# Affinity Graph Supervision for Visual Recognition

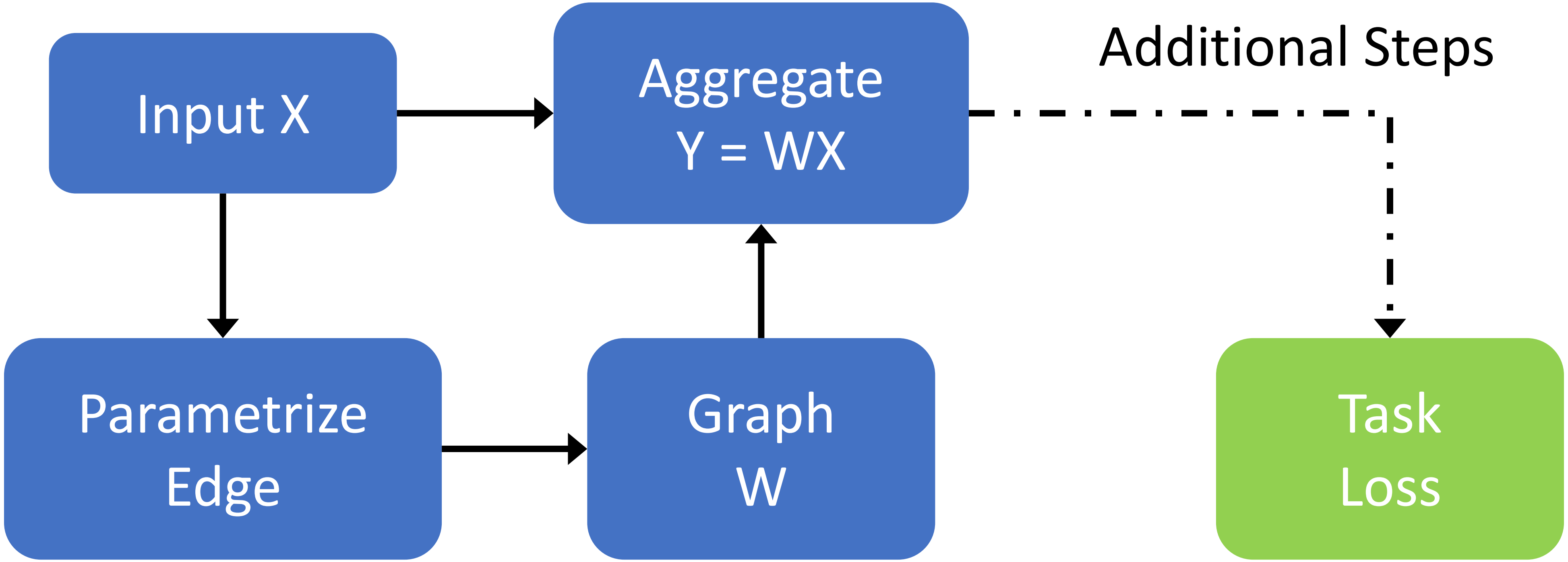
Paper ID: 7437

Chu Wang[1], Babak Samari[1], Vladimir G. Kim[2], Siddhartha Chaudhuri[2,3], Kaleem Siddiqi[1]

[1]McGill University [2]Adobe Research [3]IIT Bombay

# Learnable Graphs in Neural Networks

- **Learnable graphs:** commonly seen in adaptive GCN-like architectures, including but not limited to Self-Attention Mechanism [1] and Graph Attention Networks [2].
- **Parametrized adjacency matrix W:** can be updated during the training of the neural network.
- **Framework illustration:**



# Present Limitations in Graph Learning

- **Parametrized Graph:** comes from edge parametrization functions, which compute edge weights $e_{ij}$ given a pair of input node features $(\vec{h}_i, \vec{h}_j)$. Popular choices are listed below, where α stands for dense layer.
  - Self-Attention Mechanism [1].

$$e_{ij} = \frac{<\alpha_k(\vec{h}_i), \alpha_q(\vec{h}_j)>}{\sqrt{d_k}}$$

  - Graph Attention Networks [2].

