# Robust Hierarchical Scene Graph Generation

**Ce Zhang** **Simon Stepputtis** **Joseph Campbell** **Katia Sycara** **Yaqi Xie**
Carnegie Mellon University
{cezhang, sstepput, jacampbe, katia, yaqix}@cs.cmu.edu

## Abstract

The ability to quickly understand scenes from visual observations via structured representations, known as Scene Graph Generation (SGG), is a crucial component of perception models. Despite recent advancements, most existing models assume perfect observations, an often-unrealistic condition in real-world scenarios. Such models can struggle with visual inputs affected by natural corruptions such as sunlight glare, extreme weather conditions, and smoke. Drawing inspiration from human hierarchical reasoning skills (i.e., from higher to lower levels) as a defense against corruption, we propose a new framework called **Hi**erarchical **K**nowledge **E**nhanced **R**obust **S**cene **G**raph **G**eneration (HiKER-SGG). First, we create a hierarchical knowledge graph, facilitating machine comprehension of this structured knowledge. Then we bridge between the constructed graph and the initial scene graph and perform message passing for hierarchical graph reasoning. Finally, we propose a hierarchical inference process to enable the model to predict from a higher to lower level, thus enhancing robustness against corruptions that frequently impact only fine-grained details. Experiments on various settings confirm the superior performance of the proposed framework with both clean and corrupted images.

## 1 Introduction

Scene Graph Generation (SGG) [1, 2, 3] is a key step in understanding visual scenes, focusing on finding object instances and their visual relations. Typically, a scene graph is a visual representation where each node stands for an object and each edge represents the relation between them [1]. SGG aims at the generation of scene graphs, which has attracted a lot of interest for its practical significance in understanding the visual world [4, 5, 6, 7]. However, most existing studies assume the images used are perfect. This contrasts with real-world situations where images often have natural corruptions like sun glare, smoke, and water drops [8, 9]. To address this, our study focuses on improving SGG in situations with natural corruptions. More specifically, given the impracticality of enumerating all possible corruptions in the real world, our goal is to develop a robust SGG model that is agnostic to the corruption types. In other words, these corruptions are not specified during the training phase, and the model is designed to yield robust results across a diverse range of corruption types.

Arguably, human perception [10, 11] has its specific strategies to stay robust against corruptions, such as reasoning from the higher to the lower levels. Consider an image of a cat as an example. Humans might first identify it as an animal, and subsequently as a cat. Though this hierarchical reasoning typically occurs instantaneously, it significantly enhances the robustness of our perception. For instance, if the cat's head is obscured by sun glare, accurate identification may be compromised, but it's highly probable that we can still successfully classify it as an animal based on other discernible features. This form of hierarchical reasoning aids in furnishing the maximum amount of information, even in situations where accurate results are unattainable due to observational interference.

However, it is challenging for SGG to benefit from hierarchical reasoning. It is worth noting that, the hierarchical knowledge, such as the relationship between superclasses and subclasses (*e.g.*, animals and cats) is not given in the scene graphs. Humans may acquire this understanding through daily

NeurIPS 2023 New Frontiers in Graph Learning Workshop (NeurIPS GLFrontiers 2023).

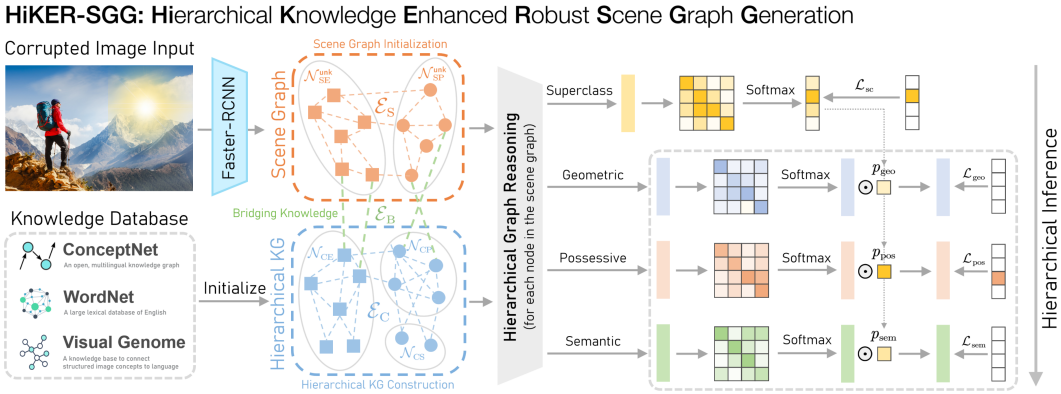


Figure 1: HiKER-SGG Overview. Hierarchical knowledge graphs are constructed from an external knowledge base. Given an image, we first initialize the scene graph using an off-the-shelf detector, Faster-RCNN. We then create bridging connections between the hierarchical knowledge graph and the initial scene graph and perform message passing for hierarchical graph reasoning. Finally, we design a hierarchical inference process to guide the model in making step-by-step predictions explicitly.

interactions with the world over many years, but it is almost intractable for machines to do so [12, 13]. Furthermore, even with access to hierarchical knowledge, determining how to effectively utilize this additional information is far from straightforward.

To this end, we propose a novel framework, **Hi**erarchical **K**nowledge **E**nhanced **R**obust **S**cene **G**raph **G**eneration (HiKER-SGG). HiKER-SGG utilizes hierarchical knowledge sourced from external knowledge bases to refine the initial scene graph produced by the off-the-shelf detector. More specifically, we enforce hierarchical information by adding relations between superclasses and subclasses. Then we connect each entity in the scene graph to the corresponding entities in the knowledge graph, matching the label predicted by the off-the-shelf detector. Subsequently, message passing is performed on the bridged graph to facilitate information flow and reasoning. Furthermore, we integrate a hierarchical algorithm prior within the prediction head of HiKER-SGG. This enables HiKER-SGG to initially make predictions for the superclass and then delve into the details to predict the subclass, conditioned on the superclass.

We conducted comprehensive experiments on the Visual Genome dataset and established that HiKER-SGG outperforms state-of-the-art models in handling both clean and corrupted images. It's important to note that all models were trained on clean images and directly tested on corrupted ones without further training. Additionally, we conducted an ablation study to discern the distinct impacts of HiKER-SGG components. Our findings reveal that both the hierarchical knowledge and the hierarchical inference process are crucial for the enhanced performance of HiKER-SGG, with the removal of hierarchical knowledge leading to a more pronounced degradation in performance. We conjecture that the hierarchical structure is essential for hierarchical inference to fully leverage its benefits. In summary, our contributions to the field are threefold:

1. We pioneer the exploration of Scene Graph Generation (SGG) in the presence of natural corruptions, a perspective not previously undertaken in existing works.
2. We introduce a novel framework, HiKER-SGG, that strategically leverages hierarchical knowledge and hierarchical inference to fortify SGG against the challenges posed by natural corruptions, ensuring more robust and reliable representations.
3. Through comprehensive empirical evaluation on both clean and corrupted images, we demonstrate HiKER-SGG's prevalent performance in robust scene graph generation.

