RASC: Relationship-Aware Scene Captioning for Accessibility

Jai Kushwaha[1] · **Caner Gel**[1]2

[1] *Learning From Images, WiSe 2025-26*

Problem Statement

Most accessibility tools provide flat object lists without spatial context. RASC addresses this gap by explicitly modeling spatial relationships between objects and converting them into natural language descriptions. Our three-stage pipeline combines YOLOv8 object detection, Neural Motifs relationship prediction, and T5-based caption generation to produce descriptions like: "A person walking a dog on the left side of a tree-lined street." This approach provides visually impaired users with contextually rich scene understanding for improved navigation and comprehension.

Traditional image captioning models describe what objects are present, but often fail to explain:

- How objects relate to each other
- Spatial arrangements (left of, behind, near)
- Interactions between objects

Three-Stage Pipeline

Stage 1: Detection

YOLOv8 detects 150 object classes with bounding boxes

Stage 2: Relationships

Neural Motifs predicts spatial relations between objects

Stage 3: Caption

T5 generates natural language descriptions

RASC: Relationship-Aware Scene Captioning for Accessibility

Problem & Motivation

Current Limitations

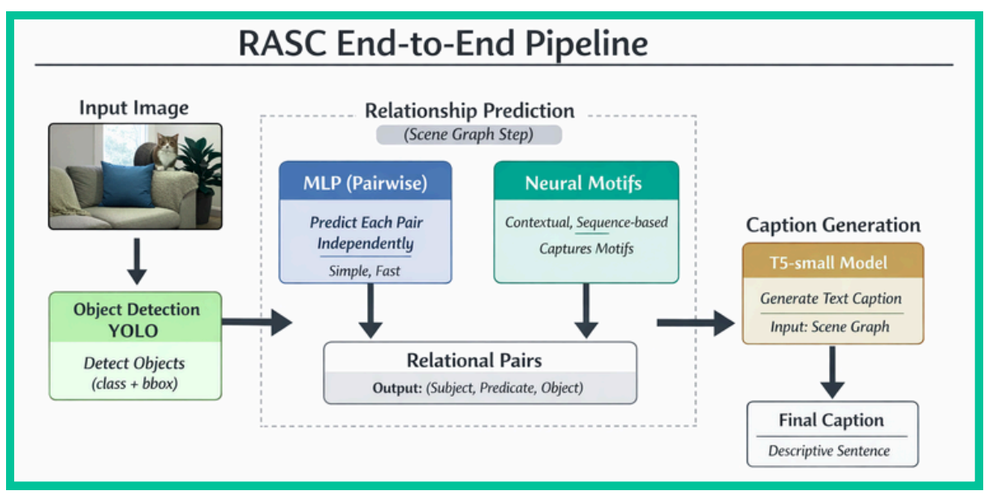
- Existing accessibility tools list objects without context
- Miss critical spatial relationships for navigation
- Generate generic captions lacking situational awareness
- Don't convey "where" objects are relative to each other

Our Solution

RASC explicitly models 10 spatial relationships (left of, right of, on, under, etc.) and converts structured scene graphs into natural, contextually-rich descriptions that support navigation and scene understanding.

Dataset: Visual Genome Subset

- 5,000 images with spatial relationships
- 150 most common object classes
- 10 spatial relationship types
- Split: 70% train / 15% val / 15% test



Method: Three-Stage Architecture

1. Object Detection (YOLOv8)

- Model: YOLOv8-nano (lightweight, fast) (Pretrained on COCO dataset)
- Training: Fine-tuned on Visual Genome
- Output: Bounding boxes + class labels
- Performance: on 150 classes

2. Relationship Prediction (Neural Motifs)

- Architecture:
- Input: Object pairs with bounding boxes
- Relations: left of, right of, on, under, inside, (10 classes)
- Training: 15 epochs, batch size 64
- Performance: F1-score on relationship prediction

