Layer aligns the dimensions of the text and image embeddings, ensuring seamless integration of the two modalities for effective cross-modal interaction. In the final stage, the CLIPSeg Decoder, built upon a Transformer-based architecture, synthesizes the combined

text and image information to segment objects with precision. This stage is specifically designed to delineate object positions and boundaries based on the given text conditions. The model ultimately outputs an object mask that visually represents the position, shape, and extent of the segmented object in accordance with the textual input.

While CLIPSeg has demonstrated robust performance across diverse text conditions by effectively leveraging the relationship between text and images, we propose a novel approach to further enhance its capabilities. Our proposed enhancements aim to enable the model to more precisely capture and integrate text conditions and more effectively learn the interactions between multiple objects in complex and visually intricate scenes.

In this paper, we introduce CRESO (CLIP-Based Referring Expression Segmentation for Object) using Text Prompt, an enhanced model that builds upon the existing CLIPSeg framework to address challenges in text-guided image segmentation. The CRESO model integrates several key advancements: the Feature-wise Linear Modulation (FiLM) Layer effectively embeds detailed text embeddings into image features, enhancing segmentation accuracy in response to textual conditions. Additionally, features extracted by the Transformer Layer are refined and combined through Multi-scale Feature Fusion, enabling the model to better capture relationships between objects and to handle objects of diverse sizes and shapes with precision. These architectural enhancements enable the CRESO model to deliver superior segmentation performance even in complex images and under diverse text conditions.

The proposed CRESO model demonstrates remarkable capability in addressing complex image structures and multi-object environments, consistently outperforming existing state-of-the-art models. Experimental results validate the superior performance of the CRESO model. Specifically, on the OCID-VLG dataset, the model achieves significant advancements, including a Precision@50 of 95.19% and a Precision@90 of 93.33% at Epoch 30, surpassing the CLIPSeg baseline across all evaluated precision thresholds.

The main contributions of this paper can be summarized as follows. First, effectively addresses the challenge of embedding the semantic meaning of text prompts into image features with high fidelity through the Feature-wise Linear Modulation (FiLM) Layer. This enhancement substantially enhances the precision of text-guided object segmentation, significantly advancing the overall performance of text-based image segmentation. Second, tackles the complexities of modeling interactions between objects within an image and handling objects of varying sizes through the integration of multi-scale feature fusion and the Transformer Layer. Experimental results on the OCID-VLG dataset demonstrate significant improvements over the CLIPSeg baseline.

II. Related Work

Feature-wise Linear Modulation (FiLM) is a powerful technique for adjusting image features based on text conditions and has been recently applied in various studies. Research [12] introduced the FiLM technique, demonstrating its capability to emphasize or suppress specific regions in an image using text conditions. Additionally, a model incorporating FiLM Layers achieved high performance on the RefCOCO dataset by enhancing interactions between text and images in vision-language tasks. Research [13] proposed a multi-hop modulation method that adjusts visual features across multiple stages using FiLM, showing its effectiveness in handling interactions between text and images in complex visual inference tasks. Furthermore, research [14] introduced an approach to improve the performance of various vision tasks, including image segmentation, by using the FiLM technique to adjust feature maps according to text conditions.

Transformer-based models have demonstrated exceptional effectiveness in capturing features at various scales, leading to extensive research on integrating there features through Multi-scale Feature Fusion techniques. For instance, the ViT model, as introduced in [15], showcased its efficiency in imaging tasks by leveraging Transformers and refining detailed features via Multi-scale Fusion. Building on this concept, [16] integrated Multi-scale Feature Fusion into the Swin Transformer architecture, achieving significant improvements in the precision of text-image alignment by effectively combining multi-scale features. Furthermore, [17] applied Multi-scale Fusion within the Transformer Layer to enhance segmentation performance in high-resolution remote sensing imagery, demonstrating its capability to address challenges associated with detailed and large-scale visual data.

