Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

FLIRL-Net: Fuzzy Logic-Driven Important Relationship Learning for Scene Graph Generation

Jin Wang
Nantong University
Zilong Yang
Nantong University
Jialing Xu
Nantong University
Sixu Wu
1179330512@qq.com

Nantong University

Research Article

Keywords: Scene Graph Generation, Important Relationships, Fuzzy Logic, Importance Weighting, Entity Pair Feature PoolFormer

Posted Date: October 31st, 2025

DOI: https://doi.org/10.21203/rs.3.rs-7500066/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Multimedia Systems on March 10th, 2026.
See the published version at https://doi.org/10.1007/s00530-026-02272-3.

FLIRL-Net: Fuzzy Logic-Driven Important Relationship Learning for Scene Graph Generation

Jin Wang[1†], Zilong Yang[1†], Jialing Xu[1†], Sixu Wu (corresponding author)[1*]

[1,2,3,4*]School of Artificial Intelligence and Computer Science, Nantong University, Nantong, 226019, Jiangsu, China.

*Corresponding author(s). E-mail(s): 1179330512@qq.com;
Contributing authors: wj@ntu.edu.cn; 2441320031@stmail.ntu.edu.cn;
1206263184@qq.com;
†These authors contributed equally to this work.

Abstract

Scene Graph Generation (SGG) is a fundamental task in computer vision. It entails classifying entities in images and predicting visual relationships among them. Existing scene graph generation research targets the identification of all possible relationships within an image. However, this approach may suffer from inefficiencies and generate redundant information, hindering the further processing of relationship data in downstream tasks. In contrast, identifying important relationships is more aligned with the practical requiements of downstream tasks. To address this issue, we introduce the Fuzzy Logic-Driven Important Relationship Learning Network (FLIRL-Net). Initially, we employ fuzzy logic to compute an importance score for each relationship. This computation accounts for factors affecting relationship significance and user requirements. Subsequently, we design the Relationship Label Loss Importance Weighting (RLIW) module. This module utilizes these importance scores as weights to adjust relationship sample losses. Such adjustments direct the model's focus towards important relationships. Thirdly, we develop the Importance-Based Entity Pair Feature PoolFormer (IEPFP) module to enhance the model's recognition of important entity pairs. Additionally, We propose metrics Important Relationship Recall (IR@K) and Important Relationship Precision (IP@K). These metrics assess the model's effectiveness in identifying important relationships. Experimental results demonstrate that our model excels at identifying important relationships in the VG150 dataset and effectively minimizes unimportant relationship outputs.

Keywords: Scene Graph Generation, Important Relationships, Fuzzy Logic, Importance Weighting,Entity Pair Feature PoolFormer

1 Introduction

Scene Graph Generation (SGG) is a frontier research topic in the field of Artificial Intelligence and a core task in computer vision. This task combines image recognition and natural language processing technologies. It aims to detect and classify all entities in an image and predict their visual relationships. Entities and relationships are described using simple nouns and verbs, respectively. Results are output as triplets in the format ⟨subject entity-relationship-object entity⟩. A scene graph is constructed by aggregating all these triplets. SGG acts as a bridge between upstream object detection tasks and downstream advanced visual understanding tasks. This includes tasks like Image Captioning [1–3], Image Retrieval [4, 5], and Video Captioning [6, 7].

Two-stage scene graph generation models typically begin by using an object detector to identify entities within an image [8–16]. To reduce bias in predicting head and tail predicates, these models adopt debiasing strategies like distribution normalization [17–19], predicate resampling [20, 21], and semantic information enhancement [10, 22–29]. While they aim to predict as many relationships as possible to boost recall rates, this approach often leads to inefficiency and redundant relationships that can hinder downstream tasks analyzing relationship data. This reveals a key issue: current models struggle to effectively identify and prioritize important relationships, which are critical for understanding the main content of the image.

A key challenge for scene graph generation models is the inability to differentiate between critical and non-essential relationships in annotated datasets. The VG150 dataset [30], widely used for scene graph tasks, provides extensive annotations of entities, bounding boxes, attributes, and relationships. However, current methods fail to effectively distinguish the importance of each relationship. As shown in Fig. 1, important relationships like ⟨man-riding-bike⟩ and ⟨man-near-train⟩ are vital for understanding the image's main content. In contrast, non-essential relationships often describe background entities or less relevant interactions. Yet, existing methods treat all relationships equally, leading to scene graphs cluttered with redundant relationships. This results in less focus on critical interactions, as irrelevant or background relationships may obscure the key content during downstream tasks like description or captioning, thus reducing overall efficiency.

Researchers are increasingly exploring Salient SGG methods to identify important relationships in images. The VG-KR dataset [31], annotated for important relationships based on factors like caption content and entity size, isolates important relationships in Visual Genome. However, scene graph applications span various fields, each prioritizing different visual elements.For instance, urban traffic studies focus on vehicles and traffic lights, while social behavior studies emphasize interpersonal interactions. Thus, training models solely on VG-KR may fail to capture the diversity of key relationships across tasks. Tailoring scene graph generation to downstream tasks improves parsing accuracy, but assessing relationship importance remains challenging.

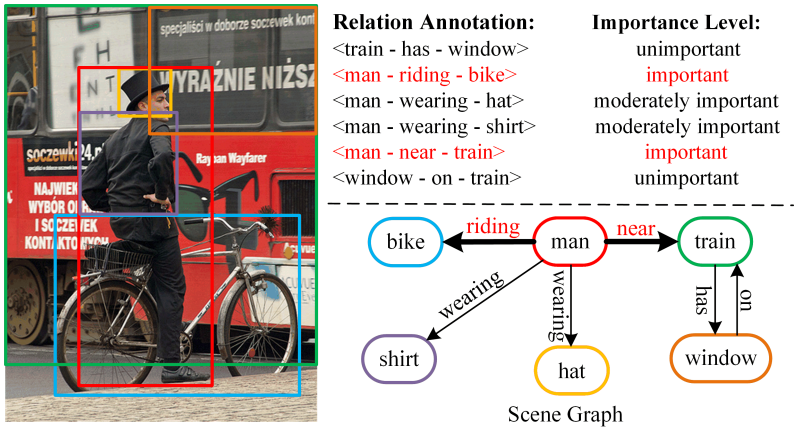


Fig. 1 Left: Image from the VG150 dataset with annotated bounding boxes. Top right: Annotated relationships and the importance of these relationships for understanding the main content of the image. Bottom right: Output scene graph

To address the challenge of calculating relationship importance, we use a fuzzy inference system to assess relationship importance based on subject and object features, attributes, and downstream task requirements. By integrating fuzzy rules and defuzzification, we assign importance scores flexibly and comprehensively. We propose the IRL method to optimize important relationship prediction. Unlike traditional models that focus on tail predicates with high semantic value, our method prioritizes key entity interactions and their overall context, ensuring the scene graph reflects the most relevant relationships for the task at hand. It uses relationship importance scores to weight the loss of relationships, thus sharpening the model's focus on important relationships. To assess performance, we propose two metrics: Important Relationship Recall (IR@K) and Important Relationship Precision (IP@K), which filter predictions to retain only important relationships in the scene graph. Importance scores filter the predicted relationships, ensuring the scene graph retains only those deemed important.

In summary, the main contributions of our work are three folds:

- We propose a strategy for calculating relationship importance scores using fuzzy logic applied to the VG150 dataset. Users can flexibly assign importance scores to each relationship based on specific requiements and established fuzzy rules;
- We propose the Important Relationship Learning method, which trains the model to focus on relationships with higher importance scores. It significantly improves the model's ability to identify important relationships;
- We propose two metrics IR@K and IP@K, to evaluate model's ability to predict important relationships. We validate the effectiveness of the IRL method on multiple baseline models. The results demonstrate that FLIRL-Net excels in predicting important relationships.

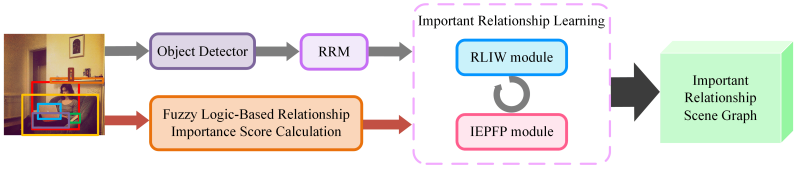


Fig. 2 Overview of the FLIRL-Net framework. RRM is the Relationship Representations Modeling module, RLIW is the Relationship Label Loss Importance Weighting module, and IEPFP is the Importance-based Entity Pair Feature PoolFormer. In the process of generating the scene graph, by setting a threshold, relationships with an importance score greater than the threshold are output to generate an important relationship scene graph

2 Related Work

The extensive annotations of entity and predicate classes, along with bounding box data in the VG150 dataset, have significantly advanced two-stage scene graph generation. In the first stage, an object detector extracts entity proposals, and in the second stage, these features are used to predict relationships. We categorize related work into traditional scene graph generation models and salient scene graph generation models.

2.1 Traditional Scene Graph Generation Model

In 2016, Lu et al. [32] introduced SGG using the RCNN method, highlighting its importance in image understanding. Traditional SGG models refine relationship prediction using contextual information. Xu et al. [30] demonstrated that contextual information improves relationship prediction through IMP mechanism. Zellers et al. [33] proposed the Motifs baseline, which predicts frequent entity relationships and employs Stacked Motif Networks to analyze entity and relationship distributions. Yang et al. [34] developed the Graph R-CNN model, integrating the RePN module for processing potential relationships and the aGCN module for capturing contextual information between objects and relationships.