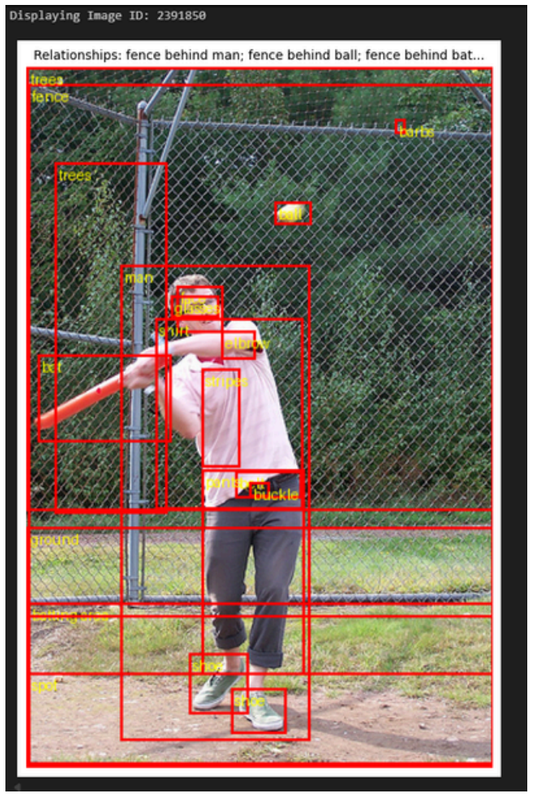
*Neural Motifs for relationship prediction, a motif refers to a repeating or common pattern of interactions between objects in a scene.

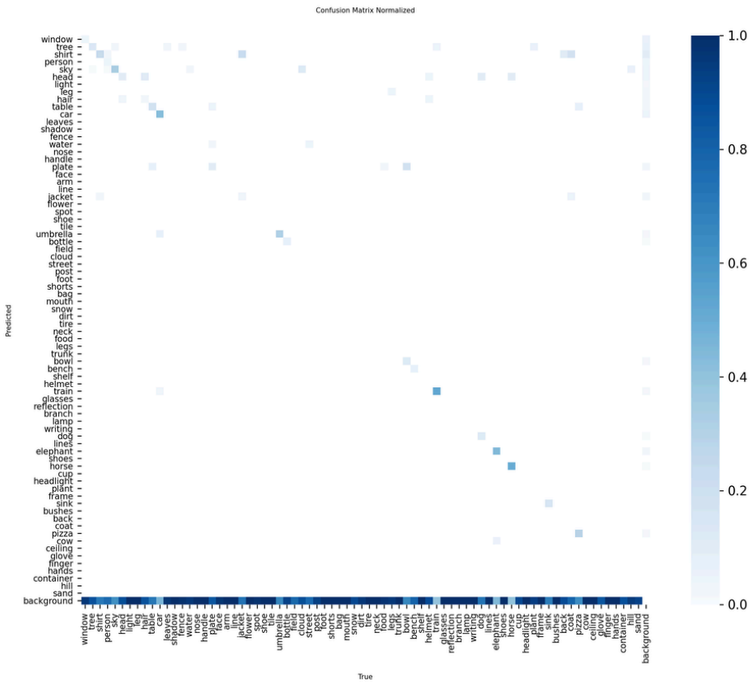
3. Caption Generation (T5-small)

