Sensors and Materials, Vol. 37, No. 3 (2025) 1193–1209
MYU Tokyo

S & M 3981

Approach to Deep-learning-based Visual Relationship Detection for Scene Analysis

Ming-Yuan Shieh, Po-Kuan Wu, and Neng-Sheng Pai*

Department of Electrical Engineering, National Chin-Yi University of Technology, Taichung 41170, Taiwan

(Received June 28, 2024; accepted March 17, 2025)

Keywords: visual relationship detection, scene analysis, convolutional neural networks, vision transformer

The paper is focused on the implementation of a deep-learning-based visual relationship detection system for scene analysis. Initially, the system employs convolutional neural networks (CNNs) for precise object detection and localization, accurately capturing detailed information about the objects in the scene. Following this, the system applies the vision transformer model to infer relationships between objects, enabling detailed analysis, interpretation of spatial layouts, and understanding of behavioral interactions among objects within the scene. This process allows the system to gain a profound understanding of the relationships and interaction patterns among the objects. Furthermore, the system integrates an intuitive and feature-rich user interface to display detailed relationships among objects and the model's inference outcomes. We not only conduct scene analysis but also provide functionality for users to select two distinct subjects for visual relationship detection. Users can choose two subjects of interest to explore their relationships with other objects in the same scene. Through this selection capability, the system analyzes the selected subjects while excluding results that do not meet specified criteria. This functionality enables users to precisely control the analysis direction, ensuring that the results align with their expectations and requirements effectively.

1. Introduction

The aim of scene understanding is to enable computers to comprehend and interpret the content in images, similar to human-like understanding. This technology is essential in various applications, such as autonomous vehicles needing to identify and understand objects and situations on the road for safe driving; intelligent surveillance systems that require the ability to differentiate between normal activities and potential threats for timely responses; and robot navigation, which necessitates the recognition of obstacles and targets in unfamiliar environments to move and operate effectively. By employing scene understanding, computers can make more accurate decisions in complex environments, thereby enhancing system reliability and efficiency. These capabilities not only improve the performance of automated

*Corresponding author: e-mail: pai@ncut.edu.tw
https://doi.org/10.18494/SAM5220

ISSN 0914-4935

systems but also expand their applicability, driving technological advancements across industries.

In scene understanding, a scene graph represents objects and their relationships within an image. The nodes in a scene graph correspond to individual objects, while the edges denote the relationships between these objects.[1] Scene graph generation is a technology focused on automatically extracting and structuring representations of objects and their relationships from images, closely associated with the detection of visual relationships.[2] Presently, scene graphs have showcased their potential in a variety of visual language tasks, including image retrieval,[3] image annotation,[4] visual question answering (VQA),[5] and image generation.[6] The task of scene graph generation continues to attract attention in the computer vision community. Within the field of scene understanding, inferring structural attributes between entities, especially visual relationships among objects, is a fundamental task. These visual relationships are usually represented by triplets. However, generating a large number of object relationships can lead to information overload, which complicates the scene, making it more difficult for users to comprehend.

To address these challenges, we utilize a visual relationship detection system for scene analysis employing deep learning. It combines CNNs for object detection and leverages the vision transformer model for relationship inference. This approach facilitates a comprehensive exploration and understanding of the relationships and behavioral patterns among objects in images, improving detection accuracy and efficiency. Moreover, users can choose specific subjects via the system interface, granting them precise control over the analysis direction to investigate the relationships between their selected subjects and other objects, thus preventing unnecessary outcomes.

2. Related Works

The paper is focused on the latest advancements in compositional semantic construction within images.[7] High-quality semantic construction provides a deeper understanding of scenes and serves as the basis for various visual language tasks, including image retrieval, image annotation, VQA, and image generation. Scene graphs present a visual framework that represents objects and their relationships within images, greatly enhancing semantic understanding. For instance, the scene graph generation model proposed by Xu *et al.* is employed for the automatic extraction and structuring of object representations and their relationships from images.[8] Ji *et al.* further applied scene graphs to image retrieval, showcasing their potential in semantic search.[9] In scene graph analysis, visual relationship detection not only delineates the areas where objects are located but also describes their interactions.

Visual relationship detection is essential for understanding the interactions and connections among objects in images, greatly enhancing image comprehension and scene analysis capabilities. Therefore, the application of visual relationship detection facilitates the resolution of semantic relationship understanding issues among objects in images, including inference types such as verbs,[10] actions,[11] spatial or comparative phrases,[12] and more. This paper is dedicated to exploring and applying visual relationship detection techniques to analyze

relationships among objects in images and implementing them in real-world scenarios. These methods primarily focus on analyzing and inferring the detected triplets (obj1, relationship, obj2) in images. Zhou *et al.* proposed a unified visual relationship detection framework incorporating the location ranking module (LRM) and label graph module (LGM) to address issues in the object-pair-proposing and predicate-recognition stages.[13] Zhu and Wang introduced a multiscale conditional relationship graph network (CRGN) framework for entity localization based on visual relationships.[14]

In the context of scene analysis, relying solely on visual relationship detection networks can present several challenges. Firstly, the lack of adequate contextual information may result in misjudgments, as the network might not fully understand the roles and interactions of objects in the scene. Secondly, visual interference, such as noise and nontarget objects, can affect detection accuracy. Additionally, visual relationship detection networks may struggle with multiscale issues, making it difficult to efficiently identify objects of various sizes and scales along with their relationships. Some models address these issues by incorporating the fully convolutional scene graph generation (FCSGG) model for simultaneous object and relationship detection.[15] A novel simple network architecture, based exclusively on attention mechanisms, that replaces traditional complex recurrent or CNN structures[16] has been proposed. Furthermore, in certain studies, CNNs and transformer models have been combined to enhance visual relationship detection, leading to a new approach called detection transformer (DETR),[17] which treats object detection as a set prediction problems to eliminate nonmaximum suppression and anchor generation. The model uses a transformer encoder–decoder architecture and achieves unique predictions through bipartite matching.