FiLM and Multi-scale Feature Fusion contribute to enhancing the performance of text-based image segmentation by adjusting features based on text conditions and fusing features across multiple scales, respectively. FiLM is particularly effective for fine-tuning image features through text embeddings, while Multi-scale Fusion facilitates the integration of global and local information within an image. The combination of these two techniques enables a detailed reflection of image features, significantly improving the accuracy of text-based image segmentation. In this paper, we propose an approach that integrates the FiLM Layer and Multi-scale Feature Fusion within a Transformer-based architecture to improve text-based object segmentation. This approach opens new possibilities aligned with the latest advancements in text-based image segmentation research.

III. Proposed Model

In this paper, we propose the CRESO (CLIP-Based Referring Expression Segmentation for Objects using Text Prompts) model for text-based object segmentation. The CRESO model is designed to take text prompts and images as inputs and accurately segment the images based on text conditions. The architecture comprises a text encoder, a visual encoder, and a decoder, with each component closely interacting to enable precise segmentation guided by textual input. Figure 1 illustrates the architecture of the proposed CRESO model. The CRESO model comprises three main steps:

Text encoding: In the first step, a given text prompt (e.g., "keyboard") is converted into an embedding vector using the CLIP Text Transformer. This vector represents the meaning of the text in a high-dimensional space. The FiLM Layer adjusts the image features based on the text embedding, enhancing the interaction between textual and visual modalities.

Image feature extraction: In the second step, the input image is processed by the CLIP Visual Transformer to extract visual features. The visual encoder, composed of multiple Transformer Blocks, processes image information at multiple

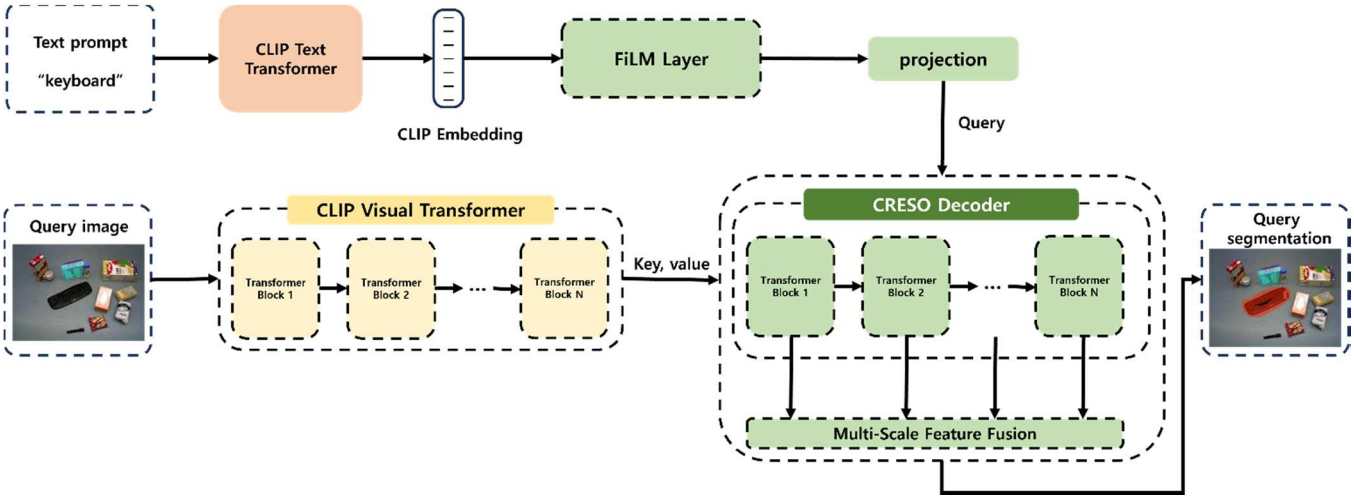

Figure 1. Structure of the CRESO model.

scales to generate visual features aligned with the text conditions. These extracted features are converted into Key and Value representations, ready to be combined with the text embedding.

Object segmentation: In the final step, the CRESO decoder performs object segmentation by combining text and image features. The decoder utilizes the query generated from the text prompt along with the Key and Value derived from the image to segment specific regions based on the text conditions. Configured with multiple Transformer Blocks and a Multi-Scale Feature Fusion module, the decoder is designed to accurately segment objects of various sizes and complex structures. The Multi-Scale Feature Fusion module integrates diverse information, enabling precise segmentation even in complex image environments.