## 2 Related Work

**Scene Graph Generation.** Scene graph generation has emerged as a key area of focus in computer vision research, with the goal of offering a structured depiction of an image through the identification of objects and their intricate relations [1, 2]. Furthermore, numerous studies illustrate that scene graphs can serve as a valuable source of auxiliary information, thereby enhancing image understanding

for applications such as image retrieval [14, 15, 16], image captioning [17, 18, 19, 20], and visual question answering [21, 22]. In recent years, a multitude of research efforts have been dedicated to enhancing the performance of SGG on the well-known Visual Genome [23] dataset. The seminal work in this domain was conducted by Xu *et al.* [4], who employs iterative message passing to generate visually grounded scene graphs. Subsequent to this pioneering work, several researchers have adopted the message passing mechanism to better comprehend visual context [5, 24, 25, 26, 27].

While traditional SGG techniques have shown promising results, they often suffer from the long-tailed distribution of relation predicates [28, 29, 30, 31]. Predicates in visual relations are often unevenly distributed, with head predicates (*e.g.*, `on`, `have`) dominating the relation expressions [32, 33, 34, 35]. Such general relation expressions, however, offer limited utility for in-depth visual relation analysis [36, 37, 38]. To address this challenge, He *et al.* [39] introduces a knowledge transfer mechanism to leverage insights from head relations to enhance the representation of tail relations. TDE [32] employs causal inference to discern and rectify harmful biases by extracting counterfactual causality from training graphs. EBM [7] introduces an energy-based learning framework that incorporates structural information into the loss function. Guo *et al.* [37] refines biased predicate predictions based on the confusion matrix generated by training data. Our work differs from conventional SGG in that we don't assume that observations are perfect. We allow for natural corruptions in images, which are typical in real-world situations.

**Knowledge Based SGG.** Recently, several approaches have been proposed to integrate external knowledge, referred to as *commonsense*, to refine predicate and object prediction and enhance the generalizability of the SGG model [24, 40, 41, 42]. Specifically, GB-Net [41] suggests that a scene graph can be perceived as an instantiation of a commonsense knowledge graph conditioned by the content of the image, and employs GGNN [43] to iteratively propagate messages between these two graphs for SGG task. Furthermore, EOA [42] advances this by enriching the knowledge graph for SGG with off-scene entities, thereby offering a more comprehensive and context-aware scene graph representation. In this work, we extend this by introducing superclass nodes and incorporating hierarchical edges into the knowledge graph, thereby facilitating hierarchical predicate prediction for SGG models. This is particularly advantageous when observations are corrupted, where features for specific classes are not easily detectable. In such cases, the hierarchical knowledge guides the model to first detect the superclass features. By adopting this approach, we can streamline the search space and facilitate more accurate predictions for finer classes.

**Corrupted Observation Perception.** In many computer vision tasks, it is a common assumption among researchers that the input image is invariably flawless and clear. However, this is often not the case in practical scenarios. To address this important issue, several benchmarks have been introduced to assess the robustness of the neural network models to real-world corruptions [8, 44, 45]. Within the context of corruption robustness, recent advancements can be broadly categorized into transfer learning [46, 47, 48], adversarial training [49, 50, 51], and data augmentation [52, 53]. Recently, LogicDef [54] proposes a logic rules based defense method for adversarial patch attacks on images with multiple objects, utilizing logic rules learned from object relations to identify the attacked object. However, their approach assumes that the attack patch is on one single object, known to be under attack, and thus it is labeled as "unknown." Additionally, they assume that the relations between objects remain unaffected by the attack. In contrast, our work allows for corruption to occur at any location, potentially impacting an unknown number of objects and relations, which is more challenging as well as more realistic. To the best of our knowledge, ours is the first work to introduce natural corruptions into scene graph generation and to propose the integration of hierarchical knowledge to ensure robust SGG in the presence of such corruptions.

## 3 Hierarchical Knowledge Enhanced Robust Scene Graph Generation

We introduce a novel framework **Hi**erarchical **K**nowledge **E**nhanced **R**obust **S**cene **G**raph **G**eneration (HiKER-SGG), as illustrated in Figure 1, to enhance scene understanding for observations with potential natural corruptions. Hierarchical knowledge graphs are constructed from an external knowledge base. Unlike conventional knowledge graphs, our hierarchical knowledge graph explicitly incorporates superclass and subclass relations. Given an image, we first initialize a scene graph using an off-the-shelf detector, Faster-RCNN [55]. Next, we establish bridging connections by linking each scene graph entity to the knowledge graph entities according to the labels predicted by Faster R-CNN [55]. We employ a Graph Neural Network (GNN)-based [43] model to facilitate

information flow within the interconnected graph and utilize the updated node representations to make predictions. Subsequently, we design a hierarchical inference process to guide the model explicitly, first predicting the superclass, followed by further categorization into specific subclasses. The final prediction probability is calculated as the product of the superclass probability and the subclass probability conditioned on the superclass. This hierarchical approach proves particularly beneficial when observations are corrupted, making direct detection of subclasses challenging.

### 3.1 Problem Definition

Given an image $\mathcal{I}$ in a dataset $\boldsymbol{\mathcal{I}}$, the SGG model aims to generate a directed scene graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$, where each node $\mathcal{N}_i \in \mathcal{N}$ in the scene graph represents a localized object with bounding box $b_i$ and object class $\mathcal{C}_i^{\mathrm{E}}$, and each edge $\mathcal{E}_i \in \mathcal{E}$ denotes a predicate class $\mathcal{C}_i^{\mathrm{P}}$ between two objects. A well-constructed scene graph $\mathcal{G}$ contains a collection of visual relation triplets ($\langle$`subject-predicate-object`$\rangle$), which can be utilized to comprehensively describe the image $\mathcal{I}$.

Our proposed HiKER-SGG follows a two-stage paradigm. We first generate a set of entity proposals with corresponding features using an off-the-shelf object detector (*e.g.* Faster-RCNN [55]) with a feature extraction network (*e.g.* VGG [56] or ResNet [57]). The features extracted from the union box between two entities are used to represent their associated predicates. Leveraging these features, we jointly make predictions for both the entity and predicate classes.

### 3.2 Hierarchical Knowledge Construction

**Commonsense Knowledge Graph.** Similar to GB-Net [41], we leverage a commonsense knowledge graph which contains the possible relations between objects derived from extensive datasets like ConceptNet [58], WordNet [59], *etc*. Its edges serve as repositories of information regarding the affordances and general knowledge associated with objects, exemplified by connections such as `man-wears-shirt` and `cat-is-animal`. For simplicity, we define our commonsense graph as comprising a set of commonsense entity (CE) nodes $\mathcal{N}_{\mathrm{CE}}$ and commonsense predicate (CP) nodes $\mathcal{N}_{\mathrm{CP}}$ that are present in our SGG task. Note that the commonsense graph also contains a special entity node representing "background/no entity" and a predicate node representing "background/no predicate." The edges in the commonsense graph $\mathcal{E}_{\mathrm{C}}$ store the relations between each pair of nodes in both sets, which can be denoted as

$$\mathcal{E}_{\mathrm{C}} = \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CE}\to\mathrm{CP}}\} \cup \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CP}\to\mathrm{CE}}\} \cup \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CE}\to\mathrm{CE}}\} \cup \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CP}\to\mathrm{CP}}\}. \tag{1}$$