From the above discussion, it is clear that visual relationship detection in scene analysis is essential for understanding the interactions and connections between objects in images. Solely relying on visual relationship detection networks can lead to mistakes and reduced accuracy because they lack comprehensive information, can be affected by irrelevant visuals, and struggle with multiple scales. In response to these challenges, researchers have proposed various methods and architectures that significantly enhance the performance of visual relationship detection through the integration of global contextual information and multiscale features. These methodologies effectively improve the accuracy of object detection and relationship understanding, laying a solid technical foundation for tasks such as image retrieval, image annotation, and VQA.

3. Scene Analysis and Its Architecture

To effectively understand and analyze complex scenes, in this paper, we propose a deep-learning-based system for visual relationship detection. The system is designed to enhance the understanding of relationships between objects in scenes for machine vision, enabling more intelligent and precise scene analysis across various applications. To achieve this goal, it is essential to tackle challenges related to data collection, model training, and user interface design.

Our system consists of two processes. Firstly, it collects images of target objects to ensure there is sufficient data for model training. Subsequently, the model undergoes training to develop

its capability for scene analysis. During this phase, deep learning methods and extensive computational resources are employed to guarantee the precision and dependability of the model. Additionally, a user-friendly and intuitive interface is developed to provide users with clear information about scene contents. During the scene analysis stage, users can select their subject of interest for visual relationship detection, utilizing spatial layout and semantic connections to understand the complex interactions and structures in the scene, resulting in more detailed object relationship outcomes and enhancing the overall comprehension of scene information. Once the analysis is done, the results are converted into spoken words and given as feedback using speech recognition technology, which improves both the user comprehension of scene details and the system interaction. Ultimately, these procedures together constitute the scene analysis and produce comprehensive results, as shown in Fig. 1.

The aim of the proposed scheme is to utilize visual relationship detection for analyzing and comprehending real-world scenes. Initially, a CNN extracts fundamental visual features, including edges, colors, and textures from scene images, which serve as the foundation for subsequent processing. Leveraging the objects identified by the CNN, the system employs its designed functionality to select subjects of interest for visual relationship detection. Next, these features are fed into a transformer model, which utilizes an attention mechanism to dynamically capture and analyze the significance of different areas in the image, generating attention maps for subjects and objects. These attention maps are used to analyze the details of the spatial and semantic relationships between objects in the scene. For example, interactions such as that between a boy and a bike can be accurately identified. This analysis assists in detecting the interrelationships between objects in the image and in describing the results of these relationships in text format, as shown in Fig. 2. The integration of the CNN and transformer architectures not only enhances the system's comprehension of scenes but also boosts object detection accuracy by effectively managing object relationships within those scenes.

The system flowchart illustrates the complete operational process of a visual relationship detection system, as shown in Fig. 3. The system starts by performing object detection using visual techniques to identify objects in the scene. Users can select a topic or an object of interest, and the system determines whether to continue on the basis of the user's selection. If the user

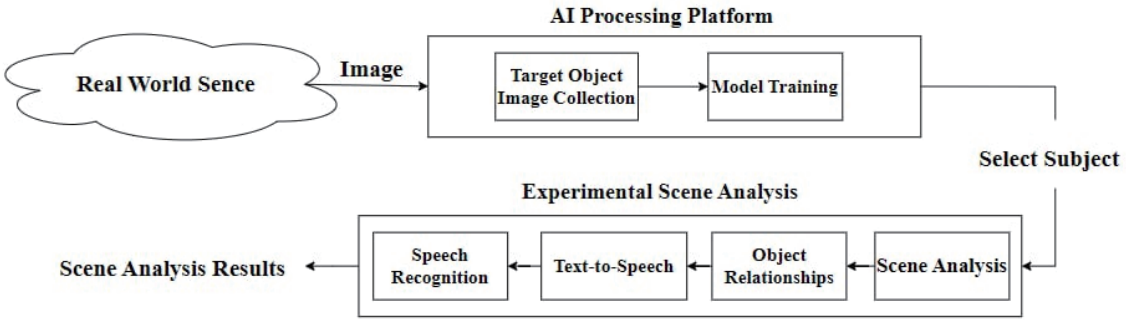

Fig. 1. Architecture of visual relationship detection system.

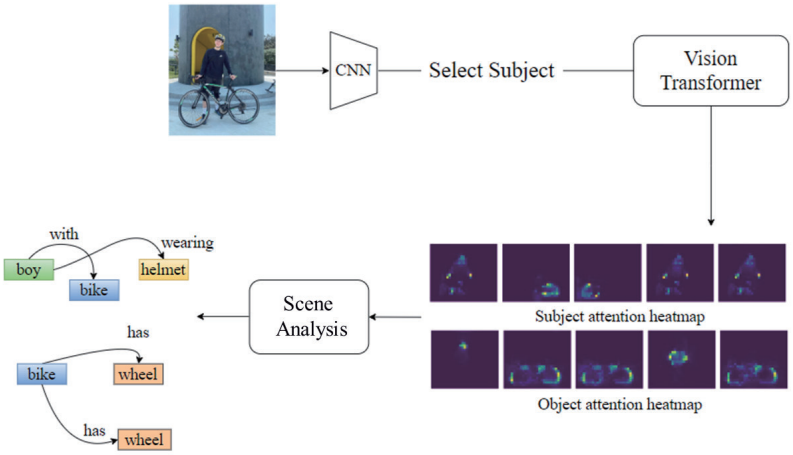

Fig. 2. (Color online) Visual relationship detection model architecture.