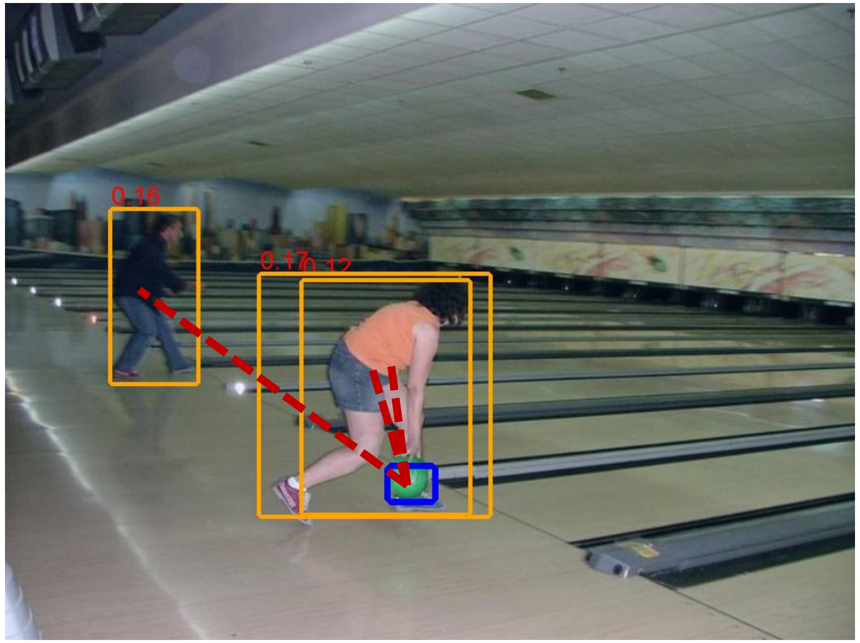
$$e_{ij} = \alpha(concat(W\vec{h}_i, W\vec{h}_j))$$

- **Learning of the parametrized graph :**
  - The graph edges are supervised **only** by the task related loss [1][2][3].

# Present Limitations in Graph Learning

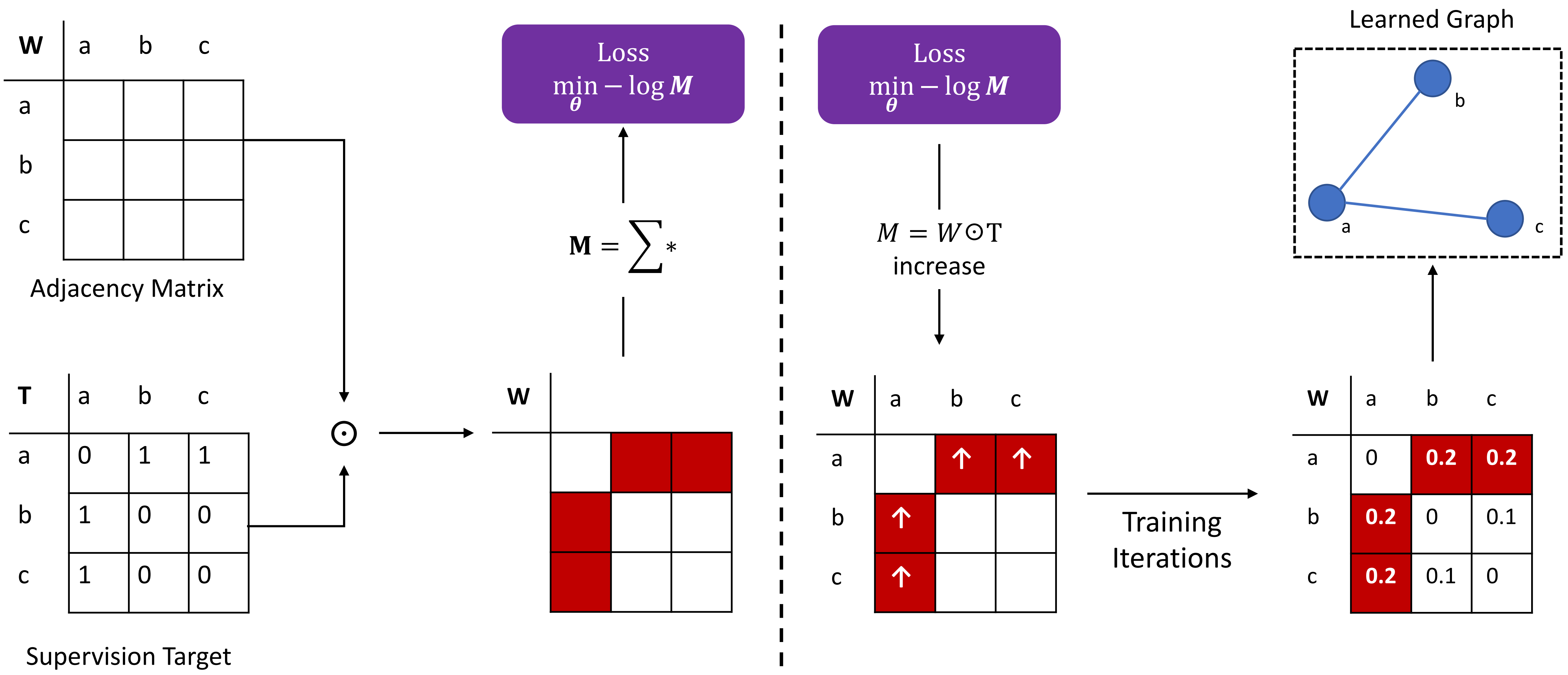
- **Learned Relationships are Not Easy to Interpret:**
  - Edge weights in converged graphs are often ad-hoc.
  - The neural network doesn't care which edges are emphasized, so long as the task related loss is minimized.
  - **<u>We can improve this by additional direct supervision of the graph learning!</u>**