We initialize the CE and CP nodes features with a linear projection of their word embeddings [60] $\mathbf{e}_i^{\mathrm{E}}$ and $\mathbf{e}_i^{\mathrm{P}}$:

$$\mathbf{x}_i^{\mathrm{CE}} = \texttt{LinearProj}(\mathbf{e}_i^{\mathrm{E}}), \quad \mathbf{x}_i^{\mathrm{CP}} = \texttt{LinearProj}(\mathbf{e}_i^{\mathrm{P}}). \tag{2}$$

**Hierarchical Commonsense Knowledge Graph.** To enable hierarchical predicate prediction, we introduce three specialized predicate nodes within the commonsense knowledge graph, referred to as commonsense superclass (CS) nodes. These nodes are denoted as $\mathcal{N}_{\mathrm{CS}}$ and correspond to three overarching predicate categories, namely `geometric`, `possessive`, and `semantic`. Below are some example subclasses associated with each of these superclasses:

- `geometric`: `above`, `behind`, `on`, `over`, `in`, `near`, `under` ... (15 subclasses in total)
- `possessive`: `belonging to`, `has`, `of`, `part of`, `to` ... (11 subclasses in total)
- `semantic`: `carrying`, `covering`, `eating`, `growing on`, `riding` ... (24 subclasses in total)

The initial representations of these superclass nodes are established by averaging the representations of $N_k$ subclass nodes associated with each superclass, as follows:

$$\mathbf{x}_k^{\mathrm{CS}} = \frac{\sum_i \mathbf{x}_i^{\mathrm{CP}}}{N_k} = \frac{\sum_i \texttt{LinearProj}(\mathbf{e}_i^{\mathrm{P}})}{N_k}. \tag{3}$$

Following the incorporation of these superclass predicate nodes, we establish dense binary connections $\mathcal{E}_{\texttt{hierarchical}}^{\mathrm{CS}\to\mathrm{CP}}$ and $\mathcal{E}_{\texttt{hierarchical}}^{\mathrm{CP}\to\mathrm{CS}}$ between $\mathcal{N}_{\mathrm{CS}}$ and $\mathcal{N}_{\mathrm{CP}}$ to encode hierarchical information. These edges also facilitate message passing, enabling the updating of superclass node representations, which are subsequently employed in computing superclass similarities. The final edges in the commonsense graph $\mathcal{E}_{\mathrm{C}}$ can be represented by

$$\mathcal{E}_{\mathrm{C}} = \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CE}\to\mathrm{CP}}\} \cup \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CP}\to\mathrm{CE}}\} \cup \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CE}\to\mathrm{CE}}\} \cup \{\mathcal{E}_{\texttt{relation}}^{\mathrm{CP}\to\mathrm{CP}}\} \cup \{\mathcal{E}_{\texttt{hierarchical}}^{\mathrm{CS}\to\mathrm{CP}}\} \cup \{\mathcal{E}_{\texttt{hierarchical}}^{\mathrm{CP}\to\mathrm{CS}}\}. \tag{4}$$

### 3.3 Scene Graph Initialization

A scene graph is different from a commonsense graph in that: (1) each scene entity (SE) node $\mathcal{N}_{\mathrm{SE}}$ is associated with a bounding box, *i.e.* $\mathcal{N}_{\mathrm{SE}} \subseteq [0,1]^4 \times \mathcal{N}_{\mathrm{CE}}$; (2) each scene predicate (SP) node $\mathcal{N}_{\mathrm{SP}}$ is associated with a pair of SE nodes, *i.e.* $\mathcal{N}_{\mathrm{SP}} \subseteq \mathcal{N}_{\mathrm{SE}} \times \mathcal{N}_{\mathrm{SE}} \times \mathcal{N}_{\mathrm{CP}}$. The directed edges $\mathcal{E}_{\mathrm{S}}$ in the scene graph can be similarly defined as

$$\mathcal{E}_{\mathrm{S}} = \{\mathcal{E}^{\mathrm{SE}\to\mathrm{SP}}_{\texttt{subjectOf}}\} \cup \{\mathcal{E}^{\mathrm{SE}\to\mathrm{SP}}_{\texttt{objectOf}}\} \cup \{\mathcal{E}^{\mathrm{SP}\to\mathrm{SE}}_{\texttt{hasSubject}}\} \cup \{\mathcal{E}^{\mathrm{SP}\to\mathrm{SE}}_{\texttt{hasObject}}\}. \tag{5}$$

In our SGG settings, the true classes for the SE/SP nodes might not be provided, and as such, we are tasked with predicting them. Therefore, we modify the scene graph entity nodes needed to be classified as $\mathcal{N}^{\texttt{unk}}_{\mathrm{SE}} \subseteq [0,1]^4$, and scene graph predicate nodes needed to be classified as $\mathcal{N}^{\texttt{unk}}_{\mathrm{SP}} \subseteq \mathcal{N}_{\mathrm{SE}} \times \mathcal{N}_{\mathrm{SE}}$, where $\mathcal{N}^{\texttt{unk}}_{\mathrm{SE/SP}}$ means the classes of the SE/SP nodes are unknown.

To initialize the scene graph for each sample, we first utilize the object detector to find potential objects. We then create a SE node for each object and a SP node for each pair of objects. The SE node is initialized by RoI-aligned [55] feature vector $\mathbf{v}^{\mathrm{E}}_i$, and the SP node is initialized by RoI feature $\mathbf{v}^{\mathrm{P}}_i$ of the union bounding box:

$$\mathbf{x}^{\mathrm{SE}}_i = \texttt{FCNet}(\mathbf{v}^{\mathrm{E}}_i)\ , \quad \mathbf{x}^{\mathrm{SP}}_i = \texttt{FCNet}(\mathbf{v}^{\mathrm{P}}_i), \tag{6}$$

where `FCNet` denotes a fully connected network. It should be noted that the weights for these two fully-connected networks are distinct and not shared. In the following equations, we denote $\mathbf{x}^{\mathrm{SE}}_i$ and $\mathbf{x}^{\mathrm{SP}}_i$ as the node representations of the the unknown nodes $\mathcal{N}^{\texttt{unk}}_{\mathrm{SE}}$ and $\mathcal{N}^{\texttt{unk}}_{\mathrm{SP}}$, respectively.

### 3.4 Bridging Hierarchical Knowledge and SGG

To bridge the knowledge graph and scene graph, we create *bridge edges* $\mathcal{E}_{\mathrm{B}}$ to facilitate the mutual information flow during training. Specifically, these bi-directional bridge edges link an entity or predicate from the scene graph to its corresponding labels in the commonsense graph. Given the symmetric nature of the relation, the bridge edges are implemented as bi-directional directed edges with shared weights. The bridge edges $\mathcal{E}_{\mathrm{B}}$ can be defined as

$$\mathcal{E}_{\mathrm{B}} = \{\mathcal{E}^{\mathrm{SE}\to\mathrm{CE}}_{\texttt{classifiedTo}}\} \cup \{\mathcal{E}^{\mathrm{SP}\to\mathrm{CP}}_{\texttt{classifiedTo}}\} \cup \{\mathcal{E}^{\mathrm{CE}\to\mathrm{SE}}_{\texttt{hasInstance}}\} \cup \{\mathcal{E}^{\mathrm{CP}\to\mathrm{SP}}_{\texttt{hasInstance}}\}. \tag{7}$$

Initially, we link each SE node to multiple CE nodes and assign weights based on the semantic labels predicted by Faster R-CNN [55]. The edges between SP and CP nodes start as an empty set and will be updated during the message propagation iterations.