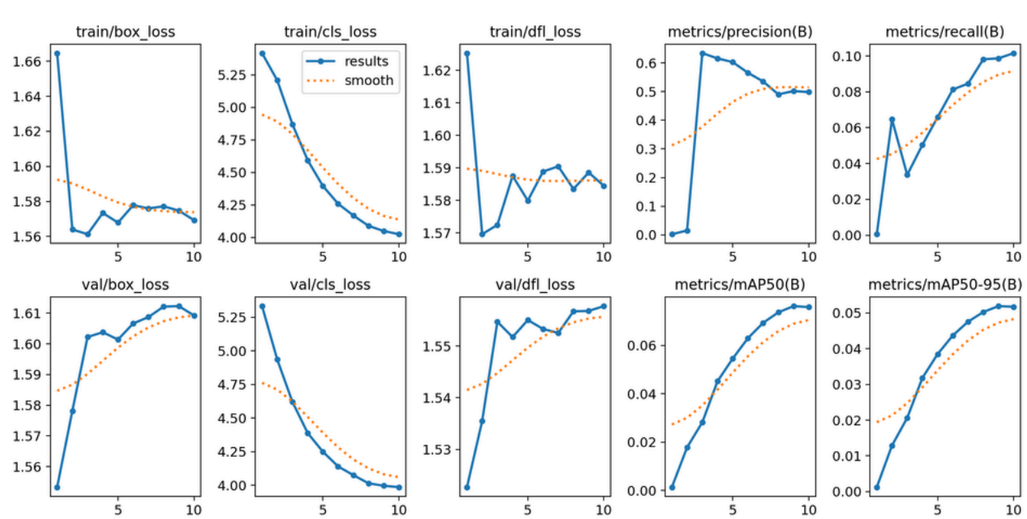
- Model: T5-small transformer (60M parameters)
- Input: Serialized scene graph (text format)
- Output: Natural language caption
- Training: Graph-to-text generation, 5 epochs
- Evaluation: CIDEr, BLEU-4

Results & Evaluation

Yolo Fine Tuning Results







Yolo Fine tuning

Validation

Relationship Pair

1. MLP (Multi-Layer Perceptron) Architecture

Purpose: Predict relationships independently for each object pair.

Input Features per pair:

- Subject class (one-hot or embedding)
- Subject bounding box (normalized)
- Object class (one-hot or embedding)
- Object bounding box (normalized)

Architecture (as inferred from `create_model` usage):

- Concatenate embeddings for subject and object:

$$\text{features} = [s_{\text{emb}}, s_{\text{bbox}}, o_{\text{emb}}, o_{\text{bbox}}]$$

- Feed into **fully connected layers (MLP)**:
  - Hidden layers with dimension `hidden_dim` (e.g., 512)
  - Dropout for regularization (`dropout`, e.g., 0.1)
- Output layer with **size = num_relations** (10 in your config)
- Activation: softmax for relationship classification

Workflow:

1. Each object pair is treated **independently**
2. Predict relationship probabilities for the pair
3. No contextual information from other objects

2. Neural Motifs Architecture

Purpose: Capture **contextual patterns** (motifs) in the entire scene, i.e., relationships often appear together in certain configurations.

Key idea: Some relationships repeat in predictable **motifs**. For example:

- "Person → riding → Horse"
- "Cup → on → Table"
- "Dog → under → Table"

Input Features per pair:

- Same as MLP (class + bbox embeddings)
- **All object embeddings** in the scene are considered to capture **context**

Architecture (from `relationship_models.py`):

1. **Object Embeddings:**
   - Subject embedding: `s_emb = Embedding(s_cls)`
   - Object embedding: `o_emb = Embedding(o_cls)`
   - Concatenate with bounding box info
2. **Contextualization / Motif Extraction:**
   - Apply **BiLSTM or RNN-like layers** over all objects to capture **scene-level context**
   - Learns **co-occurrence patterns** (motifs) in relationships
3. **Relationship Prediction:**
   - For each pair, combine:

$$\text{pair features} = [s_{\text{context}}, o_{\text{context}}]$$

   - Feed into **MLP classifier**
   - Output: probability over `num_relations`

Feature / Aspect	MLP	Neural Motifs
Input per pair	Subject + Object embeddings only	All object embeddings + bboxes
Context awareness	None (pairs independent)	Global scene context (motifs)
Architecture type	Fully connected (MLP)	MLP + BiLSTM / context modules
Speed	Faster	Slightly slower
Accuracy (ambiguous cases)	Lower	Higher
Use case	Simple scenes, obvious relations	Complex scenes, frequent motifs