As research progresses, imbalanced data distributions have been identified as a challenge for SGG models in recognizing fine-grained tail predicates. To address this, researchers have developed unbiased SGG frameworks. Tang et al. [35] introduced a causal inference-based framework using a causal graph and counterfactual relationships to reduce bias, proposing TDE module for unbiased SGG. Zheng et al. [29] addressed prediction biases from intra-class variance and inter-class similarity with PE-Net method, which uses prototypes for compact entity and predicate representations. Chang et al. [23] introduced the LANDMARK framework, which uses three methods: the channel-attention language attention module, the Transformer-based language context module, and the marginal probability-based estimation module. This approach enables multi-semantic encoding and resolves the underutilization of language in relation prediction. However, traditional SGG models often generate irrelevant relationships, affecting downstream tasks. Consequently, the field is shifting towards Salient SGG models, which focus on key visual relationships to improve downstream applications.

2.2 Salient Scene Graph Generation Model

Salient SGG models aim to accurately capture salient relationships in images to ensure downstream tasks process the most relevant and significant information. Salient relationships also known as important relationships or key relationships. Existing Salient SGG models focus on two main approaches: algorithmic enhancement of visual relationship representation, and external resource-guided scene graph generation.

Algorithmic enhancement of visual relationship representation typically uses strategies like attention mechanisms or message passing to improve the identification of important relationships. Lv et al. [36] used attention mechanisms to prioritize relationships by integrating local and global contexts. Zhang et al. [24] proposed method SMP to model the perceptual salience of relationships in scene graphs, using an object interaction encoder to enhance representations by integrating appearance, semantics, and spatial data. Lu et al. [37] introduced a bipartite graph network with dual-group message passing to enhance relationship inference. While this approach improves the model's ability to recognize important relationships, it relies heavily on complex internal structures and parameter tuning. It overlooks the diversity of real-world scenarios, limiting generalization to new scenes.

External resource-guided scene graph generation uses text descriptions or dataset annotations to build a broader knowledge framework, improving the representation of important relationships. Unlike algorithmic approaches, this method incorporates semantic layers from external sources to enhance SGG's ability to handle complex content. Wang et al. [31] developed the VG-KR dataset, annotated for key relationships and employed a hierarchical entity tree and the Hybrid-LSTM module to capture structures and contexts. Benetatos et al. [38] proposed generating salient scene graphs via weak supervision with image-text descriptions, learning entities and relationships from image-text pairs. Although leveraging external resources improves the accuracy of relationship descriptions, it may neglect varying emphases on entities and relationships across different downstream tasks. The quality and scope of text descriptions and dataset annotations directly affect the scene graphs applicability.

Our FLIRL-Net method uses fuzzy logic to compute continuous importance scores (0-1) for relationships, providing more nuanced quantification than binary classification. This enables the model to meet downstream task requirements and enhance the applicability of scene graphs across domains. For example, in social behavior research, FLIRL-Net identifies key social interactions in complex images. The scores guide model training, enhancing its ability to learn significant relationships and offering a precise, practical solution for salient scene graph generation.

3 Methodology

The entire process of our Fuzzy Logic-Driven Important Relationship Learning Network (FLIRL-Net) is illustrated in Fig. 2. Initially, a pre-trained object detector extracts labels and bounding boxes for each entity in the input image. Subsequently, the Relationship Representation Modeling (RRM) model refines entity and bounding box features, combining them to form relationship representations. Next, we propose the Fuzzy Logic-Based Relationship Importance Score Calculation (FLC) module. We

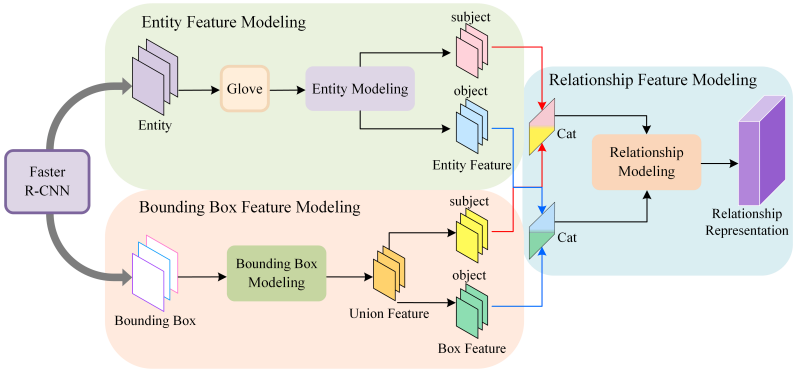


Fig. 3 Network structure of Relationship Representations Modeling. 'Cat' refers to concatenating two different representations of features. The concatenated features are then fed into the network layer to match the original feature size

preprocess the VG150 dataset and apply fuzzy logic to calculate relationship importance scores. Thirdly, we design the Important Relationship Learning (IRL) method, which trains the model to focus on relationships with higher importance scores. The Relationship Label Loss Importance Weighting (RLIW) module applies importance scores to weight the loss of relationship label predictions, guiding the model towards more important relationships. Simultaneously, the Importance-Based Entity Pair Feature PoolFormer (IEPFP) module enhances the representation of important entity pairs by integrating entity pair features with relationship importance scores via the network Feature PoolFormer. Finally, the model filters the predicted relationships based on their importance scores to generate a scene graph of important relationships.

3.1 Problem formulation

The goal of scene graph generation is to parse a given image I into a scene graph $G = \{E, R\}$, where $E = \{e_i\}_{i=1}^{N}$ represents a set of entity category labels detected from the image, and $R = \{r_k\}_{k=1}^{M}$ denotes a set of M relationships between pairs of these entities. The process begins with generating a set of N bounding boxes $B = \{b_i\}_{i=1}^{N}$ from the input image I, identifying the entity e_i within each bounding box, and then predicting the relationships r_k between pairs of entities to form triplets (e_i, r_k, e_j). These triplets are then assembled to create the scene graph G. The construction of scene graph G can be expressed by the following joint probability distribution:

$$\mathrm{P}(G \mid I) = \mathrm{P}(B \mid I)\mathrm{P}(E \mid B, I)\mathrm{P}(R \mid E, B, I), \tag{1}$$

where $\mathrm{P}(B|I)$ is the probability of generating bounding boxes from the image, $\mathrm{P}(E \mid B, I)$ is the probability of recognizing entities given the bounding boxes and the image, $\mathrm{P}(R \mid E, B, I)$ calculates the probability of predicting relationships based on the entities, bounding boxes, and the image.

In the above scene-graph generation process, R denotes all detected instance relations in the image. However, in real-world applications (e.g. image retrieval, visual question answering, automatic summarisation), not every relation in R is equally

important: some relations convey core semantic information or are more relevant to the task objective, whereas others belong to redundant background. To enable downstream modules to concentrate on the semantic key parts, we give a unified, operable definition of *important relations* (IR) as follows.

Given an image I with scene graph $G = \{E, R\}$ and the candidate triple set $\mathcal{T} = \{(s_i, p_i, o_i)\}_{i=1}^{N}$, a triple is regarded as an important relation (IR) *iff* at least one of the following conditions holds:

- Task relevance: (s_i or p_i or o_i) $\in C_{\text{task}}$, where C_{task} is the set of categories specified by the downstream task.
- Visual saliency: the bounding-box area of (s_i or o_i) occupies more than $\tau_{\text{area}} = 8\%$ of the image area, and its centre-to-image-centre normalised distance is less than $\tau_{\text{center}} = 0.2$.
- Information value: the global occurrence frequency of predicate p_i lies within the bottom quartile of the dataset, targeting the long-tail problem in scene-graph generation, where rare predicates often carry higher information value.

If none of the three criteria is satisfied, the triple is labelled as a non-important relation. The above thresholds are tuned on the validation set by five-fold cross-validation.

3.2 Relationship Representations Modeling

In Fig. 3, the SGG method uses a pre-trained object detector Faster R-CNN [39] to obtain category labels and bounding boxes for each entity from image I. First, the Glove [40] model is used to generate embeddings for the entity category labels e. Then, a semantic extractor is employed to extract the subject and object features f_s, f_o:

$$\begin{aligned} f_s &= w_s^T c_s \\ f_o &= w_o^T c_o \end{aligned}, \tag{2}$$

where w_s, w_o is the learnable weight parameters for the subject and object features, T indicates matrix transposition. c_s and c_o refer to the word embeddings for the subject and object classes, respectively. The subject and object features are further refined and modeled to generate detailed features $\hat{f}_s$ and $\hat{f}_o$:

$$\hat{f}_s = \text{LN}(\text{ReLU}(\text{FC}(f_s)) + f_s), \tag{3}$$

where $\text{FC}(\cdot)$, $\text{LN}(\cdot)$, and $\text{ReLU}(\cdot)$ represent a fully connected layer, layer normalization, and ReLU activation function, respectively. Then, a residual connection is used to enhance the expressiveness of the entity features. The features $\hat{f}_o$ are derived in the same manner. Additionally, entities have their own bounding boxes, requiring the generation of bounding box feature for each entity.

First, concatenate the coordinates of the top-left b^l, the bottom-right b^r, and the area b^a of the bounding box. Then, model these to generate the bounding box embedding. The subject bounding box embedding $\hat{b}_s$ is shown as follows:

$$\hat{b_s} = \text{LReLU}(\text{FC}([b_s^l, b_s^r, b_s^a])), \tag{4}$$

where b_s^l, b_s^r, and b_s^a are the coordinates of the top-left, bottom-right, and the area of the subject bounding box respectively, $[\cdot,\cdot]$ represents column concatenation, and $\mathrm{LReLU}(\cdot)$ is the Leaky ReLU activation function. The object bounding box embedding $\hat{b}_o$ is derived in the same manner.

To capture the positional information of relationships, a joint subject-object bounding box embedding is further designed. Take the top-left coordinates from the leftmost bounding box and the bottom-right coordinates from the rightmost bounding box to represent the coordinates of the joint bounding box. The area of the joint bounding box is then calculated using new coordinates. The joint bounding box embedding $\hat{b}_{so}$ is derived in the same manner following Eq. 4.