Enforcing the information flow between knowledge graph and scene graph, we adopt a variant of GGNN [43], used in GB-Net [41], to update node representations and propagate messages among nodes using a Gated Recurrent Unit (GRU) [61] updating rule:

$$\mathbf{x}^{\phi}_i \leftarrow \texttt{GRU Update}(\mathbf{x}^{\phi}_i), \quad \phi \in \{\mathrm{SE}, \mathrm{SP}, \mathrm{CE}, \mathrm{CP}, \mathrm{CS}\}, \tag{8}$$

After each iteration of message propagation, we compute the similarities of each SE and SP node to all CE and CP nodes by

$$\mathrm{sim}\left(\mathbf{x}^{\mathrm{SE/SP}}_i, \mathbf{x}^{\mathrm{CE/CP}}_j\right) = \left(\texttt{FCNet}\left(\mathbf{x}^{\mathrm{SE/SP}}_i\right)\right)^{\top} \left(\texttt{FCNet}\left(\mathbf{x}^{\mathrm{CE/CP}}_j\right)\right). \tag{9}$$

The pairwise similarities, which quantify the connections between scene nodes and commonsense nodes, are used to update the weights of the bridge edges after each iteration. Explicitly, the weights of the bridge edges $\mathcal{E}_{\mathrm{B}}$ are updated by:

$$\mathbf{w}^{\mathrm{SE}\leftrightarrow\mathrm{CE}}_{ij} \leftarrow \frac{\exp\left(\mathrm{sim}\left(\mathbf{x}^{\mathrm{SE}}_i, \mathbf{x}^{\mathrm{CE}}_j\right)\right)}{\sum_{j'} \exp\left(\mathrm{sim}\left(\mathbf{x}^{\mathrm{SE}}_i, \mathbf{x}^{\mathrm{CE}}_{j'}\right)\right)}, \quad \mathbf{w}^{\mathrm{SP}\leftrightarrow\mathrm{CP}}_{ij} \leftarrow \frac{\exp\left(\mathrm{sim}\left(\mathbf{x}^{\mathrm{SP}}_i, \mathbf{x}^{\mathrm{CP}}_j\right)\right)}{\sum_{j'} \exp\left(\mathrm{sim}\left(\mathbf{x}^{\mathrm{SP}}_i, \mathbf{x}^{\mathrm{CP}}_{j'}\right)\right)}, \tag{10}$$

where $\mathbf{w}^{\mathrm{SE}\leftrightarrow\mathrm{CE}}_{ij}$ represents the weight of a bi-directional edge connecting a specific pair of SE and CE nodes, and $\mathbf{w}^{\mathrm{SP}\leftrightarrow\mathrm{CP}}_{ij}$ denotes the edge weight between a pair of SP and CP nodes, respectively.

### 3.5 Hierarchical Inference

After $t$ steps of message propagation, we can leverage the node representations from both graphs to infer the unknown class of SE/SP nodes. Specifically, we first compute the similarities between the node representations of each SP node and the three CS nodes within the hierarchical knowledge graph to determine the superclass probabilities, which can be written as

$$p\left(\mathcal{C}^{\mathrm{SP}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right) = \texttt{Softmax}\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_k^{\mathrm{CS}}\right)\right). \tag{11}$$

Here, $k$ denotes the superclass predicate indices, $\mathcal{C}^{\mathrm{SP}} \in \{\mathrm{geo}, \mathrm{pos}, \mathrm{sem}\}$ represents the superclass categories, and $\mathrm{sim}(\cdot, \cdot)$ is defined according to Equation (9).

Once we have classified the superclass for each unknown predicate node in the scene graph, we then examine the conditional probability $p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}, \mathcal{C}^{\mathrm{SP}}\right)$, *i.e.*, the probability of subclass predicates given the superclass. This probability can be computed as follows:

$$p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}, \mathcal{C}^{\mathrm{SP}}\right) = \texttt{Softmax}\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_j^{\mathrm{CP}}\right)\right), \tag{12}$$

where $j$ denotes the subclass predicate indices in the given superclass $\mathcal{C}^{\mathrm{SP}}$.

In general, given an unknown predicate node, the predicted probability of each predicate category can be computed by the superclass probability $p\left(\mathcal{C}^{\mathrm{SP}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right)$ multiplying by the conditional subclass probability $p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}, \mathcal{C}^{\mathrm{SP}}\right)$:

$$p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right) = p\left(\mathcal{C}^{\mathrm{SP}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right) \cdot p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}, \mathcal{C}^{\mathrm{SP}}\right). \tag{13}$$

### 3.6 Hierarchical Semantic Adjustment

Due to the inherent bias in the Visual Genome [23] dataset, most existing SGG models tend to favor commonly occurring predicate classes. In this work, we integrate a hierarchical semantic adjustment mechanism into our model to mitigate biases in predicate classes. This enhancement aims to predict more specific and informative predicates (*e.g.*, `riding on`, `standing on`), as opposed to general ones (*e.g.*, `on`). Essentially, our goal is to find transitioning probabilities $\mathbb{P}(\mathcal{C}_s^{\mathrm{P}} | \mathcal{C}_g^{\mathrm{P}})$ that can convert a general prediction into a more specific one for the predicate classes.

Specifically, we introduce 4 transitioning probability matrices for superclass and subclass classification, denoted as $\mathcal{T}_{\mathrm{sc}}$, $\mathcal{T}_{\mathrm{geo}}$, $\mathcal{T}_{\mathrm{pos}}$, and $\mathcal{T}_{\mathrm{sem}}$. We adopt the predicate confusion matrix generated by the MotifNet [3] baseline as initialization for $\mathcal{R}_{\mathrm{sc}}$ , $\mathcal{R}_{\mathrm{geo}}$, $\mathcal{R}_{\mathrm{pos}}$, and $\mathcal{R}_{\mathrm{sem}}$. We then create transitioning probability matrices by row-normalizing the diagonal-augmented confusion matrix:

$$\mathcal{T}_\gamma = \texttt{RowNormalize}(\mathcal{R}_\gamma + I), \quad \gamma \in \{\mathrm{sc}, \mathrm{geo}, \mathrm{pos}, \mathrm{sem}\}, \tag{14}$$

where $I$ represents an identity matrix of the same size as the confusion matrix $\mathcal{R}_\gamma$. These transition probability matrices can be utilized in the computation of both the superclass probability and the conditional subclass probabilities as expressed by the following equations:

$$p\left(\mathcal{C}^{\mathrm{SP}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right) = \texttt{Softmax}\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_k^{\mathrm{CS}}\right) \cdot \mathcal{T}_{\mathrm{sc}}\right) = \frac{\exp\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_k^{\mathrm{CS}}\right) \cdot \mathcal{T}_{\mathrm{sc}}\right)}{\sum_{k'} \exp\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_{k'}^{\mathrm{CS}}\right) \cdot \mathcal{T}_{\mathrm{sc}}\right)}, \tag{15}$$

$$p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}, \mathcal{C}^{\mathrm{SP}}\right) = \texttt{Softmax}\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_j^{\mathrm{CP}}\right) \cdot \mathcal{T}_{\mathcal{C}^{\mathrm{SP}}}\right) = \frac{\exp\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_j^{\mathrm{CP}}\right) \cdot \mathcal{T}_{\mathcal{C}^{\mathrm{SP}}}\right)}{\sum_{j'} \exp\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_{j'}^{\mathrm{CP}}\right) \cdot \mathcal{T}_{\mathcal{C}^{\mathrm{SP}}}\right)}. \tag{16}$$