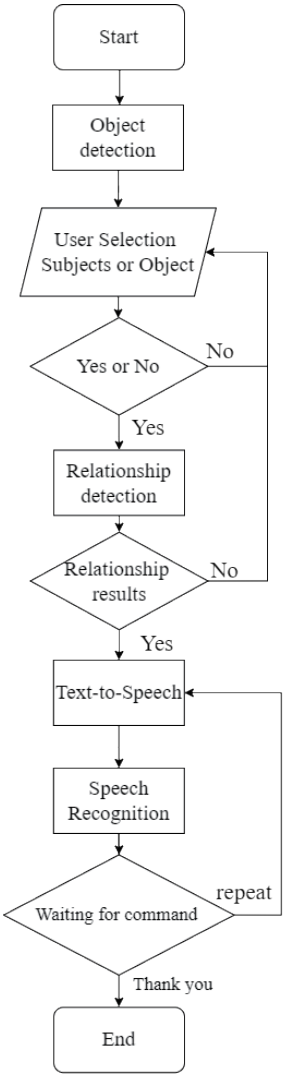

Fig. 3. Visual relationship detection system flowchart.

chooses to continue, the system will conduct relationship detection, analyze the relationships between objects, and output the results in text and voice. Users can interact with the system through voice recognition to provide further instructions. The system remains in a standby state while awaiting user instructions and politely expresses gratitude at the end of the process before terminating the operation.

3.1 Deep-learning-based object detection

We aim to use the CNN architectures as the initial step in visual relationship detection during scene analysis. The CNN, a deep-learning model specifically designed for processing and analyzing visual data, automatically learns hierarchical features from images, making it highly effective for image processing and analysis. These hierarchical feature representations effectively address various visual tasks and are crucial for scene analysis.

The task of visual relationship detection is well suited to training with a moderately deep model such as ResNet-50. In ResNet-50, with its effective residual block design, U represents the output of the residual block, V represents the input, $F(x,\{W_i\})$ denotes the residual function to be learned, and $\{W_i\}$ are the parameters of the network layers. This formula embodies the core idea of residual learning by introducing shortcut connections, allowing the network to learn the residuals between the input and output within each residual block. Therefore, the conversion equation can be modified as follows.

$$U = F\left(V, \{W_i\}\right) + V \tag{1}$$

In this study, we adopt the ResNet-50 architecture as the primary method for visual relationship detection because of its effectiveness in handling complex image data while maintaining efficiency and high accuracy in a moderately deep network design. A significant feature of ResNet-50 is its implementation of residual blocks, which allow inputs to bypass several layers and to be directly transmitted to the output. This design effectively addresses the gradient vanishing problem that often arises in deep networks, enabling deeper architectures to learn richer feature representations. As a result, the use of ResNet-50 leads to high accuracy without compromising performance, eliminating the need for more complex network architectures to meet our research objectives. Furthermore, ResNet-50 demonstrates consistent model performance and reliable training processes. Therefore, selecting ResNet-50 as the foundational model in this study establishes a robust technical framework for visual relationship detection.

3.2 Visual relationship detection using vision transformer

To effectively execute visual relationship detection and scene understanding, a model architecture based on the vision transformer is developed. The ResNet-50 for image object detection is used to extract objects and their attributes, which helps in generating high-quality features for subsequent visual relationship detection. Transitioning to the vision transformer

architecture enabled us to capture object relationships and generate detailed object attention maps, thereby enhancing scene understanding. This framework allows the efficient analysis of object relationships in images while visually monitoring the model's areas of focus during the decision-making process.

First, when the model inputs an image, it is subjected to object detection using the ResNet-50 model to extract objects and their attributes. Subsequently, the image regions based on the user-selected subject are then sent to the vision transformer encoder. The vision transformer is a deep learning model based on the transformer architecture tailored for processing image data, as shown in Fig. 4. Each image region is transformed into an embedding vector capturing local image features. Then, the upper section utilizes the vision transformer encoder to process the output embedding vectors via a linear layer, generating object image embeddings. Simultaneously, the text encoder transforms textual data into embedding vectors that represent vocabulary features. Finally, object image embeddings are matched with text embeddings to compute the similarity between each object image embedding and its corresponding text embedding, thus generating class predictions. The lower section of the model architecture processes the output embedding vectors from the vision transformer encoder using a multilayer perceptron (MLP) to create object box embeddings. These object box embeddings are then used to predict the bounding boxes of objects in the image, where each bounding box is represented as (x, y, w, h) with (x, y) as the top-left coordinates and (w, h) as the width and height of the box.

The attention map generated by the vision transformer encoder highlights the specific areas of focus during object classification and localization. Through the attention map, it becomes clear how the model allocates attention when processing particular objects. If the attention map reveals too much focus on irrelevant areas, adjustments to the model's weights or architecture may be necessary to improve the overall accuracy. This attention map offers a detailed analysis of the scene, enabling our research to concentrate on precise object classification and localization in images, thereby enhancing the model's capabilities and performance, as shown in Fig. 5.

By inputting images into the system and selecting the desired "subject" using the above framework, this system not only detects the main objects in the images but also reveals the relationships between these objects. For example, by selecting an image containing a boy and a

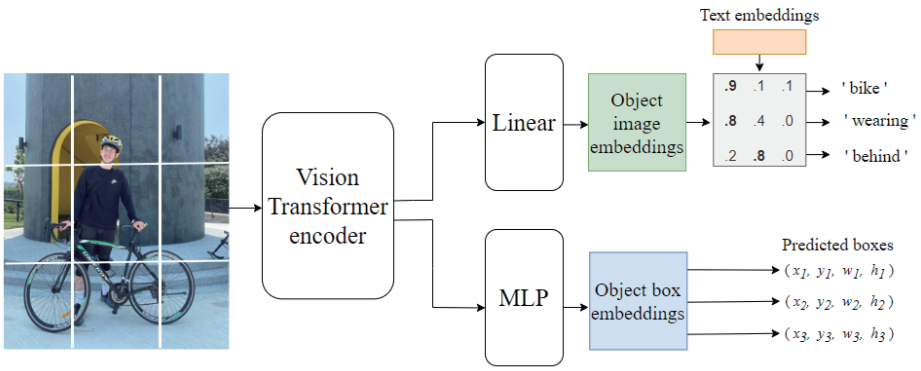

Fig. 4. (Color online) Vision transformer architecture.