The bounding box features of entities should reflect their relative positional information within the entity pair. Therefore, we concatenate the entity bounding box embedding with the joint bounding box embedding and model them to generate the entity bounding box features. The subject bounding box feature $\tilde{b}_s$ is shown as follows, and the object bounding box feature $\tilde{b}_o$ is derived in the same manner:

$$\tilde{b_s} = \mathrm{LReLU}(\mathrm{FC}([\hat{b_s}, \hat{b_{so}}])). \tag{5}$$

We add the bounding box feature to the entity feature to obtain the entity representation:

$$s = \mathrm{LN}(\mathrm{ReLU}(\mathrm{FC}([\hat{f}_s, \tilde{b}_s]))), \tag{6}$$

where s represents the subject representation, $[\hat{f}_s, \tilde{b}_s]$ is the concatenation of the subject feature and the subject bounding box feature. The object representation o is derived in the same manner.

To predict relationships, we fuse the representations of the subject and object to obtain the relationship feature $F(s, o)$:

$$F(s, o) = \mathrm{ReLU}(s + o) - (s - o)^2. \tag{7}$$

To further enhance the expressiveness of the relationship feature, we concatenate it with the previously generated joint bounding box embedding $\hat{b}_{so}$ to form the relationship representation $\tilde{r}$:

$$\tilde{r} = \mathrm{MLP}(\mathrm{LN}(\mathrm{ReLU}(\mathrm{FC}([F(s, o), \hat{b}_{so}])))), \tag{8}$$

where MLP represents a multilayer perceptron [41] . It captures essential features through multiple nonlinear transformations, providing the model with rich feature representations.

3.3 Fuzzy Logic-Based Relationship Importance Score Calculation

Fuzzy logic is a system that handles uncertainty and imprecision by extending binary logic to allow variables to assume any continuous value between extremes like ‘true’ and ‘false’. Fuzzy logic excels at addressing the vague boundaries common in natural language and human cognition. [42]

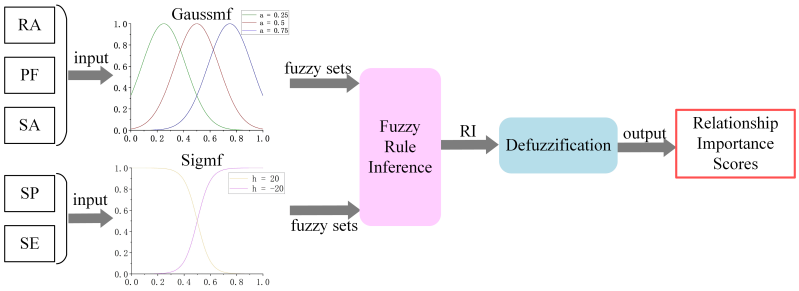


Fig. 4 Process of inferring relationship importance scores using fuzzy logic. In the Gaussian function, the x-axis values corresponding to the peaks from left to right are 0.25, 0.5, and 0.75. In the Sigmoid function, the two curves are decreasing and increasing based on the positive and negative steepness h. This is determined according to our predefined fuzzy set partitions

Fuzzy logic, renowned for its ability to handle uncertainty and imprecision, is widely applied in computer vision. Its effectiveness is particularly notable in tasks like Medical Image Segmentation and Edge Detection. Ding et al. [43] developed the method FTransCNN for medical image segmentation, combining networks CNNs and Transformers for feature extraction. This system utilizes choquet fuzzy integrals to suppress heterogeneity and the module FAFM to capture both low-level spatial features and high-level semantic contexts effectively. Ma et al. [44] introduced a deep image feature learning method using fuzzy rules, combining fuzzy modeling and deep stacked learning strategies to progressively learn image features. Bhuyan et al. [45] proposed an RGB-D fusion framework utilizing zero-shot learning, employing a fuzzy member learning network to generate saliency maps. Sumit et al. [46] used fuzzy logic to manage ambiguity and complexity in skin images, adjusting input and output thresholds to precisely localize and categorize lesions. Nachar et al. [47] proposed using fuzzy logic to fine-tune and confirm edge locations following initial feature learning and edge detection. Jawdekar et al. [48] enhanced image visibility and edge clarity with a fuzzy logic system that handles uncertainties and ambiguities, enabling more accurate recognition and analysis of medical images by doctors and diagnostic systems. These examples underscore the substantial value of fuzzy logic in computer vision, solidifying the foundation for our research.

In SGG, each relationship consists of features from subject and object entities, each accompanied by various attributes that collectively determine the relationship's importance. Traditional methods typically categorize relationship importance into binary extremes: 'important' or 'unimportant'. This binary classification neglects the nuanced differences between relationships. By employing fuzzy logic, we can compute a continuous importance score for each relationship, ranging from 0 to 1. This approach offers a more refined measure of importance, enabling users to adjust the evaluation flexibly based on real requiements, aligning better with the complexities of real-world scenarios.

In this module, we propose a fuzzy logic-based method to calculate relationship importance scores. Initially, we process five inputs: relationship region area (RA), predicate frequency (PF), specified entity class (SE), specified predicate class (SP), and specified attribute class (SA). These inputs are fed into a fuzzy inference system that

employs Gaussian and Sigmoid membership functions for fuzzification. Subsequently, we apply preset fuzzy rules to generate outputs reflecting relationship importance. Finally, we employ the centroid method to defuzzify and quantify the relationship importance scores, which comprehensively reflect the significance of relationships. The fuzzy inference process is illustrated in Fig. 4. This method can be flexibly adjusted to meet specific user requiements, offering precise and interpretable evaluations of relationship importance.

3.3.1 Data Processing

Numerous factors influence the importance of each relationship. In the human visual system, larger entities probably attract more attention, thereby making relationships involving larger entities more important. The frequency of predicate categories in the dataset is also crucial. Some predicates may appear only in specific contexts or a few images, making them essential for describing image content and expressing complex relationships. Therefore, low-frequency relationships warrant more attention. Additionally, users prioritize different relationships based on their goals and interests. Specifying entity classes, predicate classes, and attribute classes makes the calculation of relationship importance scores more targeted and flexible. This approach can be adjusted to reflect the specific importance of relationships relevant to users' interests.

The specific data processing steps are as follows:

RA: we calculate the area of each entity bounding box based on the dataset's recorded information. To eliminate the influence of varying image sizes, we normalize the entity areas within each image. We then use the weighted average of the areas of the subject and object as RA.

PF: some predicates appear tens of thousands of times in the dataset, while others only a few hundred times. We normalize predicate frequencies to a range of 0-1 to prevent high-frequency predicates from disproportionately affecting the results.

SE, SP: we use a binary classification method. If the subject or object in a relationship belongs to specified entity classes, we mark SE as 1; otherwise, we mark it as 0. The same binary classification method applies to SP.

SA: each entity may have one or more attributes. Entities with more attributes align better with users' expectations. We determine SA by calculating the total number of specified attributes contained by the subject and object.

In the fuzzy inference system, precise input data need to be fuzzified. The fuzzification process maps these precise values to fuzzy sets, each defined by a membership function.

3.3.2 Fuzzy Set Partitioning

The input parameters for the fuzzy inference system include RA, PF, SE, SP and SA. The output parameter is the relationship importance (RI). We define the fuzzy sets for each parameter:

The fuzzy set for RA : {large area (la), medium area (ma), small area (sa)};

The fuzzy set for PF: {high frequency (hf), medium frequency (mf), low frequency (lf)};

The fuzzy set for SE: {included (in), not included (ni)};

The fuzzy set for SP: {belongs to (be), not belong to (nb)};

The fuzzy set for SA: {many attributes (mt), medium attributes (md), few attributes (fa)};

The fuzzy set for RI: {important (hi), moderately important (mi), unimportant (ui)}.

3.3.3 Membership Function Definition

Based on the division of fuzzy sets and the range of input and output parameters, we use appropriate membership functions to convert input parameters into fuzzy values. These values are mapped to the corresponding fuzzy sets. We also calculate the membership degree of each element within the fuzzy set for the output parameter. This allows the system to defuzzify and output the final result.

For RA, PF, SA, and RI, the fuzzy sets exhibit different classification levels. Within their respective domains, these parameters need appropriate membership levels. To ensure smooth transitions between fuzzy set states, we use Gaussian membership functions (Gaussmf):

$$G\text{aussmf}(X_1; a, \beta) = e^{-\frac{(X_1-a)^2}{2\beta^2}}, \tag{9}$$

where $X_1 \in \{RA, PF, SA, RI\}$, a is a hyperparameter that reflects the peak value of the Gaussian function, and β is a hyperparameter that represents the decay rate from the peak. Both a and β are determined based on the value ranges of each parameter.

For SE and SP, the fuzzy sets represent the two extreme states of 'yes' and 'no'. The Sigmoid membership function (Sigmf), characterized by a continuous curve, provides a smooth transition between these two states:

$$\text{Sigmf}(X_2; h, \varepsilon) = \frac{1}{1 + e^{-h(X_2-\varepsilon)}}, \tag{10}$$

where $X_2 \in \{SE, SP\}$, ε and h are hyperparameters that indicate the midpoint and steepness of the Sigmoid function, respectively. The larger the value of h, the faster the transition near ε. Both ε and h are determined based on the value ranges of each parameter.

3.3.4 Fuzzy Rule Definition

Formulating fuzzy rules is the core step of a fuzzy inference system; their scientific soundness and rationality directly determine the accuracy of relation–importance scores. On the basis of three fundamental theories, we construct the fuzzy rules: (1) *visual-attention theory*, which holds that the human visual system prioritises instances and relations that occupy a large area or are spatially salient; (2) *information-value theory*, which posits that relations occurring at low frequency usually convey more distinctive information; and (3) *task-oriented theory*, noting that downstream tasks differ markedly in their demands for relation types.

To guarantee scientific rigour and reproducibility, a systematic formulation procedure is adopted. First, the importance of relations in 500 randomly sampled images from the VG150 data set is manually annotated to establish a reference standard. Second, statistical analysis verifies the correlation between the annotations and

the influencing factors, thereby guiding the allocation of rule weights. Finally, several rounds of cross-validation adjust the rule parameters, ensuring robustness and generalisability.