Combining this adjustment with our hierarchical inference process, we can rewrite Equation (13) as

$$p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right) = \texttt{Softmax}\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_k^{\mathrm{CS}}\right) \cdot \mathcal{T}_{\mathrm{sc}}\right) \cdot \texttt{Softmax}\left(\mathrm{sim}\left(\mathbf{x}_i^{\mathrm{SP}}, \mathbf{x}_j^{\mathrm{CP}}\right) \cdot \mathcal{T}_{\mathcal{C}^{\mathrm{SP}}}\right). \tag{17}$$

During the training stage, we update our parameters using the following two loss terms to supervise both the superclass and subclass predictions:

$$\mathcal{L}_{\mathrm{SP}} = \texttt{NLL Loss}\left(p\left(\mathcal{C}^{\mathrm{SP}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right), \texttt{OneHot}\left(\mathcal{C}_{\mathrm{GT}}^{\mathrm{SP}}\right)\right), \tag{18}$$

$$\mathcal{L}_{\mathrm{P}} = \texttt{NLL Loss}\left(p\left(\mathcal{C}^{\mathrm{P}} | \mathcal{N}_{\mathrm{SP}}^{\mathsf{unk}}\right), \texttt{OneHot}(\mathcal{C}_{\mathrm{GT}}^{\mathrm{P}})\right), \tag{19}$$

where $\mathcal{C}_{\mathrm{GT}}^{\mathrm{SP}}$ and $\mathcal{C}_{\mathrm{GT}}^{\mathrm{P}}$ represent labels for the superclass and subclass predicates, respectively.

# 4 Experiments

Following the literature, we use the large-scale Visual Genome benchmark [23] to evaluate our method. Our results indicate that HiKER-SGG excels beyond state-of-the-art models with superior performance on both clean and corrupted images. Additionally, we conduct an ablation study to quantitatively delineate the contributions of the hierarchical knowledge and hierarchical prediction head to our model's efficacy.

## 4.1 Experimental Settings

**Dataset.** We conduct extensive experiments using the widely recognized Visual Genome [23] dataset, which encompasses a total of 108,077 images, each annotated with objects and relations. Following previous work [4], we filter the dataset to use the most frequent 150 object classes and 50 predicate classes for experiments. Zellers *et al.* [3] further categorized these 50 predicate classes into 3 superclasses, namely `geometric`, `possessive`, and `semantic`, comprising 228k, 188k, and 39k images, respectively. We use the 3 predicate superclasses to build our hierarchical knowledge graph.

**Natural Corruptions.** We introduce four types of corruptions, sunlight glare, waterdrop, smoke, and dust to simulate realistic corruptions that may occur in real-world scenarios, thereby providing insights into the models' robustness under various corruption conditions. Note that all the models tested are only trained on clean images and employed directly on the corrupted images.

**Tasks.** Following previous work [41, 42], we assess the effectiveness of our proposed approach in the context of two standard SGG tasks: Predicate Classification (PredCls) and Scene Graph Classification (SGCls). In the PredCls scenario, our model is provided with ground-truth bounding boxes and their associated object classes, with the sole task of predicting the predicate class. In the SGCls scenario, the model is only provided with known bounding boxes while the object classes are treated as unknown, and our SGG model is required to predict both the object and predicate classes.

**Evaluation Metrics.** We evaluate the performance of the SGG models by top-$k$ mean triplet recall (mR@$k$) metric on both the PredCls and SGCls tasks. In specific, mR is the average recall score between the top-$k$ predicted triplets and ground-truth ones across all 50 predicate categories, which promotes unbiased prediction for less frequently occurring predicate classes. A subject-predicate-object triplet is considered a match when all three components are correctly classified, and the subject and object bounding boxes align with an IoU (Intersection over Union) score of at least 0.5. In our experiments, we report the mean recall on $k = 20, 50, 100$ to comprehensively evaluate the effectiveness of our method. We also report the constrained (C) and unconstrained (UC) performance results, depending on the presence or absence of the graph constraint. This constraint restricts our SGG model to predict only a single relation between each pair of objects.

**Implementation Details.** We use the Faster-RCNN [55] as the object detector, which is based on VGG-16 [56] backbone provided by Zellers *et al.* [3]. Regarding `FCNet` in Equations (6) and (9), we follow GB-Net [41] to use 3-layer fully connected networks with `ReLU` activation. We set the message propagation steps $t = 3$ and use a 1024-dimensional vector to represent each node. The transitioning probability matrices are frozen during the training and inference stages. We also adopt the BPL [37] method to train our SGG model with unbiased data. In our experiments, we train our model for 30 epochs, initializing the learning rate at $1 \times 10^{-4}$. This learning rate will decrease to $1/10$ of its value after every 10 epochs. A single NVIDIA Quadro RTX 6000 GPU is used for training the SGG model.

**Baselines.** We compare our performance with the following state-of-the-art SGG methods: IMP+ [4], Neural Motifs [3], VCTree [62], PCPL [34], G2S [37], HierMotifs [63], MotifNet + DLFE [64], CogTree [65], SQUAT [66]. Additionally, we compare our approach with SGG methods that are knowledge graph-based, which are closely related to our work: GB-Net [41] and EB-Net + EOA [42]. For a fair comparison, we present the performance results of these baseline methods directly from their respective original papers.

## 4.2 Results and Discussions

**Quantitative Results.** In Table 1, we report our performance results for the PredCls task and SGCls tasks on clean images in the Visual Genome [23] dataset. With the hierarchical predicate prediction paradigm, our method consistently outperforms the knowledge graph-based GB-Net [41] and EB-Net

Table 1: Performance comparison with the state-of-the-art SGG methods on the Visual Genome [23] dataset. The best results for each metric are in **bold**, while the second-best results are underlined.

| Model | PredCls | | | SGCls | | |
|---|---|---|---|---|---|---|
| | mR@20: UC/C | mR@50: UC/C | mR@100: UC/C | mR@20: UC/C | mR@50: UC/C | mR@100: UC/C |
| IMP+ [4] | - / - | 20.3 / 9.8 | 28.9 / 10.5 | - / - | 12.1 / 9.8 | 16.9 / 10.5 |
| Neural Motifs [3] | - / 10.8 | 24.8 / 14.0 | 37.3 / 15.3 | - / 6.3 | 13.5 / 7.7 | 19.6 / 8.2 |
| VCTree [62] | - / 14.0 | - / 17.9 | - / 19.4 | - / 8.2 | - / 10.1 | - / 10.8 |
| PCPL [34] | - / - | 50.6 / 35.2 | 62.6 / 37.8 | - / - | 26.8 / 18.6 | 32.8 / 19.6 |
| G2S: Transformer [37] | - / 26.7 | - / 31.9 | - / 34.2 | - / 15.7 | - / 18.5 | - / 19.4 |
| G2S: MotifNet [37] | - / 24.8 | - / 29.7 | - / 31.7 | - / 14.0 | - / 16.5 | - / 17.5 |
| G2S: VCTree [37] | - / 26.2 | - / 30.6 | - / 32.6 | - / **17.2** | - / **20.1** | - / **21.2** |
| HierMotifs [63] | - / 21.5 | - / 25.5 | - / 26.8 | - / 12.6 | - / 14.9 | - / 15.9 |
| MotifNet + DLFE [64] | - / 22.1 | - / 26.9 | - / 28.8 | - / 12.8 | - / 15.2 | - / 15.9 |
| CogTree [65] | - / 22.9 | - / 28.4 | - / 31.0 | - / 13.0 | - / 15.7 | - / 16.7 |
| SQUAT [66] | - / 25.6 | - / 30.9 | - / 33.4 | - / 14.4 | - / 17.5 | - / 18.8 |
| GB-Net [41] | 23.8 / 15.3 | 41.1 / 19.3 | 55.4 / 20.9 | 13.1 / 7.9 | 21.4 / 9.6 | 29.1 / 10.2 |
| EB-Net + EOA [42] | 39.8 / 30.8 | 54.9 / 36.7 | 66.3 / **39.2** | 19.6 / 14.9 | 26.7 / 17.3 | 32.5 / 18.3 |
| **HiKER-SGG (Ours)** | **41.6 / 32.9** | **57.3 / 37.5** | **68.1** / 38.6 | **20.8** / 15.3 | **27.7** / 19.0 | **33.7** / 19.5 |