Baseline Attention Nets [3]: ad-hoc edge weight convergence




With additional supervision: reasonable and interpretable edge weights

# A Generic Graph Supervision Method



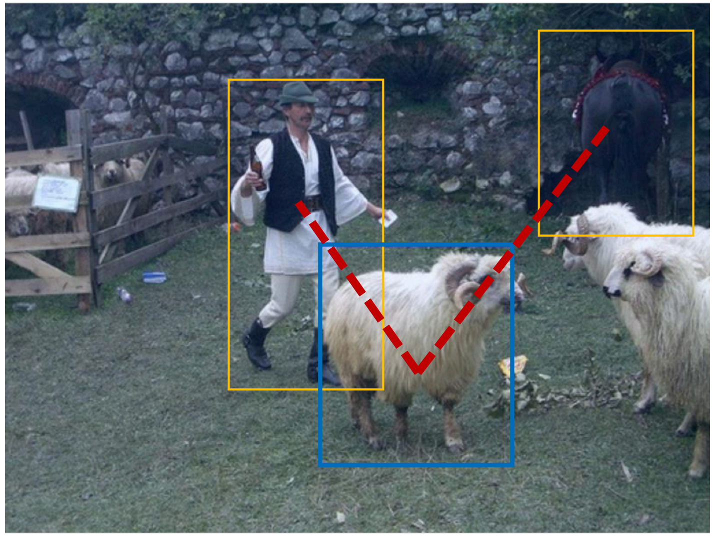

⊙ : element wise product; $\sum *$ : summation over all elements; ↑ : value increase

# Applications: Visual Relationship Learning

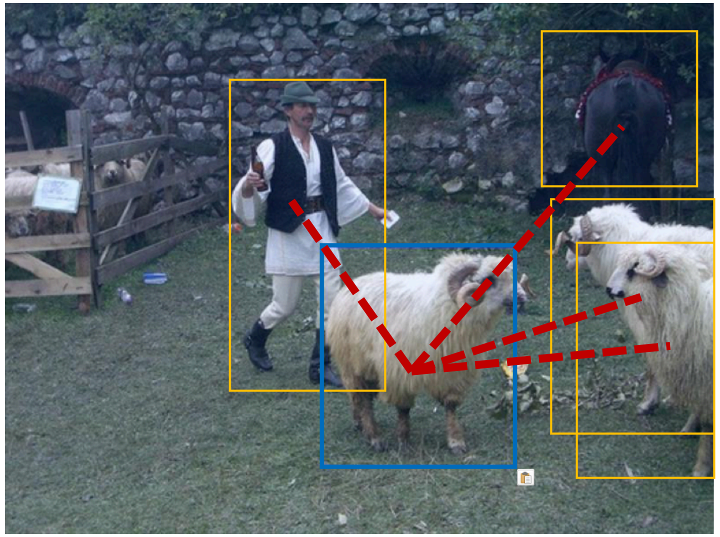
- **Goal:** use the supervision target to direct the learning of object relationships.
- **Supervision target matrix:**

$$T[i,j] = \begin{cases} 1 & if\ (i,j) \in S \\ 0 & otherwise \end{cases}$$

  - $S$ stands for a set of edges that are **chosen by the user.**
  - $(i,j)$ is a pair of <u>region proposals from a Faster-RCNN backbone.</u>