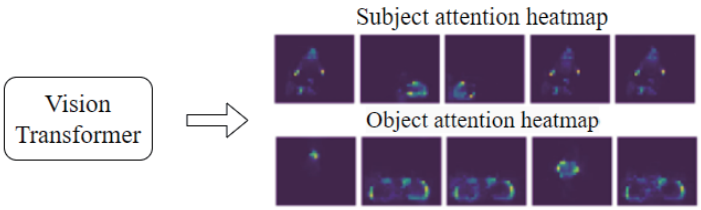

Fig. 5. (Color online) Object attention heatmap in visual relationship detection system.

bike, the model can detect objects such as the boy, the bike, the helmet, and the wheel. Through the subject selection feature developed in this study, the system accurately identifies and annotates the relationships among these objects. When the boy is selected as the subject, the system will display relationships such as "boy wearing helmet" and "boy riding bike". Furthermore, if the bike is selected as the second subject, relationships such as "the bike has wheels" will be shown, as illustrated in Fig. 6. By utilizing these detection results, users can intuitively understand the objects in the image and their interrelationships.

4. Experiments

4.1 Interface display of visual relationship detection results

Users can conduct scene analysis through the visual relationship detection interface designed for this system. The interface is aimed at providing an intuitive and user-friendly tool for uploading images of scene content and performing visual relationship detection. Specifically, users can upload an image by clicking the upload button, and the image is displayed on the interface. Subsequently, users can select the first and second objects in the image from the dropdown menu. This design feature caters to various application needs, as different users may have varying points of focus regarding object relationships within images. An example is smart home systems, where users may focus on the spatial layout relationships between specific furniture and other objects (such as sofas and tables). Analyzing these objects can help optimize the home layout and enhance living comfort. This design not only offers practical benefits for users but also holds crucial significance for researchers. Researchers can select various objects for visual relationship detection experiments to verify the accuracy and reliability of the system, while also collecting data to further improve it. Through this design feature, users can comprehensively understand the visual relationship information present in the images, as shown in Fig. 7.

In the first scene, as shown in Fig. 7, if users do not select any objects from the dropdown list, the system will automatically showcase all detected objects along with their relationships. As shown in Fig. 8, the object relationship content appears in the lower area of the interface after detection outlining the relationships between all objects. For instance, descriptions include statements like "In this picture, I see the bike in the foreground, and the bike has wheels" and "In this picture, I see the boy in the foreground, and the boy is wearing a helmet" to convey relationship information.

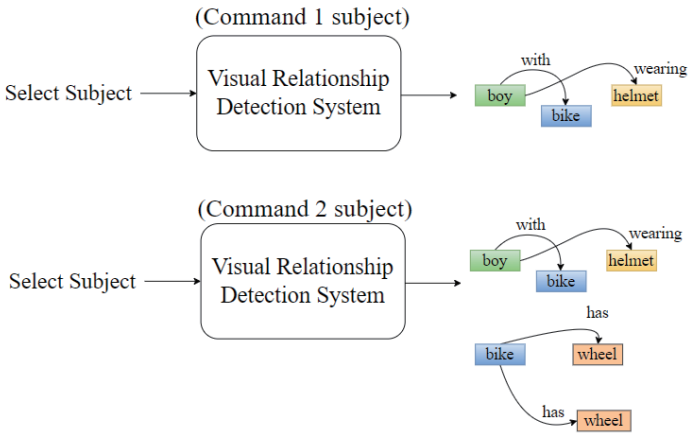

Fig. 6. (Color online) Analysis of visual relationship detection results.

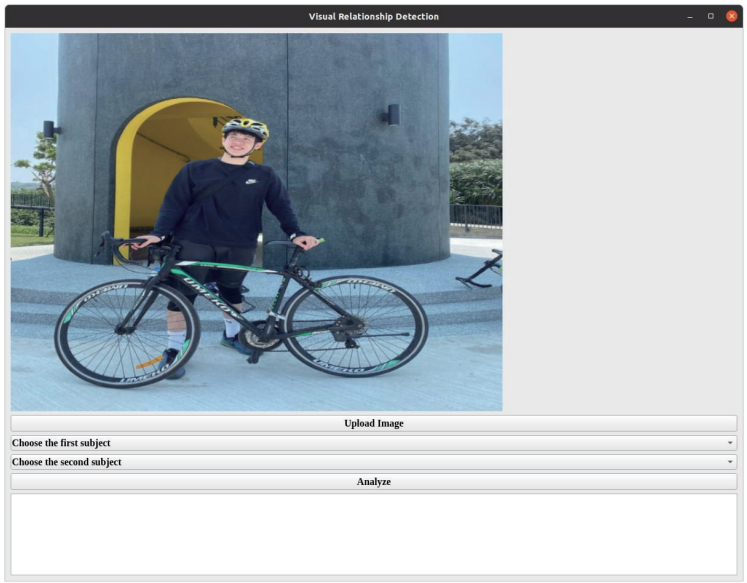

Fig. 7. (Color online) Visual relationship detection system interface.

Once all object relationships are detected, attention maps for each object are provided as well. These attention maps illustrate the importance and focus points of each object within the image. Three out of the seven different relationships have been selected for illustration within the interface. In each example, the attention maps for both the subject and the object are displayed above the image, while the corresponding bounding box detection results are presented below the image, as illustrated in Fig. 9.