A. FILAM Layer

The FiLM Layer plays a critical role in incorporating text information during image processing within the CRESO model. The text embedding generated by the CLIP Text Transformer is passed to the FiLM Layer, where it is transformed into two parameters, gamma and beta, which adjust the image features. In this process, the FiLM Layer calculates weights and bias values to embed the semantic and structural information of the text into the image features. After processing through the FiLM Layer, the text embedding undergoes a Projection step to align its dimensions with the image features produced by the CLIP Visual Transformer. Projection is necessary because text and image data have different dimensions; it adjusts the size and shape of the text embedding to facilitate comparison and computation in the same space. This step ensures that the information from the text embedding is preserved while converting it into a format compatible with the image features. Ultimately, the FiLM Layer adjusts specific channels of the image features by emphasizing, reducing, or shifting them based on the input text information.

B. CRESO Decoder and Multi-scale Feature Fusion

The CRESO Decoder and Multi-Scale Feature Fusion play a critical role in accurately segmenting objects in images based on text conditions and efficiently handling objects of various sizes and shapes. These two modules are designed to learn the correlation between text and images, enabling precise segmentation of objects that align with the given text conditions.

The CRESO Decoder combines text embeddings and image features to emphasize objects corresponding to the text conditions while de-emphasizing irrelevant regions. Text embeddings are generated using the CLIP Text Transformer and undergo a projection process before integration with image features in the CRESO decoder. During this process, the dimension of the text embedding is adjusted to match the output of the CLIP Visual Transformer, ensuring seamless integration of the two modalities. The projected text embedding is then utilized as a query in the CRESO Decoder.

Simultaneously, the CLIP Visual Transformer processes the input image to generate an image feature map, which contains information about each location and channel within the image. The output of the CLIP Visual Transformer is provided to the CRESO Decoder as the Key and Value. The Query, Key, and Value are combined through a Multi-Head Attention mechanism, enabling the model to learn the relationship between text and image. The Query represents information corresponding to the text condition, the Key calculates similarity between each image location and the text condition, and the Value emphasizes relevant regions or suppresses irrelevant areas.

For example, if the text condition is "red bag," the Query represents information about the red bag, the Key calculates the similarity between the red color and the bag-related regions in the image, and the Value highlights these regions. This process is iteratively performed through multiple Transformer Blocks, with each stage progressively extracting features that better match the text condition. In the initial Transformer Block stages, key areas related to the text are identified, while in the final Transformer Block stages, specific objects corresponding to the text condition are accurately separated.

The final output of the CRESO Decoder is a high-dimensional feature map representing the location and visual features of objects aligned with the text conditions. This output is passed to the Multi-Scale Feature Fusion module, which is used to segment objects of various sizes and shapes. Multi-Scale Feature Fusion, the final stage of the CRESO model, employs a multi-scale approach to handle objects of

different sizes. While the high-dimensional feature map from the CRESO Decoder is already enhanced based on the text conditions, the varying sizes and locations of objects in the image require hierarchical processing. The Multi-Scale Feature Fusion module extracts features at different resolutions (scales) from the input feature map: high-resolution features capture small objects, while low-resolution features are suited for large objects.

Features extracted at various scales are combined within the module. Essential features are retained, while irrelevant information is discarded. Fine-grained features from small scales enable precise capture of small object boundaries, whereas large-scale features capture the overall structure of the image and large objects. For instance, if the text condition is "a small blue box," the small scale accurately defines the boundaries of the box, while the large scale identifies its location, combining these pieces of information. The combined multi-scale features are ultimately used to accurately segment objects that meet the text condition. Multi-Scale Feature Fusion is designed to handle objects of varying sizes and plays a crucial role in enhancing the segmentation performance of the CRESO model.