Example 1:
Different *Category* Connections



Example 2:
Different *Instance* Connections

# Applications: Visual Relationship Learning

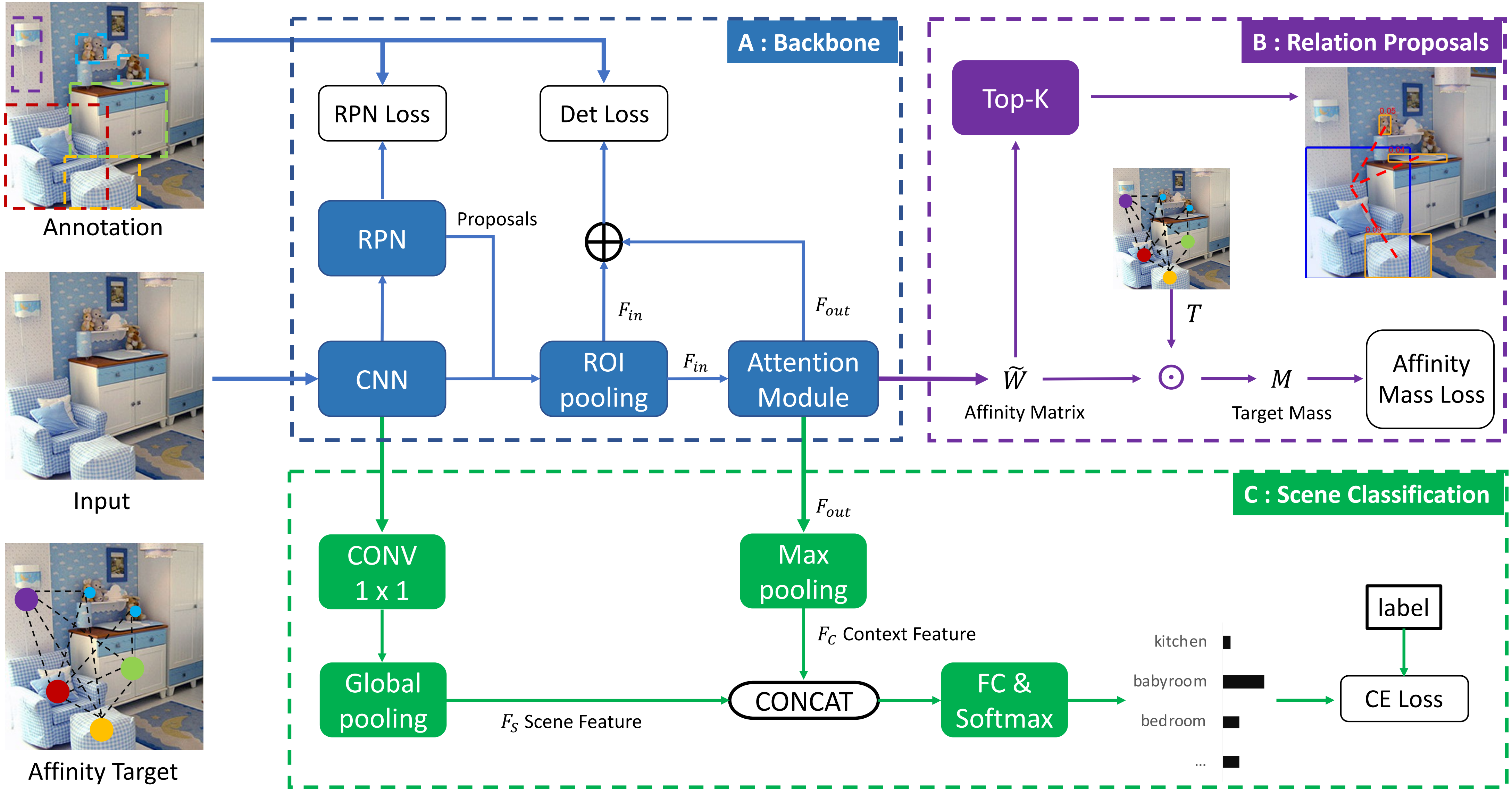


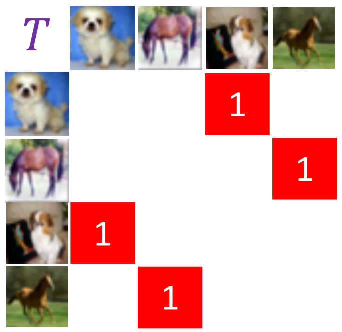

Figure 1. Affinity Graph Supervision in visual attention networks. The **blue** dashed box surrounds the relation network backbone [3]. The **purple** dashed box highlights our component for affinity graph learning and the branch for relationship learning.

# Applications: mini-Batch Training

- **Goal:** to increase feature coherence for examples within the same class and feature separation for examples between different classes.
- **Supervision target matrix:**

$$T[i,j] = \begin{cases} 1 & if\ (i,j) \in S \\ 0 & otherwise \end{cases}$$

  - $S$ stands for a set of edges that are **chosen by the user.**
  - $(i,j)$ is a pair of <u>images in the same batch during standard CNN training.</u>
  - $S = \{\ (i,j) \mid class(i) = class(j)\ \}$
  - Exemplar target in a batch of four images:



# Applications: mini-Batch Training

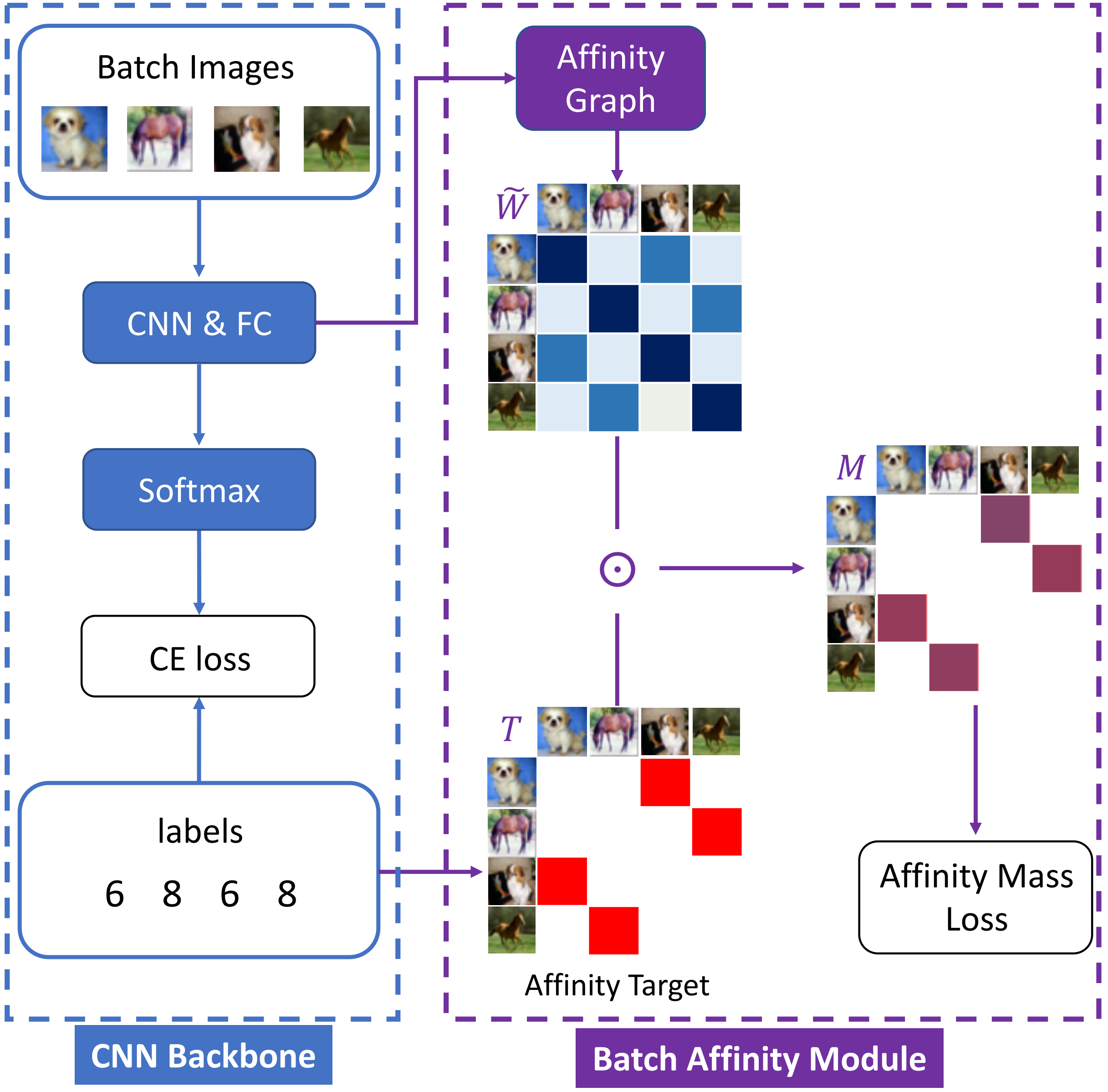


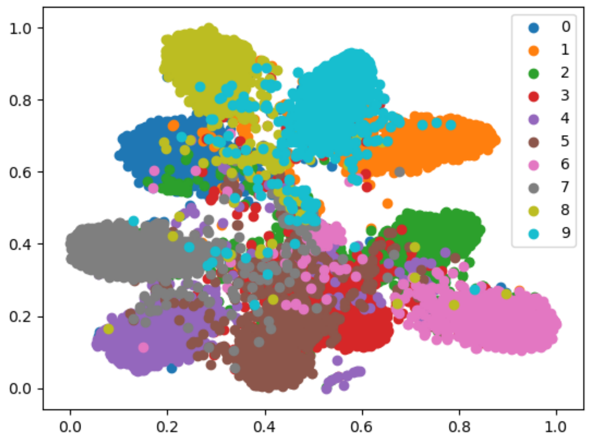

Figure 2. Affinity Graph Supervision in mini-batch training of a CNN.