After displaying the visual relationship text content, the system activates the text-to-speech function to vocalize the descriptions, allowing users to listen to the analysis results. This text-to-speech function enhances the system's usability, particularly for users encountering visual or reading challenges. Audio prompts create an intuitive and user-friendly experience. For

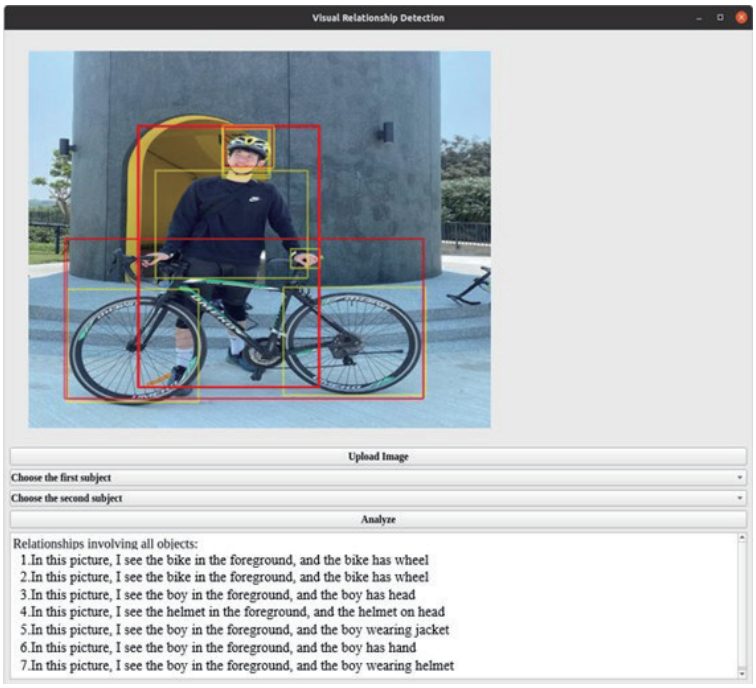

Fig. 8. (Color online) Visual relationship detection results (1st scene).

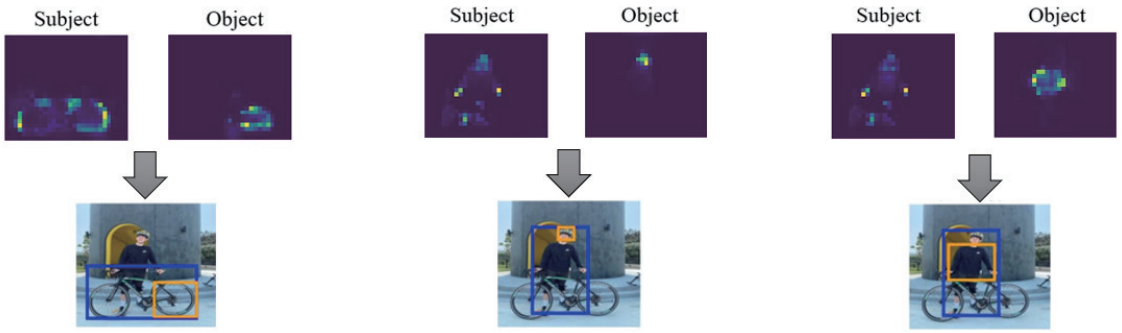

Fig. 9. (Color online) Visual relationship detection attention maps (1st scene).

individuals with visual impairments, reading text on the screen can be difficult. However, with audio prompts, they can easily grasp the system's analysis results, thereby enhancing usability. After the audio narration is complete, the system initiates a speech recognition function to listen for commands like "repeat" to replay the content or "thank you" to close the interface window. The speech recognition function enhances interactivity, allowing users to control the system through voice commands.

Specifically, the system waits for 'repeat' or 'thank you' commands and responds appropriately. If the user says "repeat," the system replays the content. If the user says "thank you," the system responds with "Thank you, goodbye, and good luck" before closing the window. If the system is unable to understand the command, it displays "Could not understand

audio," as shown in Fig. 10. The system is presently set up with these two basic commands, but it also can incorporate additional interactive voice commands to meet specific needs. For example, commands such as "show more" can be implemented to display additional content, while "stop" can pause current activities. These extra voice commands can greatly improve the system's flexibility and interactivity, enabling users to navigate it more smoothly and catering to the diverse requirements of various users.

To make it easier for users to select objects, the system interface presents a list of all available objects in two dropdown menus. In the two dropdown lists, the system employs a design that integrates both text and images. This design not only offers more comprehensive information but also helps address semantic ambiguity issues. For instance, the same object may have different names in various contexts, and the images help users confirm the specific object they have selected, as shown in Fig. 11.

Fig. 10. (Color online) Visual relationship detection voice function (1st scene).