Building on the above theoretical basis and empirical analysis, we devise a three-layer hierarchical rule set that captures the complexity of relation importance. The first layer consists of task-oriented rules, which receive the highest priority and determine relation importance according to the user-specified entity class, predicate class, and attribute class:

(1) if $(SO \in IN) \cap (SP \in BE) \cap (SA \in MA)$, Then $RI \in HI$ [weight = 0.9]

(2) if $(SO \in IN) \cap (SP \in BE) \cap (SA \in MD)$, Then $RI \in HI$ [weight = 0.8]

(3) if $(SO \in IN) \cup (SP \in BE)$, Then $RI \in MI$ [weight = 0.6]

The second layer contains visual-saliency rules, with medium priority; it judges mainly on the basis of relation-region area and predicate frequency:

(4) if $(RA \in LR) \cap (PF \in LF)$, Then $RI \in HI$ [weight = 0.85]

(5) if $(RA \in LR) \cap (PF \in MF)$, Then $RI \in MI$ [weight = 0.7]

(6) if $(RA \in MR) \cap (PF \in LF)$, Then $RI \in MI$ [weight = 0.65]

The third layer comprises basic-exclusion rules, with the lowest priority, used to identify relations that are clearly unimportant:

(7) if $(SO \in NI) \cap (SP \in NB) \cap (SA \in FA) \cap (RA \in SR)$, Then $RI \in LI$ [weight = 0.3]

(8) if $(PF \in HF) \cap (RA \in SR) \cap (SA \in FA)$, Then $RI \in LI$ [weight = 0.4]

A default rule is finally set to handle cases not covered by the above rules:

(9) if NOT($R1$–$R8$), Then $RI \in MI$ [weight = 0.5]

When multiple rules are activated simultaneously, an effective conflict-resolution mechanism is required. This paper adopts a weighted-voting mechanism to resolve rule conflicts: first, rules are processed in the priority order task-oriented → visual-saliency → basic-exclusion; then the confidence of each rule is calculated as the product of its weight w_i and the membership degree μ_i of its antecedent conditions; finally, a weighted-average method fuses the outputs of all activated rules. The specific calculation is:

$$RI_{\text{final}} = \frac{\sum_{i=1}^{n} w_i \, \mu_i \, RI_i}{\sum_{i=1}^{n} w_i \, \mu_i}, \tag{11}$$

where w_i is the weight of the i-th rule, μ_i is the product of the membership degrees of its antecedent conditions, and RI_i is the output of that rule. To enhance the adaptability of our method, we designed a configurable rule-extension mechanism. For different research domains, specialised rule sets can be added. For example, in social-behaviour studies, the rule "if $(SO \in \{$*person, man, woman*$\}) \wedge (SP \in \{$*interacting, talking, hugging*$\})$, then $RI \in HI$" can be included; in traffic-scene analysis, one may add "if $(SO \in \{$*car, bus, truck*$\}) \wedge (SP \in \{$*driving, parked_on*$\})$, then $RI \in HI$." This extension mechanism enables the fuzzy-rule system to satisfy the specific requirements of different domains.

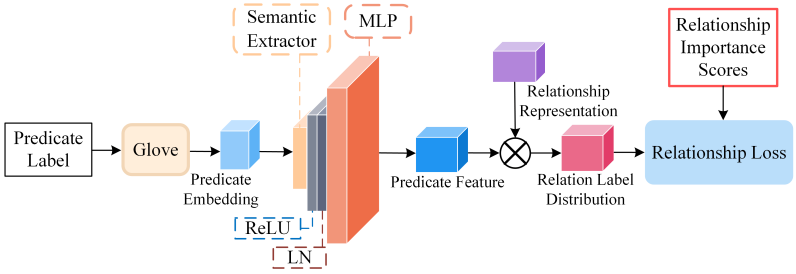


Fig. 5 Network structure of the Relationship Label Loss Importance Weighting module. ⊗ represents matrix multiplication

3.3.5 Fuzzy Inference and Defuzzification

The fuzzy inference process employs the Mamdani inference algorithm. First, it fuzzifies the input using defined membership functions and maps them to the corresponding fuzzy sets. Then, it applies fuzzy rules to generate a fuzzy output for each rule. The system aggregates these outputs to create a combined fuzzy output parameter RI, reflecting the relationship importance. Finally, we convert the fuzzy sets of RI into specific numerical values through defuzzification, resulting in the relationship importance scores (R_{im}). We use the centroid method for defuzzification, which generates smoother outputs by calculating the centroid of the fuzzy sets:

$$R_{im} = \frac{\int \alpha \mu_{RI}(\alpha) d\alpha}{\int \mu_{RI}(\alpha) d\alpha}, \tag{12}$$

where $\alpha \in \{HI, MI, LI\}_{RI}$ represents subsets within the fuzzy set of RI. μ_{RI} is the Gaussian function used for RI. $\mu_{RI}(\alpha)$ is the membership degree of each subset within RI. $R_{im} \in [0, 1]$ indicates the relationship importance scores in the dataset.

3.4 Important Relationship Learning

We design the Relationship Label Loss Importance Weighting (RLIW) module and the Importance-Based Entity Pair Feature PoolFormer (IEPFP) module. RLIW uses relationship importance scores as weights to adjust the loss for each relationship sample, directing the model's focus towards more important relationships. The network structure of RLIW is illustrated in Fig. 5. IEPFP utilizes Feature PoolFormer to merge entity pair features with relationship importance scores, enhancing the representation of important relationship entity pairs. It also employs a Euclidean distance constraint to ensure the features of predicted and actual relationship entity pairs are closely aligned. This further improves the model's ability to recognize important entity pairs. The network structure of IEPFP is illustrated in Fig. 6.

3.4.1 Relationship Label Loss Importance Weighting

To match predicted relationship labels with actual relationship labels, we calculate the label distribution of the relationship representation $\tilde{r}$. We use the GloVe [40] model

to generate predicate embeddings c_p for the predicate labels and employ a semantic extractor to extract predicate features f_p:

$$f_p = {w_p}^T c_p, \tag{13}$$

where w_p is the learnable weight parameter for the predicate feature, T indicates matrix transposition. The predicate feature is further refined to generate detailed feature $\tilde{p}$:

$$\tilde{p} = \text{MLP}(\text{LN}(\text{ReLU}(f_p))). \tag{14}$$

We normalize the relationship representation $\tilde{r}$ and the predicate features $\tilde{p}$, then calculate the label distribution for the predicted relationships:

$$r_p = (\frac{\tilde{r}}{\|\tilde{r}\|} \odot \frac{\tilde{p}}{\|\tilde{p}\|}) \exp(\gamma), \tag{15}$$

where $\|\cdot\|$ represents the L2 norm, $\odot$ is the dot product operation, exp is the exponential operation, and γ is a learnable scaling factor used to adjust the range of the relationship label distribution.

To optimize relationship prediction, we design a weighted cross-entropy loss function. The core idea is to weight each relationship sample by its importance score, making the model pay more attention to important relationships. Specifically, we use the importance scores R_{im} as weights and multiply them with the standard cross-entropy loss. This approach ensures that the loss from important relationship samples contributes more to the total loss:

$$L_{rel} = \frac{1}{N} \sum_{j=1}^{N} [-r_{t_j} \log(r_{p_j})] \times (1 + R_{im_j}), \tag{16}$$

Where r_{t_j} and r_{p_j} represent the actual label and predicted distribution of the i-th relationship sample, respectively. R_{im_j} is the importance score of the j-th relationship sample. N is the total number of relationship samples in each image.

3.4.2 Importance-based Entity Pair Feature PoolFormer

Each relationship entity pair label $[e_s, e_o]$ is composed by concatenating the subject label e_s and the object label e_o. To enhance the model's ability to recognize relationship entity pairs, we match the predicted relationship entity pair label $[e_s, e_o]_p$ with the actual relationship entity pair label $[e_s, e_o]_t$. First, we model the predicted relationship entity pair label $[e_s, e_o]_p$ to generate entity pair features E_p:

$$E_p = \text{BN}(\text{ReLU}(\text{FC}([e_s, e_o]_p))) \ , \tag{17}$$

where BN represents the batch normalization operation. The actual relationship entity pair features E_t are derived in the same manner.

Compared to Transformers [49] , which rely on complex self-attention mechanisms to capture features, PoolFormer [50] achieves similar feature extraction using pooling operations. It significantly reduces computational complexity. Our IEPFP module is an improved version of the original PoolFormer model. It processes features of actual

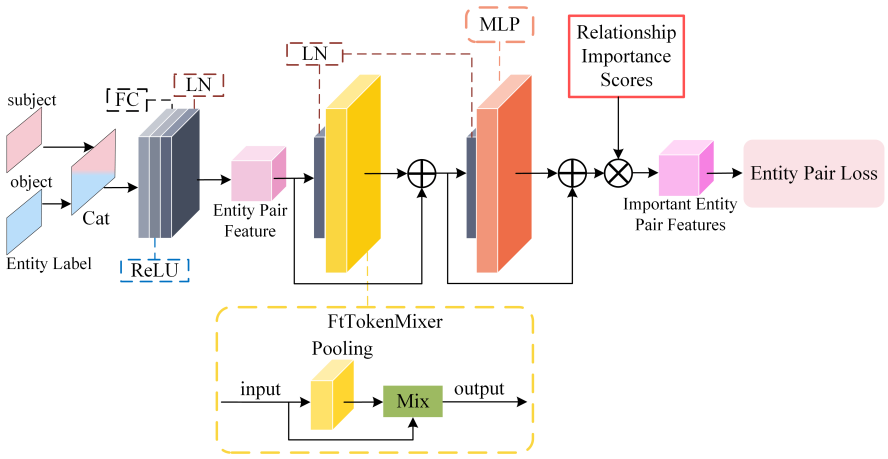


Fig. 6 Network structure of Importance-based Entity Pair Feature PoolFormer module. The Feature Pooling Token Mixer is the most important step in PoolFormer. Feature mixing subtracts the pooled feature from the original feature. This aims to highlight detailed features and reduce redundant features. It improves the model's sensitivity to details. $\oplus$ represents residual connection, and $\otimes$ represents matrix multiplication

relationship entity pairs and introduces relationship importance scores in each block. It allows the model to effectively capture important relationship entity pair information.