Table 2: Performance comparison with the state-of-the-art SGG methods on the Visual Genome [23] dataset with 4 different corruptions. The best results for each metric are in **bold**.

| Model | PredCls | | | SGCls | | |
|---|---|---|---|---|---|---|
| | mR@20: UC/C | mR@50: UC/C | mR@100: UC/C | mR@20: UC/C | mR@50: UC/C | mR@100: UC/C |
| *Corruption: Sunlight glare* | | | | | | |
| GB-Net [41] | 16.7 / 11.1 | 29.5 / 14.5 | 42.5 / 16.1 | 6.7 / 4.2 | 11.7 / 5.0 | 17.3 / 5.5 |
| EB-Net + EOA [42] | 30.6 / 24.0 | 45.4 / 29.2 | 56.4 / 31.4 | 10.3 / 6.7 | 14.6 / 10.2 | 19.1 / 10.4 |
| **HiKER-SGG (Ours)** | **33.5 / 26.3** | **48.4 / 32.1** | **59.5 / 33.8** | **12.2 / 7.5** | **16.1 / 12.5** | **21.7 / 12.9** |
| *Corruption: Waterdrop* | | | | | | |
| GB-Net [41] | 17.8 / 11.2 | 32.3 / 14.6 | 46.1 / 16.1 | 7.3 / 4.7 | 11.9 / 5.3 | 16.5 / 5.5 |
| EB-Net + EOA [42] | 30.7 / 23.4 | 45.5 / 28.6 | 57.6 / 31.0 | 10.7 / 6.8 | 14.8 / 9.8 | 20.3 / 10.8 |
| **HiKER-SGG (Ours)** | **33.7 / 26.5** | **49.1 / 31.3** | **60.9 / 32.7** | **12.3 / 7.8** | **16.0 / 11.6** | **21.9 / 12.4** |
| *Corruption: Smoke* | | | | | | |
| GB-Net [41] | 16.0 / 10.5 | 28.7 / 13.6 | 41.2 / 15.0 | 6.3 / 3.9 | 11.1 / 4.5 | 15.9 / 4.9 |
| EB-Net + EOA [42] | 33.7 / 25.6 | 48.4 / 30.8 | 59.0 / 31.8 | 10.2 / 7.4 | 15.3 / 10.2 | 20.6 / 11.3 |
| **HiKER-SGG (Ours)** | **36.8 / 28.3** | **52.1 / 31.5** | **60.7 / 32.7** | **13.3 / 8.9** | **16.9 / 12.1** | **22.1 / 13.5** |
| *Corruption: Dust* | | | | | | |
| GB-Net [41] | 18.5 / 12.1 | 32.8 / 15.4 | 45.9 / 17.0 | 6.8 / 4.9 | 11.5 / 5.5 | 17.2 / 5.6 |
| EB-Net + EOA [42] | 30.1 / 22.9 | 44.6 / 27.2 | 54.8 / 29.6 | 9.8 / 6.4 | 14.2 / 9.5 | 19.6 / 10.7 |
| **HiKER-SGG (Ours)** | **32.2 / 24.4** | **46.5 / 28.7** | **57.4 / 30.1** | **12.3 / 8.1** | **16.7 / 10.9** | **21.1 / 13.6** |

+ EOA [42] methods. When compared with other state-of-the-art SGG methods, our HiKER-SGG still achieves competitive performance in terms of mean recall. These convicining results demonstrate the effectiveness of our hierarchical predicate prediction method.

We also show our results on 4 corrupted scenarios introduced in Section 4.1 in Table 2 to demonstrate our method also generalizes well to unseen real-world corruptions. Table 2 illustrates that our method achieves an average improvement of around 3% for the PredCls task and 2% for the SGCls task, across all six metrics for all 4 types of corruption. Moreover, relative to the clean image benchmark, our method exhibits a lower percentage of performance degradation, showcasing our model's resilience in handling such corrupted scenarios. For instance, in the presence of smoke corruption, our mR@100, when considering graph constraints, experiences an 5.9% reduction, dropping from 38.6% to 32.7%. In comparison, the EB-Net [42] method shows a greater 7.4% degradation, decreasing from 39.2% to 31.8%.

**Qualitative Results.** To provide further insights into the effectiveness of our method, we visualize some scene graphs generated by our method and the baseline GB-Net [41] method in Figure 2. In the top row of the image, we can observe the scene graphs generated by both methods alongside the ground-truth labels. Notably, while GB-Net tends to predict more general predicate classes (*e.g.*, `on`), our method accurately predicts the ⟨`train-has-engine`⟩ and ⟨`logo-in-train`⟩ triplets. In the bottom row, we illustrate the SGG results under sunlight glare corruption obtained by both methods. In this challenging scenario, our proposed approach employs a hierarchical paradigm: it first identifies the superclass to minimize interference from unrelated superclasses before concentrating on the subclass classification. For instance, when classifying the predicate between `engine` and `track`, non-hierarchical approaches such as GB-Net [41], struggles to detect the relation since the region feature is corrupted. In comparison, our method firstly determines that the relation is `geometric` rather than directly proceeding to subclass classification. This strategy enhances the robustness of our proposed method, enabling it to consistently generate the same scene graph as in clean images.

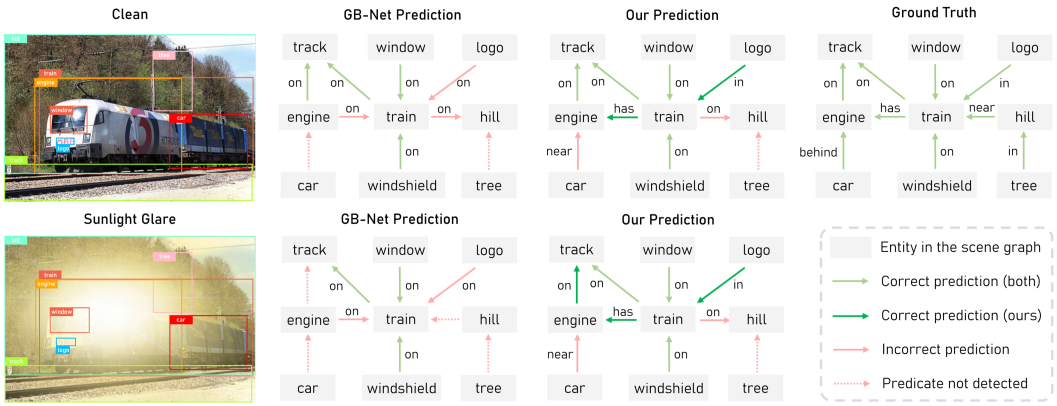


Figure 2: Qualitative comparisons of our proposed HiKER-SGG method with GB-Net [41] baseline method on the PredCls task. Only detected boxes overlapped with GT are shown. The visualized predicted predicates are picked from the top 50 predicted triplets of SGG models.