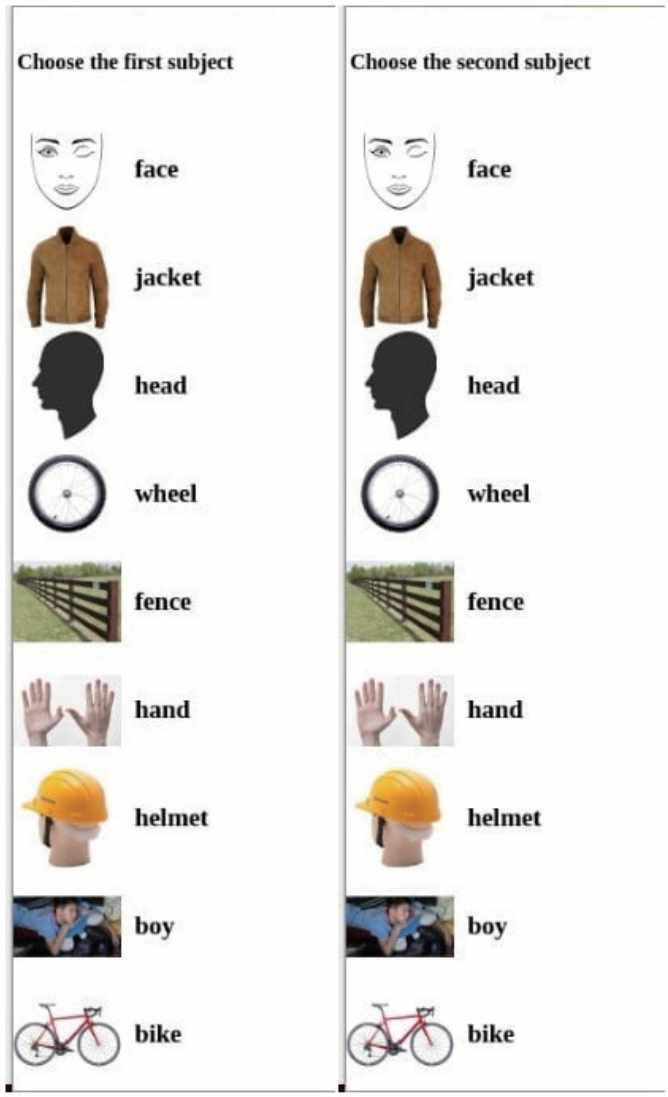

Fig. 11. (Color online) Dropdown lists in the visual relationship detection interface (1st scene).

Upon selecting a subject from the first dropdown list, the system promptly generates and presents all relationship descriptions associated with the chosen subject. For example, selecting "boy" from the first dropdown list will display "In this picture, I see the boy in the foreground, and the boy is wearing a helmet", as shown in Fig. 12. Subsequently, when the user selects the second subject from the second dropdown list, the system analyzes the interactive relationships between the two selected subjects and other objects. For instance, if the user chooses the boy as the first subject and the bike as the second subject, the system promptly generates and displays the interactive relationship descriptions in the result area, such as "I see the boy in the foreground, and the boy is wearing a helmet" and "In this picture, I see the bike in the foreground, and the bike has wheels", as shown in Fig. 13.

To ensure the system's feasibility, after the detection in the first scene is completed, the system will proceed to the scene analysis in the second scene. On the basis of the content of the

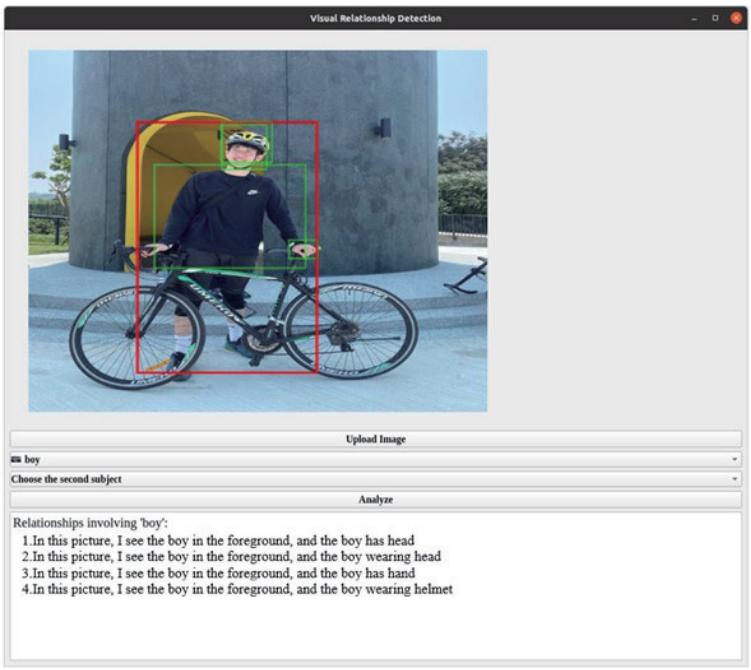

Fig. 12. (Color online) One-subject result of visual relationship detection (1st scene).

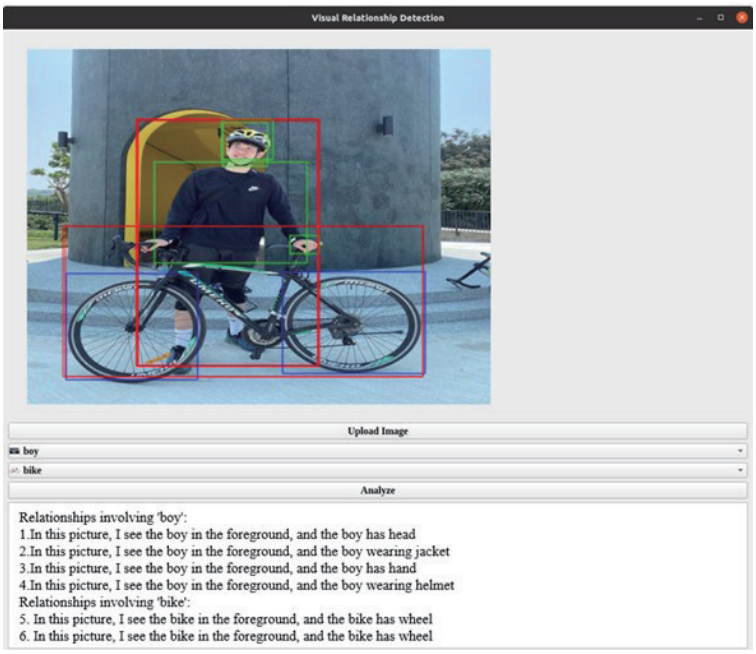

Fig. 13. (Color online) Two-subject result of visual relationship detection (1st scene).

image in the second scene, the system first performs object detection and stores the detected object category names in a dropdown list. Users select subjects from two dropdown lists that display all detected objects, as shown in Fig. 14.