IV. Experimental Results

A. Datasets

In this paper, the OCID-VLG dataset [18] was utilized for training and evaluating the CRESO model. This dataset is specifically designed for text-based object segmentation and consists of image-text pairs, with images containing objects of various sizes and shapes alongside their corresponding text descriptions. It is well-suited for performance evaluation in real-world application scenarios due to its inclusion of complex backgrounds and multiple objects. The image size was resized to 352 × 352 to match the input size requirements of the model.

The OCID-VLG dataset comprises a total of 32 classes, each containing a variety of objects commonly encountered in daily life. The text descriptions are designed to clearly refer to the corresponding objects, facilitating accurate understanding and segmentation by the model. Additionally, the dataset addresses the challenge of text-image matching, enabling the model to correctly segment specific objects in images based on text conditions. With its diverse range of objects, varying sizes and shapes, and complex backgrounds, the dataset provides a robust benchmark for evaluating the model's generalization performance and its ability to distinguish objects from their backgrounds. Furthermore, the inclusion of multiple objects within images allows for a precise assessment of the model's object detection and segmentation capabilities.

B. Experimental Setup

The main hyperparameters used in the training process of the CRESO model are as follows. The learning rate was set to $1e^{-4}$, and the batch size was set to 8. Binary Cross-Entropy was employed as the loss function to minimize the difference between the predicted mask and the actual mask. The model was trained for 10, 15, 20, and 30 epochs, respectively. The Adam optimizer was utilized as the optimization algorithm. The experiments were conducted in a CUDA environment using an NVIDIA GeForce RTX 3090, and the model was implemented and trained using the PyTorch framework.

C. Metric

In this paper, the Precision@Threshold method is employed to evaluate the model's performance. This method assesses segmentation performance by calculating the precision between the predicted mask, generated from the object segmentation results, and the actual mask (Ground Truth) at various threshold values. Specifically, Precision@Threshold is calculated using five threshold values: 0.5, 0.6, 0.7, 0.8, and 0.9. The method involves generating a binarized mask based on a specific threshold value from the predicted mask created by the model and calculating precision using True Positive (TP) and False Positive (FP) values between this mask and the actual mask. The calculation is mathematically expressed as shown in Equation (1).

$$Precision@k = \frac{TP_k}{TP_k + FP_k} \tag{1}$$

Here, k represents a specific threshold, TP_k denotes the number of pixels correctly classified as objects in the predicted mask, and FP_k denotes the number of pixels incorrectly classified as objects but do not correspond to actual objects. This metric evaluates the accuracy with which the predicted object regions align with the actual object regions.

To calculate Precision@Threshold, the text condition (prompt) and the image are first input into the model to generate the prediction mask. The prediction mask outputs continuous values within the range [0, 1] for each pixel, representing the probability of being part of an object. Binarization is then performed on the prediction mask based on each threshold value (0.5, 0.6, 0.7, 0.8, 0.9). For example, if the threshold value is 0.7, pixel values of 0.7 or higher in the prediction mask are converted to 1 (object), while those below 0.7 are converted to 0 (background). The Precision value is then calculated using the binarized mask, and this process is repeated for each threshold value.

D. Results and Discussion

As shown in Table 1 and Figure 2, the CRESO model demonstrated superior performance compared to the existing CLIPSeg [2] model across various Precision thresholds (Precision@50 to Precision@90). In the results for Epoch 10, the CRESO model achieved an accuracy of 94.29% at Precision@50, 92.98% at Precision@70, and 85.15% at Precision@90, which represent improvements of 1.0%, 2.12%, and 7.73% over the CLIPSeg model, respectively. These results highlight the CRESO model's ability to perform more detailed and accurate object detection in text-based object segmentation tasks compared to the CLIPSeg model.

The results at Epoch 30 further highlight the enhanced performance of the CRESO model. At this epoch, the CRESO model achieved 95.19% at Precision@50, 95.44% at Precision@70, and 93.33% at Precision@90, representing improvements of 0.55%, 0.71%, and 1.18% over the CLIPSeg model, respectively. These results confirm that the CRESO model can provide stable and consistent object segmentation even in environments with complex image structures and multiple objects. Notably, the high precision achieved at Precision@90 demonstrates the model's capability for detailed segmentation performance based on text conditions.