Table 3: Ablation studies on the PredCls and SGCls tasks on the Visual Genome [23] dataset.

| Setting | PredCls | | SGCls | |
|---|---|---|---|---|
| | mR@20: UC/C | mR@50: UC/C | mR@20: UC/C | mR@50: UC/C |
| *Clean images* | | | | |
| **HiKER-SGG (Ours)** | **41.6 / 32.9** | **57.3 / 37.5** | **20.8 / 15.3** | **27.7 / 19.0** |
| *w/o* superclass transition $\mathcal{T}_{\text{sc}}$ | 41.3 / 32.4 | 56.9 / 37.2 | 20.3 / 15.2 | 27.6 / 18.7 |
| *w/o* superclass loss $\mathcal{L}_{\text{SP}}$ | 40.5 / 31.7 | 55.8 / 36.7 | 20.1 / 15.2 | 27.3 / 17.9 |
| *w/o* superclass nodes $\mathcal{N}_{\text{CS}}$ | 39.8 / 30.8 | 54.9 / 36.7 | 19.6 / 14.9 | 26.7 / 17.3 |
| *Corrupted images* | | | | |
| **HiKER-SGG (Ours)** | **34.1 / 26.4** | **49.0 / 30.9** | **12.5 / 8.1** | **16.4 / 11.8** |
| *w/o* superclass transition $\mathcal{T}_{\text{sc}}$ | 33.5 / 25.8 | 48.3 / 30.6 | 12.1 / 7.8 | 15.7 / 11.5 |
| *w/o* superclass loss $\mathcal{L}_{\text{SP}}$ | 32.6 / 24.7 | 47.0 / 29.7 | 11.6 / 7.3 | 14.9 / 10.8 |
| *w/o* superclass nodes $\mathcal{N}_{\text{CS}}$ | 31.9 / 23.8 | 46.3 / 29.0 | 10.7 / 6.9 | 14.4 / 10.0 |

### 4.3 Ablation Studies

To systematically evaluate the effectiveness of our proposed HiKER-SGG, we conduct an ablation study on the Visual Genome [23] dataset to analyze the impacts of different components on both clean and corrupted images. For corrupted images, the results are averaged across the four distinct corruptions discussed in Section 4.1. We systematically exclude each component one by one and present the corresponding performance results in Table 3. The superclass transitioning probability matrices, $\mathcal{T}_{\text{sc}}$, are designed to counteract category bias inherent in the training data. Excluding this component would hinder the ability of our SGG model to make accurate predictions towards specific predicates that occur less frequently. The superclass loss, $\mathcal{L}_{\text{SP}}$ supervises the superclass classification. Removing it could result in imprecise probability estimations for the superclass, thereby affecting the efficacy of our hierarchical approach. Both the superclass transition and superclass loss contribute to the hierarchical prediction head. Moreover, the superclass nodes, $\mathcal{N}_{\text{CS}}$, account for the hierarchical component in the external knowledge. In all, both the hierarchical knowledge and the hierarchical prediction head contribute to the effectiveness of HiKER-SGG.

## 5 Conclusion

To comprehend visual scenes that may contain common natural corruptions in the real world, we propose a novel framework, **Hi**erarchical **K**nowledge **E**nhanced **R**obust **S**cene **G**raph **G**eneration (HiKER-SGG). This framework is designed to be agnostic to the corruption types and is capable of robustly generating scene graphs under various conditions of corruption. HiKER-SGG utilizes hierarchical knowledge derived from external sources and employs a hierarchical inference process, serving as an algorithmic prior during decision-making, to reason and correct potential inaccuracies introduced by off-the-shelf detectors. Through extensive experiments, we have demonstrated that HiKER-SGG outperforms the state-of-the-art models in performance on both clean and corrupted images.

## Acknowledgement

This work has been funded in part by the Army Research Laboratory (ARL) under grant W911NF-23-2-0007 and W911NF-19-2-0146, and the Air Force Office of Scientific Research (AFOSR) under grants FA9550-18-1-0097 and FA9550-18-1-0251.

## References

[1] Xiaojun Chang, Pengzhen Ren, Pengfei Xu, Zhihui Li, Xiaojiang Chen, and Alex Hauptmann. A comprehensive survey of scene graphs: Generation and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):1–26, 2021. 1, 2

[2] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1261–1270, 2017. 1, 2

[3] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, 2018. 1, 6, 7, 8

[4] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5419, 2017. 1, 3, 7, 8

[5] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *European Conference on Computer Vision*, pages 670–685, 2018. 1, 3

[6] Alakh Desai, Tz-Ying Wu, Subarna Tripathi, and Nuno Vasconcelos. Learning of visual relations: The devil is in the tails. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15404–15413, 2021. 1

[7] Mohammed Suhail, Abhay Mittal, Behjat Siddiquie, Chris Broaddus, Jayan Eledath, Gerard Medioni, and Leonid Sigal. Energy-based learning for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13936–13945, 2021. 1, 3

[8] Eric Mintun, Alexander Kirillov, and Saining Xie. On interaction between augmentations and corruptions in natural corruption robustness. In *Advances in Neural Information Processing Systems*, volume 34, pages 3571–3583, 2021. 1, 3

[9] Nicholas Gray, Megan Moraes, Jiang Bian, Alex Wang, Allen Tian, Kurt Wilson, Yan Huang, Haoyi Xiong, and Zhishan Guo. Glare: A dataset for traffic sign detection in sun glare. *IEEE Transactions on Intelligent Transportation Systems*, 2023. 1

[10] Maximilian Riesenhuber and Tomaso Poggio. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2(11):1019–1025, 1999. 1

[11] Johannes Bill, Hrag Pailian, Samuel J Gershman, and Jan Drugowitsch. Hierarchical structure is employed by humans during visual motion perception. *Proceedings of the National Academy of Sciences*, 117(39):24581–24589, 2020. 1

[12] Geoffrey E Hinton. Mapping part-whole hierarchies into connectionist networks. *Artificial Intelligence*, 46(1-2):47–75, 1990. 2

[13] Lior Wolf, Stan Bileschi, and Ethan Meyers. Perception strategies in hierarchical vision systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2153–2160. IEEE, 2006. 2

[14] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3668–3678, 2015. 3

[15] Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D Manning. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the 4th Workshop on Vision and Language*, pages 70–80, 2015. 3

[16] Brigit Schroeder and Subarna Tripathi. Structured query-based image retrieval using scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 178–179, 2020. 3

[17] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *European Conference on Computer Vision*, pages 684–699, 2018. 3