Fig. 14. (Color online) Dropdown lists in the visual relationship detection interface (2nd scene).

When the user selects an object from the first dropdown list, the system immediately generates and displays all relationship descriptions related to the selected object. For example, selecting "man" as the first subject will display "In this picture, I see the man in the foreground, and the man wearing a shirt", as shown in Fig. 15. Subsequently, when the user selects the second subject from the second dropdown list, for instance, "book", the system will instantly generate and display interactive relationship descriptions in the result area. These descriptions may include, "I see the man in the foreground, and the man is wearing a shirt" and "I see the book in the foreground, and the book is on the table", showcasing various relationship contexts, as depicted in Fig. 16.

To ensure stable convergence and enhance the model's performance and accuracy during training, the following training parameter settings are implemented in this study. The initial learning rate is established at 0.0001, gradually decreasing to 0.00001 by the 50th epoch. This strategy facilitates swift learning during the initial stages of training while fine-tuning parameters as training progresses. Utilizing the Adam optimizer with a weight decay of 0.0001 mitigates the risk of overfitting. The batch size is configured at 128, enabling efficient model training even with limited GPU memory capacity. Training is conducted over 150 epochs to allow ample time for the model to learn and converge.

Moreover, we applied recall at a specific number of detections (R@Call) as the accuracy metric derived from training on the Visual Genome dataset,[7] comparing it with the accuracies reported in scholarly works focusing on scene graph detection. R@Call serves as an evaluative metric to gauge system performance and juxtapose the efficacy of different methodologies. The results are shown in Table 1.

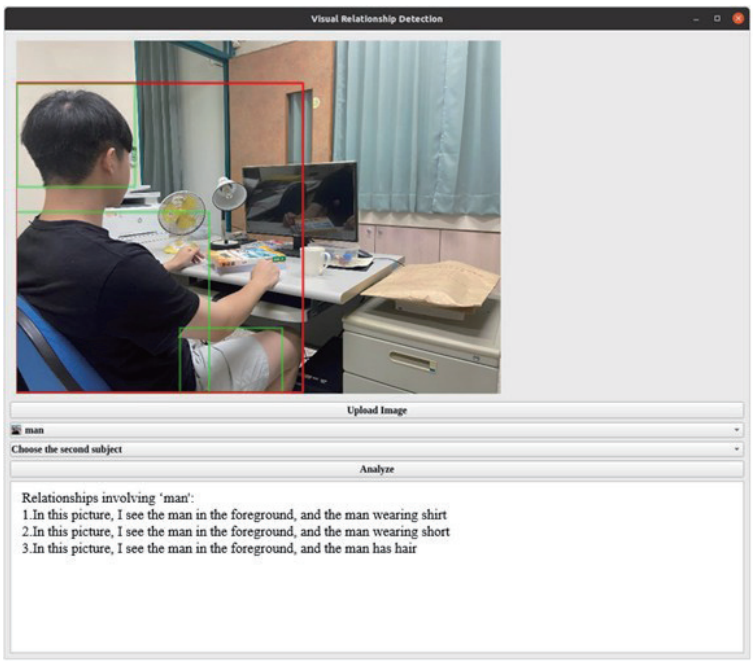

Fig. 15. (Color online) One-subject result of visual relationship detection (2nd scene).

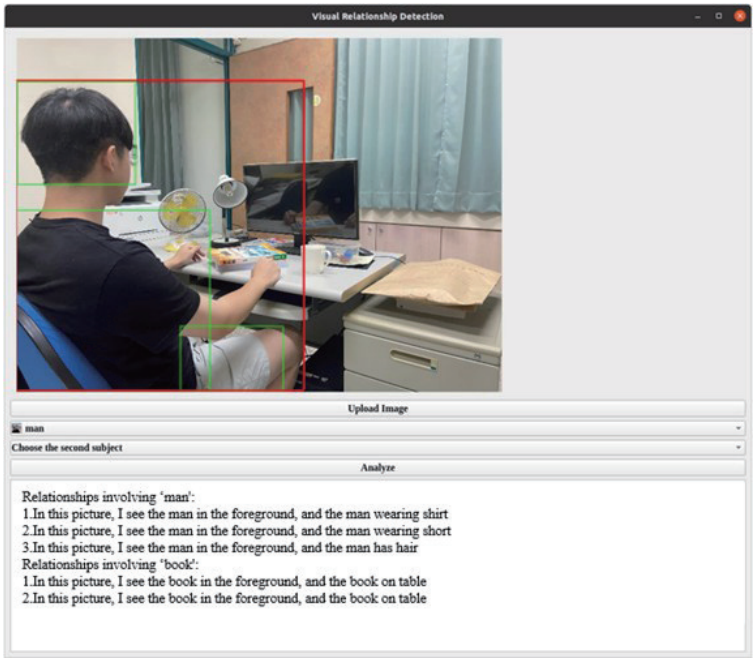

Fig. 16. (Color online) Two-subject result of visual relationship detection (2nd scene).

Table 1
Accuracy of scene graph detection.

Scene Graph Detection				
Model	R@20	R@50	mR@20	mR@50
CISC[18]	7.7	11.4	—	—
GPS-Net[19]	22.3	28.9	6.9	9.3
FCSGG[20]	16.1	21.3	2.7	3.6
SGTR[21]	—	20.6	—	15.8
Ours	19.1	23.7	5.8	8.9

The results demonstrate that our approach achieves competitive R@Call values compared with other advanced methods, showcasing strong performance across multiple metrics. Specifically, our system achieves an R@20 value of 19.1, which falls within the range of 7.7 (CISC) to 22.3 (GPS-Net). For the R@50 metric, our system obtains a value of 23.7, which is within the range of 11.4 (CISC) to 28.9 (GPS-Net). In terms of mR@20, our system attains 5.8, which is within the range of 2.7 (FCSGG) to 6.9 (GPS-Net). As for mR@50, our system achieves a value of 8.9, which is within the range of 3.6 (FCSGG) to 15.8 (SGTR). Despite not having the highest post-training R@Call value, our system excels in predictive capabilities, demonstrating its effectiveness in visual relationship detection and scene comprehension tasks. Furthermore, our system consistently demonstrates stable and reliable performance across various metrics, illustrating its efficiency and suitability for managing visual relationship detection tasks.