IEPFP includes a Feature Pooling Token Mixer, MLP and Importance Weighting. The Feature Pooling Token Mixer pools the normalized entity pair features to capture local feature variations. It retains the original feature information through residual connections,

$$E'_t = \text{FtTokenMixer}(\text{LN}(E_t)) + E_t, \tag{18}$$

where E'_t represents the actual entity pair features that capture local feature variations. The specific operation of the Feature Pooling Token Mixer FtTokenMixer is as follows:

$$T'_{:,i} = \frac{1}{M} \sum_{\varphi=1}^{M} T_{:,i+\varphi-\frac{M+1}{2}} - T_{:,i}, \tag{19}$$

where $T_{:,i}$ represents the original input features at position i, $T'_{:,i}$ represents the output features at position i after the pooling token mixing operation, and M is the size of the pooling window.

Applying MLP to the features E'_t enhances the representation capability of the entity pair features. On this basis, we use the relationship importance scores R_{im_j} for weighting to generate the actual entity pair feature representation $\bar{E}'_t$, capturing important entity pair information:

$$\overline{E}'_t = (\text{GeLU}(\text{LN}(E'_t)w_1)w_2 + E'_t) \times (1 + R_{im}), \tag{20}$$

where GeLU is the activation function, w_1 and w_2 are the learnable weight parameters. Each block in the IEPFP will sequentially perform the operations of these three

components, ultimately generating the important entity pair features $\tilde{E}_t$.We calculate the Euclidean distance set $D = \{d_j\}_{j=0}^{N}$ between E_p and $\tilde{E}_t$, where d_j is computed as:

$$d_j = \left\| E_{p_j} - \tilde{E}_{t_j} \right\|_2^2. \tag{21}$$

We then calculate the mean ν and standard deviation σ of the distance set D to obtain the normalized distance set $D' = \{d'_j\}_{j=0}^{N}$, where d'_j is computed as:

$$d'_j = \frac{d_j - \nu}{\sigma}. \tag{22}$$

To avoid the impact of extreme values in D' on the calculation of distance loss, we sort D' in descending order and trim the extreme values. The relationship entity pair distance loss L_{pair} is then calculated by averaging the absolute values of the trimmed distances:

$$L_{pair} = \frac{1}{N - 2\eta} \sum_{j=\eta+1}^{N-\eta} |d'_j|, \tag{23}$$

where $\eta = \lfloor \delta \cdot N \rfloor$ represents the number of trimmed values from both ends of N sorted distances, δ is the hyperparameter indicating the trimming proportion.

3.5 Scene Graph Prediction

During the training stage, the total training loss of our Fuzzy Logic-Driven Important Relationship Learning Network is computed as:

$$L = L_{rel} + L_{pair} + L_{\text{obj}}, \tag{24}$$

where $L_{\text{obj}} = \text{CE}(e_p, e_t)$ is the cross-entropy loss matching the predicted entity label e_p with the actual entity label e_t.

During the testing stage, we rank the predicted relationships in descending order based on their importance scores. We select and output relationships with importance scores greater than 0.5. This ensures the generated scene graph includes only key relationship triplets. This step effectively filters out unimportant relationships. It retains only the most crucial information for researchers' specific requiements, enhancing the efficacy of SGG.

4 Experimental Analysis

4.1 Datasets

To evaluate the performance of the proposed network model, we conducted training and evaluation on the publicly available Visual Genome 150 (VG150) [30] dataset. VG150 is a subset of the Visual Genome dataset and is the most commonly used dataset for SGG. It contains extensive annotations of entities and relationships. The VG150 dataset consists of 108,077 images, encompassing the 150 most common entity categories, 50 predicate categories, and 200 attribute categories. On average, each

Table 1 Comparison results of FLIRL-Net and several state-of-the-art SGG methods on the VG150 dataset.

Model	PredCls		SGCls		SGDet	
	R@50	R@100	R@50	R@100	R@50	R@100
IMP [30]	61.1	63.1	37.4	38.3	23.6	28.7
Transformer [15]	65.6	67.2	39.3	40.5	26.9	31.3
G-RCNN [34]	65.4	67.2	37	38.5	29.7	32.8
Motifs [33]	65.3	67.2	38.9	39.8	32.1	36.8
KERN [51]	65.8	67.6	36.7	37.4	27.1	29.8
VTransE [52]	65.7	67.6	38.6	39.4	29.7	34.3
VCTree [53]	65.5	67.4	40.3	41.6	31.9	36
GPS-Net [54]	65.2	67.1	37.8	39.2	31.3	35.9
BGNN [17]	66.1	67.7	40	41.2	30.9	35.4
VETO [28]	64.2	66.3	35.7	36.9	27.5	31.5
SMP [24]	66.3	68	39.9	40.7	32.6	36.9
PE-Net [29]	64.9	67.2	39.4	40.7	30.7	35.2
FLIRL-Net (Ours)	**67.1**	**69.2**	**41.3**	**42.4**	**32.7**	**37.1**

image contains approximately 11.5 annotated entities, with recorded position and attribute information for each entity. We used 70% of the VG150 images for training, 5,000 images for validation, and the remaining 26,446 images for testing. The dataset contains 622,705 relationship annotations, but these relationships do not distinguish between different levels of importance.

To calculate relationship importance scores using fuzzy logic, we preprocess the relevant data in VG150. We calculated the area of relationships based on the bounding box coordinates of the subject and object. We scaled the predicate frequencies to the range [0,1] to prevent high-frequency predicates from disproportionately affecting the results. Additionally, we enabled users to modify the specified entity classes, predicate classes, and attribute classes according to their requiements. Specified entity and predicate classes were marked using binary classification. The specified attribute class was determined by calculating the total number of specified attribute categories for the subject and object.

4.2 Experimental Setup and Evaluation Metrics

We use the PyTorch framework on an NVIDIA GeForce RTX 4090 GPU for training and testing. The experimental environment is torch1.7.1+cu110. For VG150, we use the same pre-trained object detector, Faster R-CNN (backbone: ResNeXt-101-FPN) [29, 35, 39]. The FLIRL-Net trains for 50K iterations using the SGD optimizer, with an initial learning rate of 0.001 and a batch size of 8.

In the fuzzification process using Gaussian functions as membership functions, the input parameters' fuzzy sets are evenly distributed within the range of 0 to 1. Therefore, the centers of the three fuzzy sets a are set to 0.25, 0.5, and 0.75, respectively. The parameter β is set to 1/6 to ensure smooth transitions between the fuzzy sets. In

the fuzzification process using the Sigmoid function as the membership function, the input parameters' fuzzy sets are divided into the two extreme states of yes and no. The Sigmoid function ensures a smooth transition between 0 and 1, with the midpoint $\mathcal{E}$ set to 0.5 and the steepness parameter h set to 20 and -20. In calculating the distance loss for relationship entity pairs, we set the extreme value trimming proportion δ to 0.1.

We evaluate our method FLIRL-Net on three subtasks: Predicate Classification (PredCls), Scene Graph Classification (SGCls), and Scene Graph Detection (SGDet). In the PredCls task, the input image includes the location and class labels of all entities. The model aims to predict the relationships between these entities, focusing on understanding interactions while ignoring entity recognition challenges. In the SGCls task, the input image contains the locations of all entities but no class labels. The model must first classify each entity and then predict the relationships between them. This task is more challenging than the PredCls task because the accuracy of entity classification directly impacts relationship prediction. The SGDet task is the most challenging task. The input image lacks any location or class information for entities. The model must identify the locations and classes of all entities and then predict the relationships between them. Since the model recognizes entities from scratch, there may be discrepancies compared to manually annotated data, increasing the difficulty of relationship prediction and potentially affecting overall accuracy.

4.3 Experimental Results

To validate the effectiveness of our proposed FLIRL-Net method in relationship prediction, we select several state-of-the-art SGG methods [15, 17, 24, 28–30, 33, 34, 51–54] and compare them with our method using the R@K metric. These models achieve high R@50 and R@100 results on the VG150 dataset. We conduct comparisons on the PredCls, SGCls, and SGDet subtasks. The results are shown in Tab. 1.

According to the results in Tab. 1, our method demonstrates superior performance in relationship prediction across all three subtasks. In the PredCls task, FLIRL-Net improves by 0.8% on metric R@50 and 1.2% on metric R@100 compared to the best-performing method. In the SGCls task, FLIRL-Net achieves an increase of 1% on R@50 and 0.8% on R@100 compared to the top method. In the SGDet task, FLIRL-Net improves by 0.1% on metric R@50 and 0.2% on metric R@100 compared to the best-performing method. These results indicate that FLIRL-Net achieves a high level on the fundamental evaluation metric Recall@K, demonstrating strong relationship prediction capabilities.

To evaluate FLIRL-Net's ability to predict important relationships, we use the IR@K and IP@K metrics. We select three baseline models for scene graph generation: Motifs [33], Transformer [15], and PE-Net [29], to validate the effectiveness of the complete FLIRL-Net method and its Important Relationship Learning (IRL) module.

We reproduce the baseline models and integrate our IRL module into them. Models named PE-Net+IRL, Motifs+IRL, and Transformer+IRL represent these enhanced versions. This approach verifies whether the IRL module improves the baseline models' ability to predict important relationships. We also compare our baseline model without the IRL module (Ours-) with the full FLIRL-Net. All models are trained and tested on

Table 2 Comparison results of FLIRL-Net and baseline models on IR@K and IP@K metrics.