[18] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10685–10694, 2019. 3

[19] Xiangyang Li and Shuqiang Jiang. Know more say less: Image captioning based on scene graphs. *IEEE Transactions on Multimedia*, 21(8):2117–2130, 2019. 3

[20] Junhua Jia, Xiangqian Ding, Shunpeng Pang, Xiaoyan Gao, Xiaowei Xin, Ruotong Hu, and Jie Nie. Image captioning based on scene graphs: A survey. *Expert Systems with Applications*, page 120698, 2023. 3

[21] Cheng Zhang, Wei-Lun Chao, and Dong Xuan. An empirical study on leveraging scene graphs for visual question answering. In *British Machine Vision Conference*, 2019. 3

[22] Tianwen Qian, Jingjing Chen, Shaoxiang Chen, Bo Wu, and Yu-Gang Jiang. Scene graph refinement network for visual question answering. *IEEE Transactions on Multimedia*, 25:3950–3961, 2023. 3

[23] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73, 2017. 3, 6, 7, 8, 9

[24] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1969–1978, 2019. 3

[25] Bo Dai, Yuqi Zhang, and Dahua Lin. Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3076–3086, 2017. 3

[26] Yikang Li, Wanli Ouyang, Bolei Zhou, Jianping Shi, Chao Zhang, and Xiaogang Wang. Factorizable net: an efficient subgraph-based framework for scene graph generation. In *European Conference on Computer Vision*, pages 335–351, 2018. 3

[27] Wenbin Wang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Exploring context and visual pattern of relationship for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2019. 3

[28] Chao Chen, Yibing Zhan, Baosheng Yu, Liu Liu, Yong Luo, and Bo Du. Resistance training using prior bias: toward unbiased scene graph generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 212–220, 2022. 3

[29] Xingning Dong, Tian Gan, Xuemeng Song, Jianlong Wu, Yuan Cheng, and Liqiang Nie. Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19427–19436, 2022. 3

[30] Xingchen Li, Long Chen, Jian Shao, Shaoning Xiao, Songyang Zhang, and Jun Xiao. Rethinking the evaluation of unbiased scene graph generation. In *British Machine Vision Conference*, 2022. 3

[31] Tao He, Lianli Gao, Jingkuan Song, and Yuan-Fang Li. State-aware compositional learning toward unbiased training for scene graph generation. *IEEE Transactions on Image Processing*, 32:43–56, 2022. 3

[32] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3716–3725, 2020. 3

[33] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11109–11119, 2021. 3

[34] Shaotian Yan, Chen Shen, Zhongming Jin, Jianqiang Huang, Rongxin Jiang, Yaowu Chen, and Xian-Sheng Hua. Pcpl: Predicate-correlation perception learning for unbiased scene graph generation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 265–273, 2020. 3, 7, 8

[35] Xianjing Han, Xingning Dong, Xuemeng Song, Tian Gan, Yibing Zhan, Yan Yan, and Liqiang Nie. Divide-and-conquer predictor for unbiased scene graph generation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(12):8611–8622, 2022. 3

[36] Aniket Agarwal, Ayush Mangal, et al. Visual relationship detection using scene graphs: A survey. *arXiv preprint arXiv:2005.08045*, 2020. 3

[37] Yuyu Guo, Lianli Gao, Xuanhan Wang, Yuxuan Hu, Xing Xu, Xu Lu, Heng Tao Shen, and Jingkuan Song. From general to specific: Informative scene graph generation via balance adjustment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16383–16392, 2021. 3, 7, 8

[38] Arushi Goel, Basura Fernando, Frank Keller, and Hakan Bilen. Not all relations are equal: Mining informative labels for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15596–15606, 2022. 3

[39] Tao He, Lianli Gao, Jingkuan Song, Jianfei Cai, and Yuan-Fang Li. Learning from the scene and borrowing from the rich: tackling the long tail in scene graph generation. In *Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence*, pages 587–593, 2021. 3

[40] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6163–6171, 2019. 3

[41] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Bridging knowledge graphs to generate scene graphs. In *European Conference on Computer Vision*, pages 606–623. Springer, 2020. 3, 4, 5, 7, 8, 9

[42] Zhanwen Chen, Saed Rezayi, and Sheng Li. More knowledge, less bias: Unbiasing scene graph generation with explicit ontological adjustment. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4023–4032, 2023. 3, 7, 8

[43] Yujia Li, Richard Zemel, Marc Brockschmidt, and Daniel Tarlow. Gated graph sequence neural networks. In *International Conference on Learning Representations*, 2016. 3, 5

[44] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2018. 3

[45] Norman Mu and Justin Gilmer. Mnist-c: A robustness benchmark for computer vision. *arXiv preprint arXiv:1906.02337*, 2019. 3

[46] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2020. 3

[47] Yushun Tang, Ce Zhang, Heng Xu, Shuoshuo Chen, Jie Cheng, Luziwei Leng, Qinghai Guo, and Zhihai He. Neuro-modulated hebbian learning for fully test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3728–3738, 2023. 3

[48] Yushun Tang, Qinghai Guo, and Zhihai He. Cross-inferential networks for source-free unsupervised domain adaptation. In *IEEE International Conference on Image Processing*, pages 96–100. IEEE, 2023. 3

[49] Evgenia Rusak, Lukas Schott, Roland S Zimmermann, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, and Wieland Brendel. A simple way to make neural networks robust against diverse image corruptions. In *European Conference on Computer Vision*, pages 53–69, 2020. 3

[50] Charles Herrmann, Kyle Sargent, Lu Jiang, Ramin Zabih, Huiwen Chang, Ce Liu, Dilip Krishnan, and Deqing Sun. Pyramid adversarial training improves vit performance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13419–13429, 2022. 3

[51] Klim Kireev, Maksym Andriushchenko, and Nicolas Flammarion. On the effectiveness of adversarial training against common corruptions. In *Uncertainty in Artificial Intelligence*, pages 1012–1021. PMLR, 2022. 3

[52] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 3

[53] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2019. 3

[54] Yuan Yang, James C Kerce, and Faramarz Fekri. Logicdef: An interpretable defense framework against adversarial examples via inductive scene graph reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8840–8848, 2022. 3

[55] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, volume 28, page 91–99, 2015. 3, 4, 5, 7

[56] K Simonyan and A Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 4, 7

[57] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 4

[58] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 31, page 4444–4451, 2017. 4

[59] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 4

[60] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543, 2014. 4

[61] Kyunghyun Cho, Bart van Merriënboer, Çağlar Gulçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734, 2014. 5

[62] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6619–6628, 2019. 7, 8

[63] Bowen Jiang and Camillo J Taylor. Scene graph generation from hierarchical relationship reasoning. *arXiv preprint arXiv:2303.06842*, 2023. 7, 8

[64] Meng-Jiun Chiou, Henghui Ding, Hanshu Yan, Changhu Wang, Roger Zimmermann, and Jiashi Feng. Recovering the unbiased scene graphs from the biased ones. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1581–1590, 2021. 7, 8

[65] Jing Yu, Yuan Chai, Yujing Wang, Yue Hu, and Qi Wu. Cogtree: Cognition tree loss for unbiased scene graph generation. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pages 1274–1280, 2021. 7, 8

[66] Deunsol Jung, Sanghyun Kim, Won Hwa Kim, and Minsu Cho. Devil's on the edges: Selective quad attention for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18664–18674, 2023. 7, 8