The performance trends visualized in Figure 2 illustrate the changes in Precision@70 (Pr@70) across epochs. Both models exhibit gradual performance improvements as the number of epochs increases. However, from Epoch 10 to Epoch 30, the CRESO model consistently outperformed the CLIPSeg model [2]. Specifically, at Epoch 30, the CRESO

TABLE I. PERFORMANCE RESULTS OF CLIPSEG AND CRESO MODEL

Epoch	Model	Pr@50	Pr@60	Pr@70	Pr@80	Pr@90
10	CLIPSeg	93.29	92.84	90.86	87.46	77.42
	CRESO (ours)	**94.29**	**94.19**	**92.98**	**90.67**	**85.15**
15	CLIPSeg	92.72	92.84	91.70	90.05	84.43
	CRESO (ours)	**93.94**	**93.80**	**93.19**	**91.28**	**87.68**
20	CLIPSeg	93.74	93.71	92.25	90.55	86.10
	CRESO (ours)	**94.47**	**94.59**	**93.95**	**91.80**	**87.91**
30	CLIPSeg	94.64	94.76	94.73	94.02	92.15
	CRESO (ours)	**95.19**	**95.34**	**95.44**	**94.84**	**93.33**

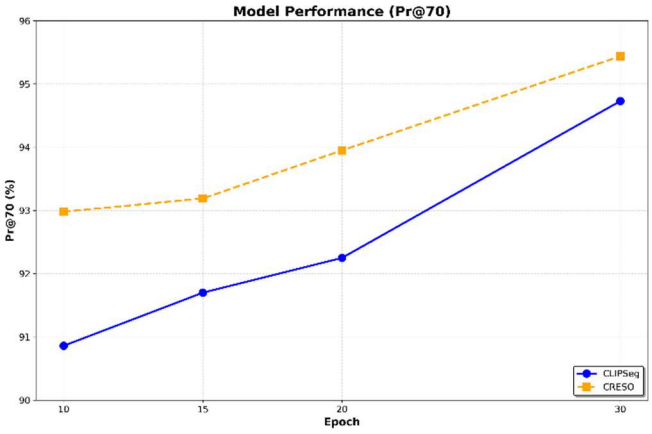

Figure 2. Performance comparison of CLIPSeg and CRESO at Pr@70 across Epochs 10, 15, 20 and 30.

model achieved 95.44% at Pr@70, a 0.71% higher precision compared to the CLIPSeg model. These results indicate that the CRESO model not only achieves superior accuracy but also maintains stable and consistent performance in complex image structures and multi-object environments.

V. LIMITATIONS

As the CRESO model was primarily trained and evaluated on the OCID-VLG dataset, potential limitations in its generalization performance may arise when applied to other datasets or real-world environments. The OCID-VLG dataset includes objects of specific sizes and types, which may result in suboptimal performance when the model encounters novel objects or complex environments outside the dataset's scope. This reliance on the characteristics of a specific dataset could lead to limitations in the model's adaptability. To address this issue, additional validation and training on diverse datasets, as well as real-world application scenarios, are required.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed the CRESO model to address the limitations of the existing CLIPSeg model and significantly enhance performance in text-based object segmentation tasks. The CRESO model extends the CLIPSeg architecture by incorporating the FiLM Layer and Transformer Layer to precisely capture the relationship between text and images. The FiLM Layer finely adjusts image features based on text embeddings, ensuring that text conditions are accurately reflected, while the Transformer Layer learns complex relationships between objects in the image, enabling highly accurate object segmentation under diverse text conditions. Additionally, the Multi-Scale Feature Fusion technique effectively handles objects of varying sizes and complex structures.

Experimental results demonstrate that the CRESO model outperforms the existing CLIPSeg model across various Precision thresholds, ranging from Precision@50 to Precision@90. Notably, the model maintains high accuracy even in complex image structures and multi-object environments, exhibiting superior performance in text-based object segmentation tasks. The proposed CRESO model, through its innovative architecture, achieves superior text-guided segmentation accuracy. Future work will focus on expanding its generalization capability to real-world datasets, reducing its dependency on the OCID-VLG dataset, and exploring its applicability in diverse industrial contexts.