# Mini Batch Training

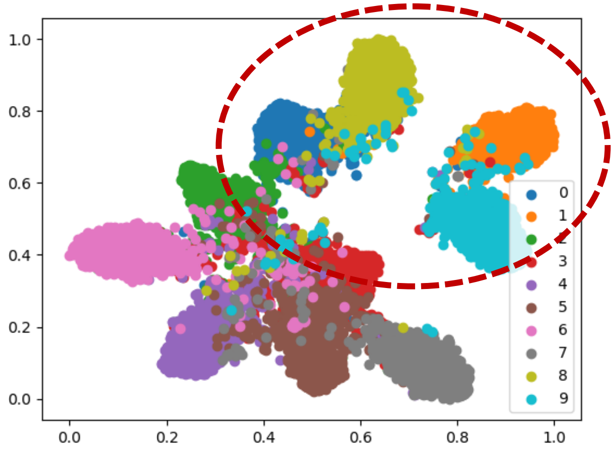
**Results:**

- **1-2% consistent boost in accuracy**
- **Cross-category feature separation:**

baseline



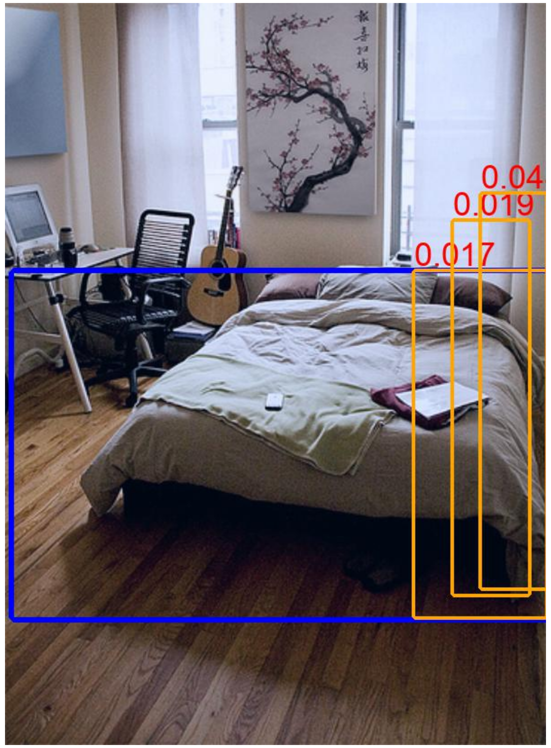

Baseline
+
Affinity Sup

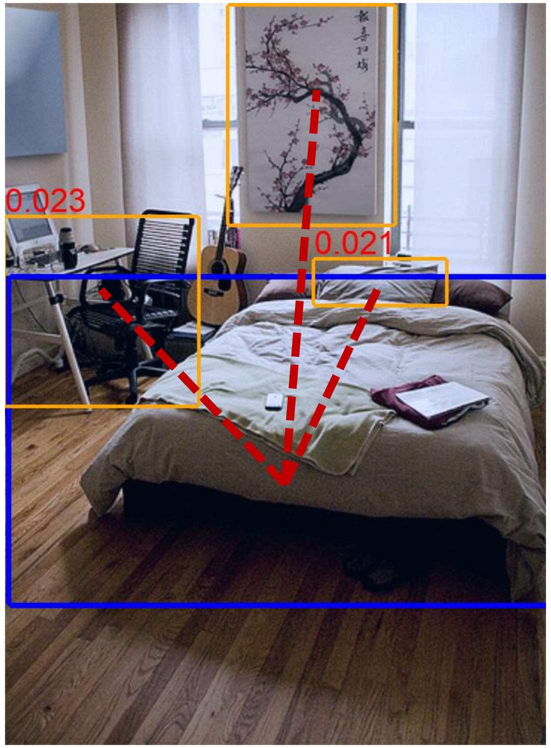



# Visual Relationship Learning

**Results:**

- **25% relative recall boost**
- **Plausible relationship prediction with NO ground truth relationship labels used:**




Relationships between the blue box and the orange boxes are predicted, with weights shown in red.
Left: baseline. Right: baseline + affinity supervision.

# Summary

- Additional applications:
  - Scene categorization.
  - Object detection.
- Contributions
  - Affinity loss: a novel loss function for supervising graph structures.
  - Supervision target: flexible, allowing user control in specific applications.
  - Interpretable graph structure learning in GCN like architectures.

**Please see our paper for further details!**

# References


[1] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems*. 2017.

[2] Veličković, Petar, et al. "Graph attention networks." *arXiv preprint arXiv:1710.10903* (2017). ICLR 2017.

[3] Hu, Han, et al. "Relation networks for object detection." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

[4] Zhang, Ji, et al. "Relationship proposal networks." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

# Appendix

- Affinity Mass Loss Forms.
- Affinity Mass Loss Ablation Study.
- Visual relationship learning results.
- Scene categorization results.
- Mini Batch Training Ablation Studies.
- Mini Batch Training results.
- arXiv version: arxiv.org/abs/2003.09049

# Affinity Mass Loss Forms

## Affinity Mass Loss

- Focal loss form: on the affinity mass $M$, is defined as a negative log likelihood loss, weighted by the focal normalization term. Formally written as:

$$\boldsymbol{L_G = L_{focal}\,(M) = -(1 - M)^r \log(M)}.$$

- The focal term $(\mathbf{1} - \boldsymbol{M})^{\boldsymbol{r}}$ helps narrow the gap between well converged affinity masses and those that are far from convergence. This is the chosen loss function in the paper.