T5 Captioning

Relationships:

apple
cat
dog man

Generate Caption

Generated Caption:

apple; cat; dog man

Example Relationships

```
obj0 left of obj1
obj1 on top of obj2
obj2 next to obj3
```

Results & Evaluation

T5-Small Caption Generator

Metric	Value
Test Length (tokens)	39,936
Reference Length	46,158
Length Ratio (test/ref)	**0.8652**
Number of Samples	4,225

N-gram	Guess	Correct
1-gram	39,936	39,851
2-gram	35,711	35,587
3-gram	31,486	31,349
4-gram	27,261	27,115

CIDEr usually ranges:
0–1 for weak models
1–3 moderate
3–5 strong
5+ very strong

Metric	Score
CIDEr	**8.887**
BLEU-4	**0.8525**
Model	T5-small
Samples	4,225

Relationship parir Metrics:

Metric	Test	Val	Train
Accuracy	0.632	0.609	0.756
Macro F1	0.446	0.466	0.37
F1 "on top of"	0.861	0.844	0.836
F1 "behind"	0.673	0.73	0.737
F1 "in front of"	0.566	0.556	0.511
F1 "inside"	0.581	0.5	0.533
F1 "under"	0.545	0.714	0.651
F1 "next to"	0.484	0.381	0.318
F1 "over"	0.308	0	0.109
F1 others ("left	0	0	0

Relationship Pair Metrics:

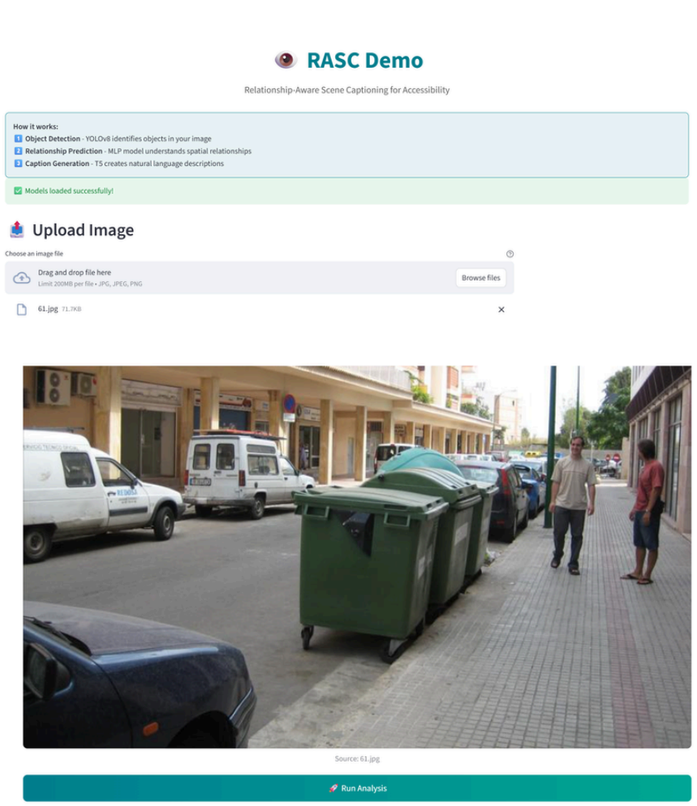
- **Accuracy and F1 show moderate performance.**
- **Some predicates (like "over") are rarely predicted.**