Model	PredCls				SGCls				SGDet			
	IR@K		IP@K		IR@K		IP@K		IR@K		IP@K	
	IR@5	IR@10	IP@5	IP@10	IR@5	IR@10	IP@5	IP@10	IR@5	IR@10	IP@5	IP@10
PE-Net	32.47	44.42	86.43	85.28	20.04	27.77	82.70	83.13	11.34	16.48	87.35	86.32
PE-Net+IML	38.05	50.53	89.24	87.34	22.92	30.51	85.91	84.81	11.45	16.83	91.74	89.78
Motifs	38.15	50.16	89.22	87.25	23.92	30.81	88.40	86.07	12.07	17.36	90.55	87.88
Motifs+IML	39.48	51.95	90.26	88.15	24.52	31.61	**89.65**	**87.00**	12.48	17.78	90.72	88.47
Transformer	38.25	50.25	89.51	87.49	24.38	31.49	87.97	85.76	12.41	17.83	91.01	88.91
Transformer+IML	39.57	52.09	90.49	88.45	25.18	32.44	88.66	86.36	12.70	18.25	91.43	89.25
Ours-	38.29	50.29	89.65	87.62	24.40	31.68	88.82	86.47	13.69	19.08	90.28	88.35
FLRIW-Net(Ours)	**39.69**	**52.12**	**90.79**	**88.57**	**25.25**	**32.60**	89.51	86.96	**13.97**	**19.42**	**91.94**	**89.85**

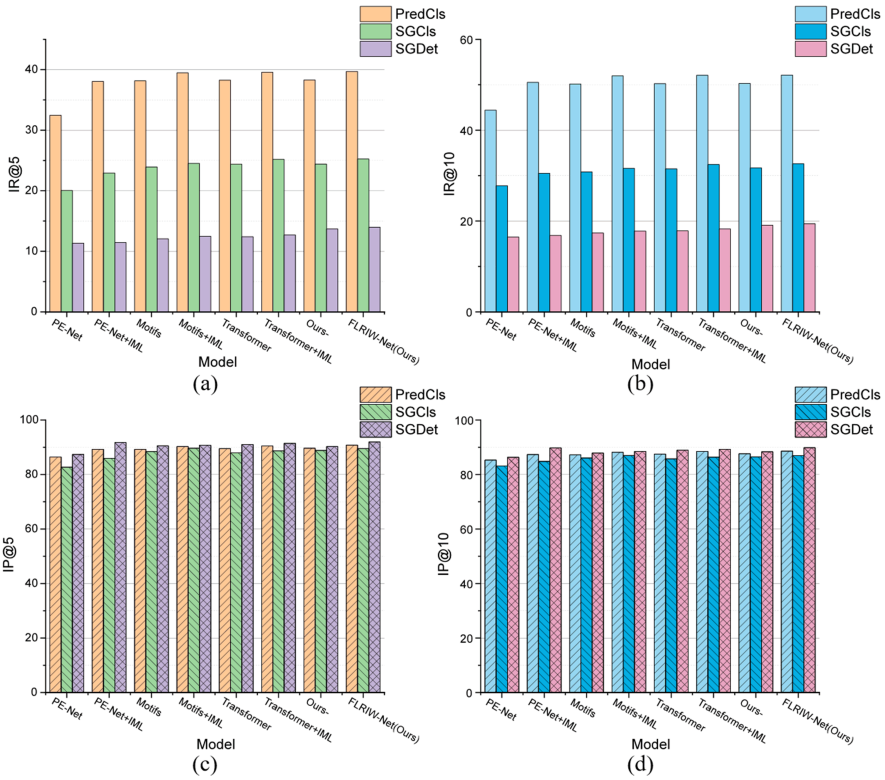


Fig. 7 Bar charts compare baseline models without the Important Relationship Learning (IRL) module and those with it across PredCls, SGCls, and SGDet tasks. (a) displays IR@5 results, (b) IR@10, (c) IP@5, and (d) IP@10

the VG150 dataset with the same batch size and iterations, recording IR@5, IR@10, IP@5, and IP@10 metrics. The results for PredCls, SGCls, and SGDet subtasks are shown in Tab. 2.

In PredCls, the full FLIRL-Net achieves the highest performance on both IR@K and IP@K metrics. Adding the IRL module to PE-Net improves IR@5, IR@10, IP@5,

Table 3 Ablation study of FLIRL-Net method on the PredCls task.

Model	PredCls					
	R@K		IR@K		IP@K	
	K=50	K=100	K=5	K=10	K=5	K=10
1: A	65.77	67.98	38.29	50.29	89.65	87.62
2: A+B	66.14	68.36	38.78	51.02	90.65	88.49
3: A+C	66.56	68.7	39.08	51.4	90.03	87.96
4: A+B+C	**67.06**	**69.17**	**39.69**	**52.12**	**90.79**	**88.57**

Table 4 Ablation study of FLIRL-Net method on the SGCls task.

Model	SGCls					
	R@K		IR@K		IP@K	
	K=50	K=100	K=5	K=10	K=5	K=10
1: A	40.54	41.65	24.4	31.68	88.82	86.47
2: A+B	40.66	41.85	24.46	31.82	**89.52**	**87.08**
3: A+C	41.3	42.38	25.01	32.34	88.41	85.96
4: A+B+C	**41.25**	**42.36**	**25.25**	**32.6**	89.51	86.96

Table 5 Ablation study of FLIRL-Net on the SGDet task.

Model	SGDet					
	R@K		IR@K		IP@K	
	K=50	K=100	K=5	K=10	K=5	K=10
1: A	31.69	36.07	13.69	19.08	90.28	88.35
2: A+B	31.78	36.13	13.6	19.07	91.08	89.03
3: A+C	32.08	36.43	**14.09**	**19.51**	90.8	88.91
4: A+B+C	**32.69**	**37.12**	13.97	19.42	**91.94**	**89.85**

and IP@10 by 5.58%, 6.11%, 2.81%, and 2.06%, respectively. PE-Net, being an unbiased model, tends to predict more tail predicates, resulting in lower performance on top K relationships. The IRL module significantly boosts PE-Net's performance. Even for strong baseline models like Motifs and Transformer, incorporating the IRL module further enhances their ability to predict important relationships.

In SGCls, the full FLIRL-Net shows superior performance on the IR@K metric. Models with the IRL module exhibit significant improvements in both IR@K and IP@K metrics compared to the original models. Notably, Motifs had lower IP@K

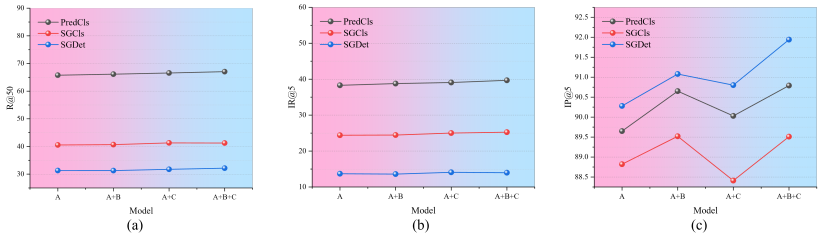


Fig. 8 Line charts comparing ablation study results across PredCls, SGCls, and SGDet tasks. (a) shows metric R@50 comparison results, (b) shows metric IR@5 comparison results, and (c) shows metric IP@5 comparison results

results compared to Ours-, but Motifs+IRL achieved the highest IP@5 and IP@10 results, surpassing FLIRL-Net by 0.14% and 0.04%. This suggests that integrating our IRL module with motif can lead to enhanced results. This improvement may stem from Motifs' enhanced ability to capture common relationships through detailed analysis of entity pair distributions and visual relationships. The integration with our IRL module creates a synergistic effect, preserving Motifs' strength in managing common relationships while enhancing its sensitivity to important relationships. This demonstrates that our model significantly improves the performance of existing models in predicting important relationships.

In SGDet, the full FLIRL-Net achieves the highest performance on both IR@K and IP@K metrics, improving IP@5 and IP@10 by 1.66% and 1.5%, respectively, compared to Ours-. Additionally, Ours- outperforms baseline models with the IRL module on IR@5 and IR@10, demonstrating that our Relationship Representation Modeling baseline effectively predicts important relationships. Incorporating the IRL module further enhances the baseline models' ability to predict important relationships.

To clearly illustrate the comparison of our method with other methods on metrics IR@K and IP@K, we also provide bar charts of the comparison results. The results are shown in Fig. 7.

4.4 Ablation Studies

Ablation studies are conducted to validate the effectiveness of each module in FLIRL-Net. The Relationship Representations Modeling (RRM) model is used as the baseline, with the RLIW and IEPFP modules sequentially added.

Scheme 1 is the baseline for our method, using the proposed RRM model (A) for relationship prediction. It refines subject and object features, along with bounding box features, by modeling the labels and bounding boxes of each entity. The fused relationship representation effectively expresses the interactions between entities. Scheme 2 builds upon Scheme 1 by adding the RLIW module (B). This module uses fuzzy logic-based relationship importance scores to weight the relationship label loss, ensuring the model focuses more on important relationships during training. Scheme 3 builds upon Scheme 1 by adding the IEPFP module (C). This module uses the Feature Pool-Former network to merge entity pair features with relationship importance scores, enhancing the representation of important relationship entity pairs and improving

Table 6 Relationship importance scores calculated by fuzzy logic for different subjects, objects and predicates.

Subject	Predicate	Object	R_{im}
child	wearing	jacket	0.73
woman	wearing	coat	0.78
cow	with	head	0.43
train	on	track	0.56
woman	sitting on	chair	0.80
windshield	on	train	0.45
book	on	chair	0.39
horse	has	tail	0.47
glass	on	shelf	0.32
woman	holding	laptop	0.75
horse	has	face	0.43

the model's ability to recognize them. Scheme 4 is our proposed FLIRL-Net method, which combines the RRM model with the IRL module.

We validate the evaluation results of R@K, IR@K, and IP@K metrics across the PredCls, SGCls, and SGDet subtasks. These experiments intuitively demonstrate the improvement each module brings to the overall network model. The ablation studies results for the PredCls task are shown in Tab. 3, for the SGCls task in Tab. 4, and for the SGDet task in Tab. 5.