## Other Loss Forms

- L2 form: $L_2(x) = x^2$, where $x = 1 - M \in [0,1]$.
- Smooth L1: $L_{1-smooth}(x) = \begin{cases} x^2 & if \quad |x| < 0.5 \\ |x| - 0.25 & otherwise. \end{cases}$

## Optimization and Convergence

- The total loss when training a neural network with our method is

$$\boldsymbol{L = L_{main} + \lambda L_G}$$

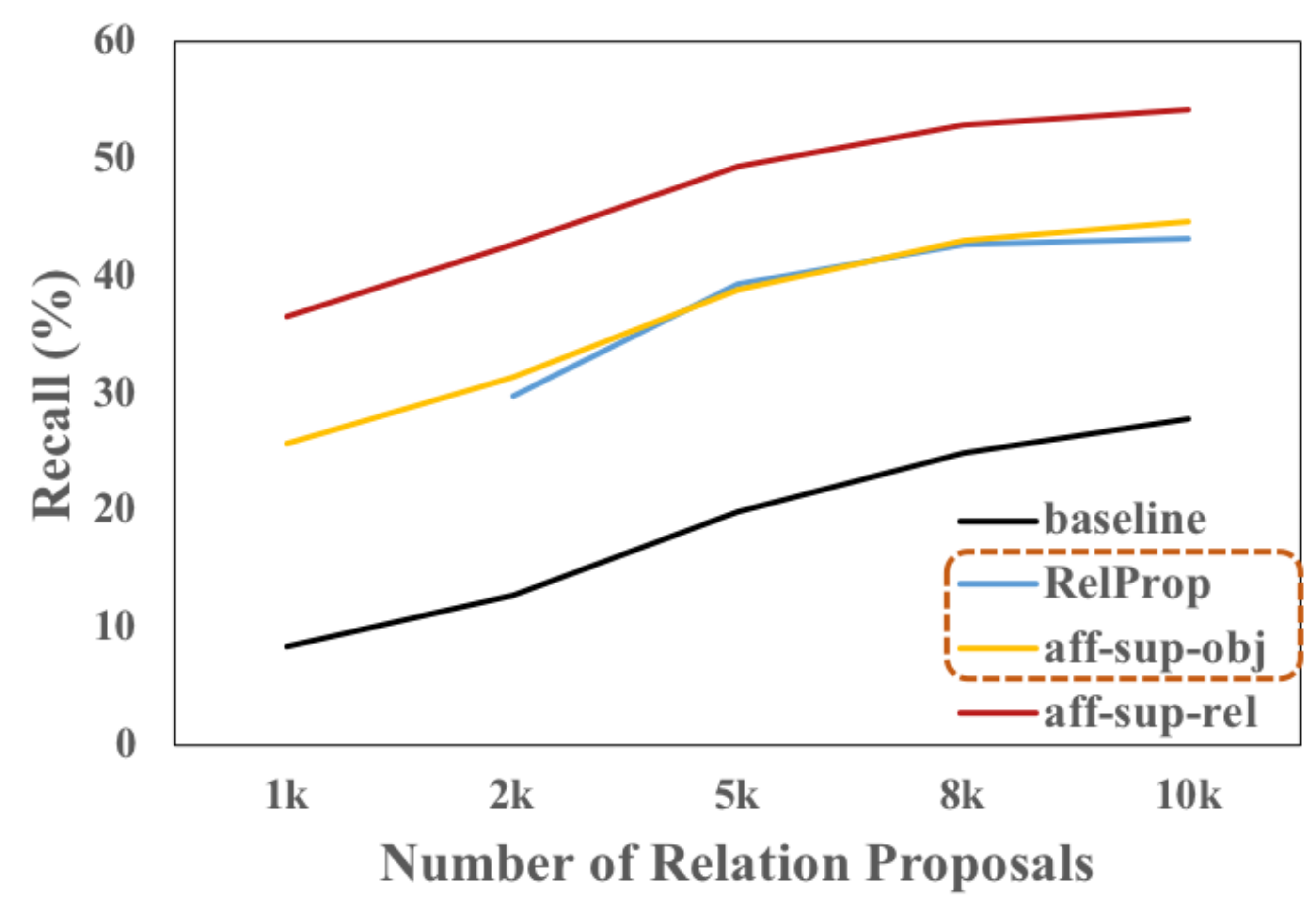
where $L_{main}$ is the main objective loss, which can be detection loss or classification loss.
- $\lambda$ controls the balance between affinity loss and the main objective loss.