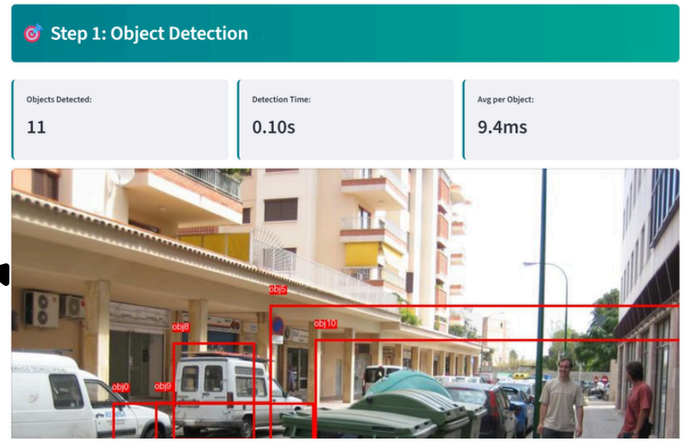
Yolo Prediction Results

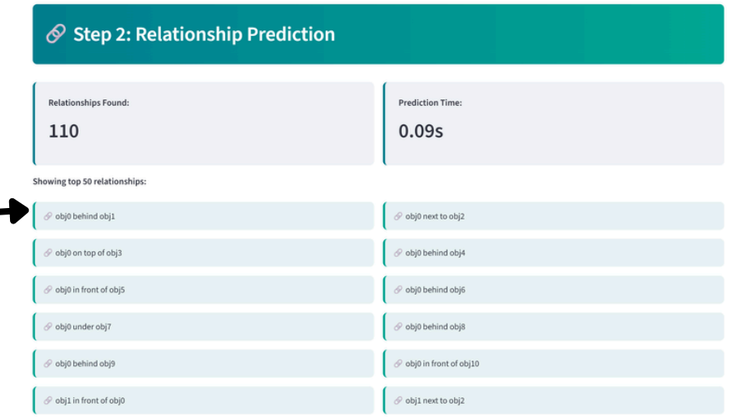
Metric	Value	Notes
mAP@50	0.076	Final evaluation
mAP@50:95	0.052	Final evaluation
Precision	0.501	Final evaluation
Recall	0.1	Final evaluation
Best mAP@50	~0.076	Epoch 10 (peak)
Best mAP@50:95	~0.052	Epoch 10

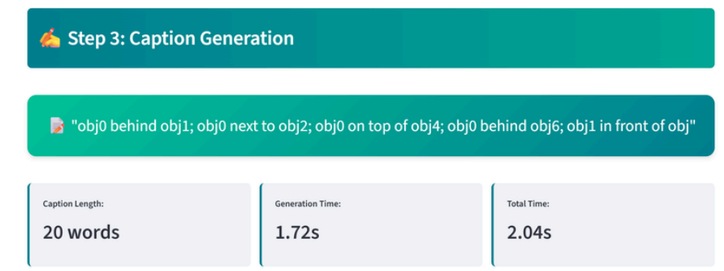
Model plateaus early (~epoch 9–10). Very low recall → few detections.

APP









Comparison: RASC vs. DSGG

Approach	Architecture	Focus
RASC (Ours)	3-stage pipeline	Accessibility + NL
DSGG (CVPR'24)	End-to-end Transformer	Scene graph accuracy
Neural Motifs	LSTM context	Relationship modeling
Image Captioning	Encoder-decoder	General description

References

- Krishna et al., Visual Genome, IJCV 2017
- Zellers et al., Neural Motifs, CVPR 2018
- Raffel et al., T5, JMLR 2020
- Ultralytics, YOLOv8, 2023
- https://zeeshanhayder.github.io/DSGG/

Key Limitations of RASC

Error Propagation from Detection
Object detection mistakes (missed or false detections) directly affect relationship prediction and caption quality.
Pairwise Relationship Modeling Only
Model predicts relationships between object pairs independently — no global scene graph reasoning.
Limited Predicate Vocabulary
Only 10 spatial relationships are supported, restricting semantic richness and scene understanding.
No End-to-End Training
Detection, relationship prediction, and captioning are trained separately — no joint optimization across modules.
Scalability & Efficiency Issues ($O(N^2)$)
Relationship prediction scales quadratically with number of detected objects, making it inefficient for complex scenes.