According to the results in Tab. 3, FLIRL-Net method achieves the best test results on the R@K, IR@K, and IP@K metrics in the PredCls task. This indicates that the combined effect of modules RLIW and IEPFP enables the model to better predict important relationships.

According to the results in Tab. 4, FLIRL-Net method achieves the best test results on the R@K and IR@K metrics in the SGCls task. Scheme 2, which only includes module RLIW, achieves the best results on the IP@K metric. This indicates that module RLIW can better improve the model's ability to accurately judge the importance of relationships when correctly predicting them.

According to the results in Tab. 5, FLIRL-Net method achieves the best test results on the R@K and IP@K metrics in the SGDet task. Scheme 3, which only includes module IEPFP, achieves the best results on the IR@K metric. This indicates that module IEPFP can better enhance the model's ability to recognize important relationships.

Fig. 8 shows line charts comparing ablation study results across PredCls, SGCls, and SGDet tasks. We can see that method FLIRL-Net achieves the best R@K metric results in all three tasks. This indicates that the combined effect of the RLIW and IEPFP modules can further enhance the model's ability to predict relationships.

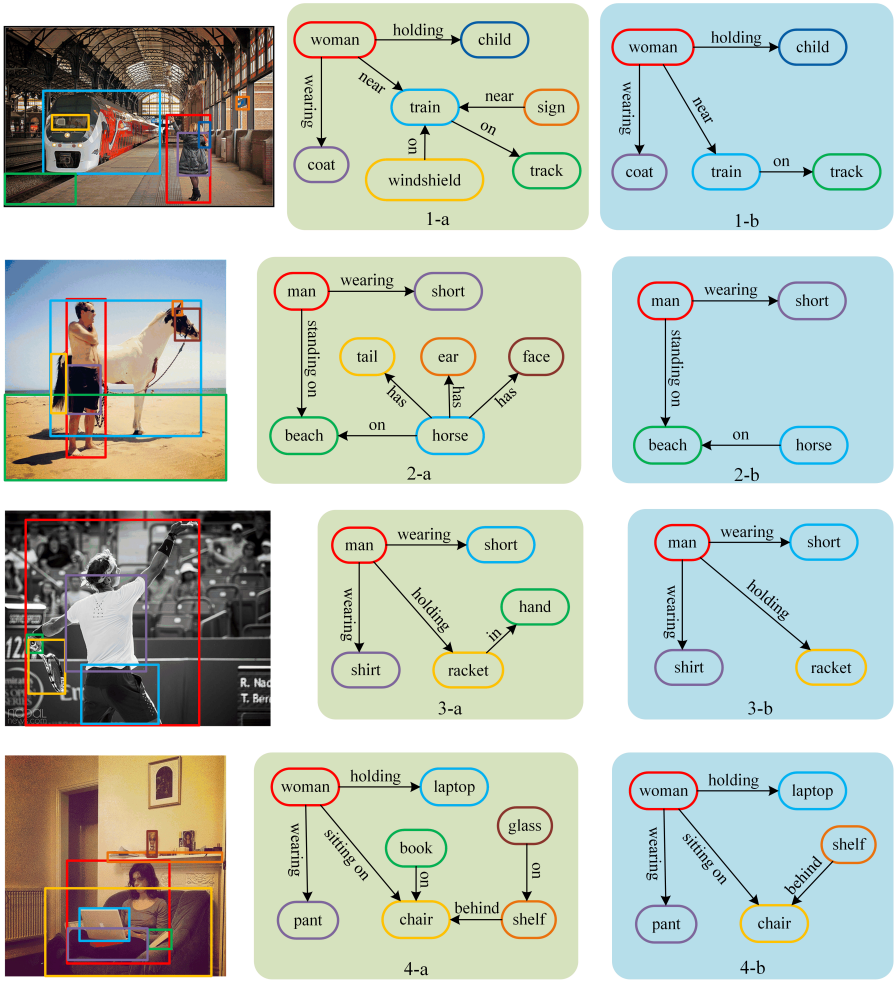


Fig. 9 Visualization results. The left shows the image and annotated bounding boxes. The right shows the scene graph generated by PE-Net (green) method and the scene graph generated by FLIRL-Net (blue) method

4.5 Visualization Results

In calculating the relationship importance scores, we specify entity classes such as 'man', 'woman', 'boy', 'girl', and 'child', which are related to human entities. We specify predicate classes such as 'wearing', 'holding', and 'riding', which are related to human body actions. Additionally, we specify attribute classes like 'orange' and 'tall'.

Tab. 6 presents several relationship importance scores calculated using fuzzy logic, along with their corresponding subject and object classes and predicate classes. It is important to note that varying entities and predicates based on users' requiements can lead to significant changes in relationship importance scores, resulting in different scene graphs. Additionally, the relationship importance scores are affected by the

entity bounding boxes and predicate frequencies. Therefore, the final results must be calculated using the fuzzy inference system according to actual conditions. Regardless of the relationship importance scores, our FLIRL-Net method can generate the scene graphs effectively.

To validate the effectiveness of FLIRL-Net method in predicting important relationships, we compare the scene graphs generated by PE-Net [29] (green) and FLIRL-Net (blue) methods in Fig. 9.

In the first example, our method's scene graph (1-b) filters out the ⟨sign-near-train⟩ and ⟨windshield-on-train⟩ relationships from PE-Net (1-a) due to lower importance scores. In the second example, FLIRL-Net (2-b) removes relationships involving the horse, focusing on human-related entities. In the third example, ⟨racket-in-hand⟩ is considered low importance and excluded from our scene graph (3-b). Similarly, in the fourth example, FLIRL-Net (4-b) excludes ⟨book-on-chair⟩ and ⟨glass-on-shelf⟩ as unimportant relationships. These results indicate that FLIRL-Net method outperforms PE-Net [29] method in predicting important relationships, enhancing the quality of scene graph generation and aiding downstream tasks in accurately parsing and utilizing the information within the scene graphs.

5 Conclusion

We propose the Fuzzy Logic-Driven Important Relationship Learning Network (FLIRL-Net), the first method in our survey to apply fuzzy logic to SGG. FLIRL-Net calculates relationship importance scores in the VG150 dataset tailored to user requirements, avoiding binary classification and meeting personalized needs in different domains. The IRL module enhances the model's ability to predict important relationships. Comparative experiments across three SGG tasks demonstrate superior performance, and ablation studies validate the effectiveness of each module. Experimental results confirm that our model improves SGG quality, introducing a novel perspective with fuzzy logic integration. Future research will extend the use of fuzzy logic and the proposed modules to datasets like Open Images and COCO, enabling the identification of important relationships in more complex images. This will broaden the applicability of scene graphs in a wider range of downstream tasks, offering stronger support for future visual understanding tasks.

Acknowledgements. This work was supported in part by the Postgraduate Research & Practice Innovation Program of Jiangsu Province, China under Grant SJCX25_2006.

Author contributions. Jin Wang conceptualized the research framework, designed and conducted the main experiments, and prepared the initial draft of the manuscript. Zilong Yang developed the methodology, implemented the algorithms, and performed data analysis. Jialing Xu designed the experimental protocols, collected and preprocessed the data, and conducted statistical analysis. Sixu Wu supervised the project, provided conceptual guidance and approved the final version of the paper to be published. All authors reviewed the manuscript.

Data Availability. No datasets were generated or analysed during the current study.

Declarations

Conflict of interest. The authors declare no Conflict of interest.

Funding. This work was supported in part by the Postgraduate Research & Practice Innovation Program of Jiangsu Province, China under Grant SJCX25_2006.

References

[1] Nguyen, K., Tripathi, S., Du, B., Guha, T., Nguyen, T.Q.: In defense of scene graphs for image captioning. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1387–1396 (2021). https://doi.org/10.1109/ICCV48922.2021.00144

[2] Chen, S., Jin, Q., Wang, P., Wu, Q.: Say as you wish: Fine-grained control of image caption generation with abstract scene graphs. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9959–9968 (2020). https://doi.org/10.1109/CVPR42600.2020.00998

[3] Sortino, R., Palazzo, S., Rundo, F., Spampinato, C.: Transformer-based image generation from scene graphs. Computer Vision and Image Understanding **233**, 103721 (2023) https://doi.org/10.1016/j.cviu.2023.103721

[4] Schroeder, B., Tripathi, S.: Structured query-based image retrieval using scene graphs. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 680–684 (2020). https://doi.org/10.1109/CVPRW50498.2020.00097

[5] Johnson, J., Krishna, R., Stark, M., Li, L.-J., Shamma, D.A., Bernstein, M.S., Fei-Fei, L.: Image retrieval using scene graphs. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3668–3678 (2015). https://doi.org/10.1109/CVPR.2015.7298990

[6] Lu, X., Gao, Y.: Guide and interact: scene-graph based generation and control of video captions. Multimedia Systems **29**, 797–809 (2023) https://doi.org/10.1007/s00530-022-01012-7

[7] Xiong, H., Ding, J., Zhou, Y., *et al.*: Stagvid2c: enhancing video-based commonsense captioning with spatio-temporal action graph. Multimedia Systems **31**, 332 (2025) https://doi.org/10.1007/s00530-025-01909-z

[8] Chen, M., Lyu, X., Guo, Y., Liu, J., Gao, L., Song, J.: Multi-scale graph attention network for scene graph generation. In: 2022 IEEE International Conference

on Multimedia and Expo (ICME), pp. 1–6 (2022). https://doi.org/10.1109/ICME52920.2022.9859970

[9] Yu, J., Chai, Y., Wang, Y., Hu, Y., Wu, Q.: Cogtree: Cognition tree loss for unbiased scene graph generation. IJCAI International Joint Conference on Artificial Intelligence, 1274–1280 (2021)

[10] Zhang, R., Xu, P., Kang, K., Yang, Y.: Geometric and semantic improvement for unbiased scene graph generation. Available at SSRN 4237467 (2023)