In future studies, we aim to reduce the CRESO model's dependency on the OCID-VLG dataset and enhance its generalization to broader real-world environments. While this paper primarily conducted experiments on the OCID-VLG dataset, validation in other domains or real-world scenarios remains limited. To address this, additional training and validation on multi-domain datasets will be pursued to improve the model's robustness and adaptability.

ACKNOWLEDGMENT

This work was supported by the Technology Innovation Program(RS-2024-00421828, Development of artificial intelligence software for unseen object manipulation that integrates prompt and situation-specific unseen object recognition and arbitrary gripper shape analysis through gripper self-observation) funded by the Ministry of Trade Industry & Energy (MOTIE, Korea).

REFERENCES

[1] G. Mehul, P. Ankita, D. Namrata, G. Rahul, and S. Sheth, "Text-based image segmentation methodology," Procedia Technology, vol. 14, pp. 465-472, 2014.

[2] T. Lüddecke and A. Ecker, "Image segmentation using text and image prompts," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 7086-7096.

[3] J. Lampkins, D. Chan, A. Perry, S. Strelnikoff, J. Xu, and A. E. Ashari, "Multimodal road sign interpretation for autonomous vehicles," in 2022 IEEE International Conference on Big Data (Big Data), 2022, pp. 5979-5987: IEEE.

[4] H. Ali, M. F. Bulbul, and Z. Shah, "Prompt Engineering in Medical Image Segmentation: An Overview of the Paradigm Shift," in 2023 IEEE International Conference on Artificial Intelligence, Blockchain, and Internet of Things (AIBThings), 2023, pp. 1-4: IEEE.

[5] X. Zeng et al., "ABP: Asymmetric Bilateral Prompting for Text-Guided Medical Image Segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention, 2024, pp. 54-64: Springer.

[6] S. Jiao and Y. Fang, "LaVida Drive: Vision-Text Interaction VLM for Autonomous Driving with Token Selection, Recovery and Enhancement," arXiv preprint arXiv:2411.12980, 2024.

[7] K. Pnvr, B. Singh, P. Ghosh, B. Siddiquie, and D. Jacobs, "Ld-znet: A latent diffusion approach for text-based image segmentation," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4157-4168.

[8] A. Radford et al., "Learning transferable visual models from natural language supervision," in International conference on machine learning, 2021, pp. 8748-8763: PMLR.

[9] M. Ali and S. Khan, "Clip-decoder: Zeroshot multilabel classification using multimodal clip aligned representations," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4675-4679.

[10] H. Wang et al., "ViLT-CLIP: Video and Language Tuning CLIP with Multimodal Prompt Learning and Scenario-Guided Optimization," in

Proceedings of the AAAI Conference on Artificial Intelligence, 2024, vol. 38, no. 6, pp. 5390-5400.
[11] A. Ghosh, A. Acharya, R. Jain, S. Saha, A. Chadha, and S. Sinha, "Clipsyntel: clip and llm synergy for multimodal question summarization in healthcare," in Proceedings of the AAAI Conference on Artificial Intelligence, 2024, vol. 38, no. 20, pp. 22031-22039.
[12] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in Proceedings of the AAAI conference on artificial intelligence, 2018, vol. 32, no. 1.
[13] F. Strub et al., "Visual reasoning with multi-hop feature modulation," in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 784-800.
[14] M. Brockschmidt, "Gnn-film: Graph neural networks with feature-wise linear modulation," in International Conference on Machine Learning, 2020, pp. 1144-1152: PMLR.
[15] A. Dosovitskiy, "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.
[16] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 10012-10022.
[17] P. Zhang et al., "Multi-scale vision longformer: A new vision transformer for high-resolution image encoding," in Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 2998-3008.
[18] G. Tziafas, Y. Xu, A. Goel, M. Kasaei, Z. Li, and H. Kasaei, "Language-guided robot grasping: Clip-based referring grasp synthesis in clutter," arXiv preprint arXiv:2311.05779, 2023.