5. Conclusions

We utilized a novel scene analysis system featuring a crucial “subject selection” capability that allows users to select their preferred subjects for visual relationship analysis. This feature promotes a deeper understanding of the interrelationships between selected subjects and other elements, enabling users to exclude irrelevant topics and focus on analyzing specific subjects. This improves the precision and effectiveness of analysis. Moreover, the system integrates CNNs for object detection and employs a vision transformer model architecture for analyzing relationships among objects. By employing the attention maps generated by the vision transformer, users can visualize the distribution of attention among objects, enhancing their understanding of the level of attention and impact among objects. Both text-to-speech and speech recognition capabilities are integrated into the interface, allowing users to interact with the system using voice commands, significantly improving user experience and convenience. Experimental results demonstrated that the system effectively analyzes spatial layouts, behavioral interactions, and interrelationships among multiple objects within a scene.

Acknowledgments

This study was supported by the National Science and Technology Council of Taiwan under contract number NSTC 113-2221-E-167-029, for the duration from August 1, 2024 to July 31, 2025.

References

1 J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and F.-F. Li: Proc. 2015 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2015) 3668–3678. https://doi.org/10.1109/CVPR.2015.7298990
2 C. Lu, R. Krishna, M. Bernstein, and F.-F. Li: Lecture Notes in Computer Vision-ECCV 2016, B. Leibe, J. Matas, N. Sebe, and M. Welling Eds. (Springer, Cham, 2016) pp. 852–869. https://doi.org/10.48550/arXiv.1608.00187
3 J. Johnson, A. Gupta, and F.-F. Li: Proc. 2018 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2018) 1219–1228. https://doi.org/10.1109/CVPR.2015.7298990
4 K. Nguyen, S. Tripathi, B. Du, T. Guha, and T. Q. Nguyen: Proc. 2021 IEEE Computer Vision Conf. (IEEE, 2021) 1407–1416. https://doi.org/10.1109/ICCV48922.2021.00144
5 J. Johnson, B. Hariharan, L. Van Der Maaten, J. Hoffman, F.-F. Li, C. Lawrence Zitnick, and R. Girshick: Proc. 2017 IEEE Computer Vision Conf. (IEEE, 2017) 2989–2998. https://doi.org/10.1109/ICCV.2017.325
6 O. Ashual, L. Wolf: Proc. 2019 IEEE Computer Vision Conf. (IEEE, 2019) 4561–4569. https://doi.org/10.1109/ICCV.2019.00466
7 R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and F.-F. Li: Int. J. Comput. Vision **123** (2017) 32. https://doi.org/10.1007/s11263-016-0981-7
8 D. Xu, Y. Zhu, C. B. Choy, and F.-F. Li: Proc. 2017 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2017) 5410–5419. https://doi.org/10.1109/CVPR.2017.330
9 J. Ji, R. Krishna, J. C. Niebles, and F.-F. Li: Proc. 2020 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2020) 10236–10247. https://doi.org/10.1109/CVPR.2015.7298990
10 Y. W. Chao, Z. Wang, Y. He, J. Wang, and J. Deng: Proc. 2015 IEEE Computer Vision Conf. (IEEE, 2015) 1017–1025. https://doi.org/10.1109/ICCV.2015.122
11 V. Ramanathan, C. Li, J. Denge, W. Han, Z. Li, K. Gu, Y. Song, S. Bengio, C. Rosenberg, and F.-F. Li: Proc. 2015 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2015) 1100–1109. https://doi.org/10.1109/CVPR.2015.7298713
12 Y. Atzmon, J. Berant, V. Kezami, A. Globerson, and G. Chechik: eprint arXiv:1608.07639 (2016). https://doi.org/10.48550/arXiv.1608.07639
13 H. Zhou, C. Zhang, M. Zhao, Y. Luo, and C. Hu: IEEE Trans. Circuits Syst. Video Technol. **31** (2020) 2751. https://doi.org/10.1109/TCSVT.2020.3032650
14 J. Zhu and H. Wang: IEEE Trans. Cognit. Dev. Syst. **14** (2021) 752. https://doi.org/10.1109/TCDS.2021.3079278
15 H. Liu, N. Yan, M. Mortazavi, and B. Bhanu: Proc. 2021 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2021) 11546–11556. https://doi.org/10.1109/CVPR46437.2021.01138
16 A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin: Proc. 2017 Ann. Conf. Neural Information Processing Systems (NIPS, 2017) 5998–6008. https://doi.org/10.48550/arXiv.1706.03762
17 N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko: Proc. 2020 European Computer Vision Conf. (2020) 213–229. https://doi.org/10.48550/arXiv.2005.12872
18 W. Wang, R. Wang, S. Shan, and X. Chen: 2019 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2019) 8180–8189. https://doi.org/10.1109/CVPR.2019.00838
19 X. Lin, C. Ding, J. Zeng, and D. Tao: 2020 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2020) 3746–3753. https://doi.org/10.1109/CVPR42600.2020.00380
20 H. Liu, N. Yan, M. Mortazavi, and B. Bhanu: 2021 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2021) 546–556. https://doi.org/10.1109/CVPR46437.2021.01138
21 R. Li, S. Zhang, and X. He: 2022 IEEE Computer Vision and Pattern Recognition Conf. (IEEE, 2022) 486–496. https://doi.org/10.1109/CVPR52688.2022.01888