[11] Li, L., Xiao, J., Shi, H., Zhang, H., Yang, Y., Liu, W., Chen, L.: Nicest: Noisy label correction and training for robust scene graph generation. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(10), 6873–6888 (2024) https://doi.org/10.1109/TPAMI.2024.3387349

[12] Li, W., Zhang, H., Bai, Q., Zhao, G., Jiang, N., Yuan, X.: Ppdl: Predicate probability distribution based loss for unbiased scene graph generation. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 19425–19434 (2022). https://doi.org/10.1109/CVPR52688.2022.01884

[13] Duan, J., Min, W., Lin, D., Xu, J., Xiong, X.: Multimodal graph inference network for scene graph generation. Applied Intelligence **51**, 8768–8783 (2021)

[14] Phueaksri, I., Kastner, M.A., Kawanishi, Y., Komamizu, T., Ide, I.: Image-collection summarization using scene-graph generation with external knowledge. IEEE Access **12**, 17499–17512 (2024) https://doi.org/10.1109/ACCESS.2024.3360113

[15] Wang, Y., Gao, Y., Yu, W., Guo, R., Wan, W., Yang, S., Huang, B.: Transformer networks with adaptive inference for scene graph generation. Applied Intelligence **53**(8), 9621–9633 (2023)

[16] Chen, Z., Rezayi, S., Li, S.: More knowledge, less bias: Unbiasing scene graph generation with explicit ontological adjustment. In: 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 4012–4021 (2023). https://doi.org/10.1109/WACV56688.2023.00401

[17] Li, R., Zhang, S., Wan, B., He, X.: Bipartite graph network with adaptive message passing for unbiased scene graph generation. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11104–11114 (2021). https://doi.org/10.1109/CVPR46437.2021.01096

[18] Lyu, X., Gao, L., Zeng, P., Shen, H.T., Song, J.: Adaptive fine-grained predicates learning for scene graph generation. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(11), 13921–13940 (2023) https://doi.org/10.1109/TPAMI.2023.3298356

[19] Min, Y., Wu, A., Deng, C.: Environment-invariant curriculum relation learning for fine-grained scene graph generation. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 13250–13261 (2023). https://doi.org/10.1109/ICCV51070.2023.01223

[20] Li, L., Chen, G., Xiao, J., Yang, Y., Wang, C., Chen, L.: Compositional feature augmentation for unbiased scene graph generation. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 21628–21638 (2023). https://doi.org/10.1109/ICCV51070.2023.01982

[21] Kang, H., Yoo, C.D.: Skew class-balanced re-weighting for unbiased scene graph generation. Machine Learning and Knowledge Extraction **5**(1), 287–303 (2023)

[22] Hu, Y., Zhang, F., Wei, R., *et al.*: Learning semantic-unified cross-modal representations for open-vocabulary video scene graph generation. Multimedia Systems **31**, 188 (2025) https://doi.org/10.1007/s00530-025-01767-9

[23] Chang, X., Wang, T., Cai, S., Sun, C.: Landmark: Language-guided representation enhancement framework for scene graph generation. Applied Intelligence **53**(21), 26126–26138 (2023)

[24] Zhang, Y., Pan, Y., Yao, T., Huang, R., Mei, T., Chen, C.-W.: Boosting scene graph generation with visual relation saliency. ACM Transactions on Multimedia Computing, Communications and Applications **19**(1), 1–17 (2023)

[25] Tian, P., Mo, H., Jiang, L.: Scene graph generation by multi-level semantic tasks. Applied Intelligence **51**(11), 7781–7793 (2021)

[26] Kim, K., Yoon, K., Jeon, J., In, Y., Moon, J., Kim, D., Park, C.: Llm4sgg: Large language models for weakly supervised scene graph generation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 28306–28316 (2024)

[27] Ren, C., Liu, X., Cao, M., Zhang, J., Wang, H.: A novel end-to-end transformer for scene graph generation. In: 2023 International Joint Conference on Neural Networks (IJCNN), pp. 1–7 (2023). https://doi.org/10.1109/IJCNN54540.2023.10191798

[28] Sudhakaran, G., Dhami, D.S., Kersting, K., Roth, S.: Vision relation transformer for unbiased scene graph generation. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 21825–21836 (2023). https://doi.org/10.1109/ICCV51070.2023.02000

[29] Zheng, C., Lyu, X., Gao, L., Dai, B., Song, J.: Prototype-based embedding network for scene graph generation. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 22783–22792 (2023). https://doi.org/10.1109/CVPR52729.2023.02182

[30] Xu, D., Zhu, Y., Choy, C.B., Fei-Fei, L.: Scene graph generation by iterative message passing. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3097–3106 (2017). https://doi.org/10.1109/CVPR.2017.330

[31] Wang, W., Wang, R., Shan, S., Chen, X.: Sketching image gist: Human-mimetic hierarchical scene graph generation. In: European Conference on Computer Vision, pp. 222–239 (2020). Springer

[32] Lu, C., Krishna, R., Bernstein, M., Fei-Fei, L.: Visual relationship detection with language priors. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, pp. 852–869 (2016). Springer

[33] Zellers, R., Yatskar, M., Thomson, S., Choi, Y.: Neural motifs: Scene graph parsing with global context. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5831–5840 (2018). https://doi.org/10.1109/CVPR.2018.00611

[34] Teng, Y., Wang, L.: Structured sparse r-cnn for direct scene graph generation. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 19415–19424 (2022). https://doi.org/10.1109/CVPR52688.2022.01883

[35] Tang, K., Niu, Y., Huang, J., Shi, J., Zhang, H.: Unbiased scene graph generation from biased training. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3713–3722 (2020). https://doi.org/10.1109/CVPR42600.2020.00377

[36] Li, H., Lv, J., Xiao, Q., Liu, Y., Wu, S., Huang, X.: Avr: Attention based salient visual relationship detection. In: Fourteenth International Conference on Digital Image Processing (ICDIP 2022), vol. 12342, pp. 697–708 (2022). SPIE

[37] Lu, J., Chen, L., Guan, H., Lin, S., Gu, C., Wang, C., He, G.: Improving rare relation inferring for scene graph generation using bipartite graph network. Computer Vision and Image Understanding **239**, 103901 (2024)

[38] Benetatos, A., Diomataris, M., Pitsikalis, V., Maragos, P.: Generating salient scene graphs with weak language supervision. In: 2023 31st European Signal Processing Conference (EUSIPCO), pp. 526–530 (2023). https://doi.org/10.23919/EUSIPCO58844.2023.10289877

[39] Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(6), 1137–1149 (2017) https://doi.org/10.1109/TPAMI.2016.2577031

[40] Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)

[41] Popescu, M.-C., Balas, V.E., Perescu-Popescu, L., Mastorakis, N.: Multilayer perceptron and neural networks. WSEAS Transactions on Circuits and Systems **8**(7), 579–588 (2009)

[42] Thang, N.T., Giang, L.T., Son, L.H., Giang, N.L., Taniar, D., Thien, N.V., Tuan, T.M.: A novel spatial complex fuzzy inference system for detection of changes in remote sensing images. Applied Intelligence **55**(2), 178 (2025)

[43] Ding, W., Wang, H., Huang, J., Ju, H., Geng, Y., Lin, C.-T., Pedrycz, W.: Ftranscnn: Fusing transformer and a cnn based on fuzzy logic for uncertain medical image segmentation. Information Fusion **99**, 101880 (2023)

[44] Ma, X., Chen, L., Deng, Z., Xu, P., Yan, Q., Choi, K.-S., Wang, S.: Deep image feature learning with fuzzy rules. IEEE Transactions on Emerging Topics in Computational Intelligence **8**(1), 724–737 (2024) https://doi.org/10.1109/TETCI.2023.3259447

[45] Bhuyan, S., Kar, A., Sen, D., Deb, S.: Rgb-d fusion through zero-shot fuzzy membership learning for salient object detection. IEEE Transactions on Artificial Intelligence **5**(7), 3638–3652 (2024) https://doi.org/10.1109/TAI.2024.3376640

[46] Singh, S.K., Abolghasemi, V., Anisi, M.H.: Fuzzy logic with deep learning for detection of skin cancer. Applied Sciences **13**(15), 8927 (2023)

[47] Nachar, R., Inaty, E., Bonnin, P.J.: A high performance neural network and fuzzy logic based edge corner detector (nf-ecd). Multimedia Tools and Applications **82**(25), 39459–39480 (2023)

[48] Jawdekar, A., Dixit, M.: Deep learning and fuzzy logic based intelligent technique for the image enhancement and edge detection framework. Traitement Du Signal **40**(1) (2023)

[49] Papanastasiou, G., Dikaios, N., Huang, J., Wang, C., Yang, G.: Is attention all you need in medical image analysis? a review. IEEE Journal of Biomedical and Health Informatics **28**(3), 1398–1411 (2024) https://doi.org/10.1109/JBHI.2023.3348436

[50] Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., Yan, S.: Metaformer is actually what you need for vision. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10809–10819 (2022). https://doi.org/10.1109/CVPR52688.2022.01055

[51] Chen, T., Yu, W., Chen, R., Lin, L.: Knowledge-embedded routing network for

scene graph generation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6156–6164 (2019). https://doi.org/10.1109/CVPR.2019.00632

[52] Zhang, H., Kyaw, Z., Chang, S.-F., Chua, T.-S.: Visual translation embedding network for visual relation detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3107–3115 (2017). https://doi.org/10.1109/CVPR.2017.331

[53] Tang, K., Zhang, H., Wu, B., Luo, W., Liu, W.: Learning to compose dynamic tree structures for visual contexts. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6612–6621 (2019). https://doi.org/10.1109/CVPR.2019.00678

[54] Lin, X., Ding, C., Zeng, J., Tao, D.: Gps-net: Graph property sensing network for scene graph generation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3743–3752 (2020). https://doi.org/10.1109/CVPR42600.2020.00380