# Affinity Mass Loss Ablation Study

| VOC07 | Smooth L1 | L2 | $r = 0$ | $r = 2$ | $r = 5$ |
|---|---|---|---|---|---|
| mAP@all(%) | 48.0 ± 0.1 | 47.7 ± 0.2 | 47.9 ± 0.2 | 48.2 ± 0.1 | **48.6 ± 0.1** |
| mAP@0.5(%) | 79.6 ± 0.2 | 79.7 ± 0.2 | 79.4 ± 0.1 | 79.9 ± 0.2 | **80.0 ± 0.2** |
| recall@5k(%) | 60.3 ± 0.3 | 64.6 ± 0.5 | 62.1 ± 0.3 | **69.9 ± 0.3** | 66.8 ± 0.2 |

Table 1. An ablation study on loss functions using the VOC07 database, with evaluation metrics being detection mAP and relationship recall. The results are reported as percentages (%) averaged over 3 runs. The ground truth relation labels are constructed following the different category connections as described in Slide 6, with only object class labels used.

# Visual Relationship Learning Results




**Black:** Relation Networks [3]
**Blue:** Relation Proposal Nets [4]
**Obj:** Ours + Object Class Label
**Rel:** Ours + Relation Ground Truth

Figure 3. Visual Genome relationship proposal generation. We match the state of the art [4] **with no ground truth relation labels used**. We outperform the state of the art by a large margin (25%) when ground truth relations are used.

# Scene Categorization Results

**Scene Architecture**: visual attention network (Slide 7, Figure 1, part A) with scene task branch (Slide 7, Figure 1, part C). <u>Part A's parameters are fixed in training.</u>

| **Methods** | **CNN** | **CNN** | **CNN + ROIs** | **CNN + Attn** | **CNN + Affinity** |
|---|---|---|---|---|---|
| Pretraining | Imagenet | Imagenet + COCO | Imagenet + COCO | Imagenet + COCO | Imagenet + COCO |
| Features | $F_S$ | $F_S$ | $F_S$, $\max(F_{in})$ | $F_S$, $F_C$ | $F_S$, $F_C$ |
| Accuracy(%) | 75.1 | 76.8 | $78.0 \pm 0.3$ | **$77.1 \pm 0.2$** | **$80.2 \pm 0.3$** |

Table 2. MIT67 scene categorization results, averaged over 3 runs. A visual attention network with affinity supervision gives the best result (the entry in blue), with an evident improvement over a non-affinity supervised version (the entry in green).

# Mini Batch Training Ablation Study

Ablation study on mini-batch training, with the evaluation metric on a test set over epochs (horizontal axis). The best results are highlighted with a red dashed box.

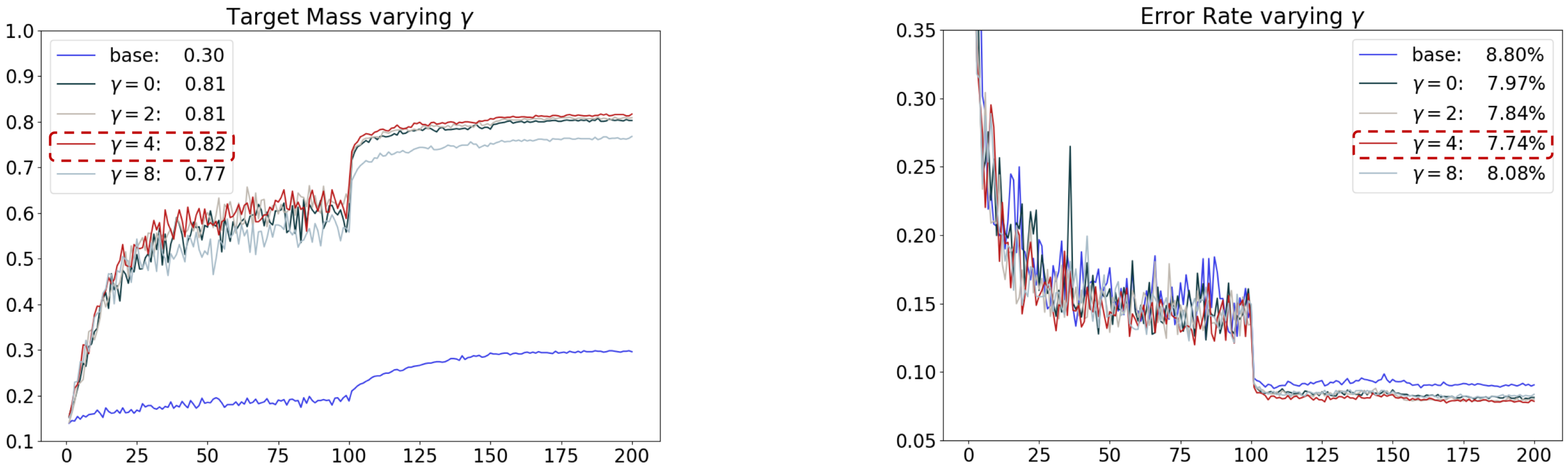



Figure 4. Classification error rates and target mass with varying focal loss' γ parameter.

# Mini Batch Training Ablation Study

Ablation study on mini-batch training, with the evaluation metric on a test set over epochs (horizontal axis). The best results are highlighted with a red dashed box.

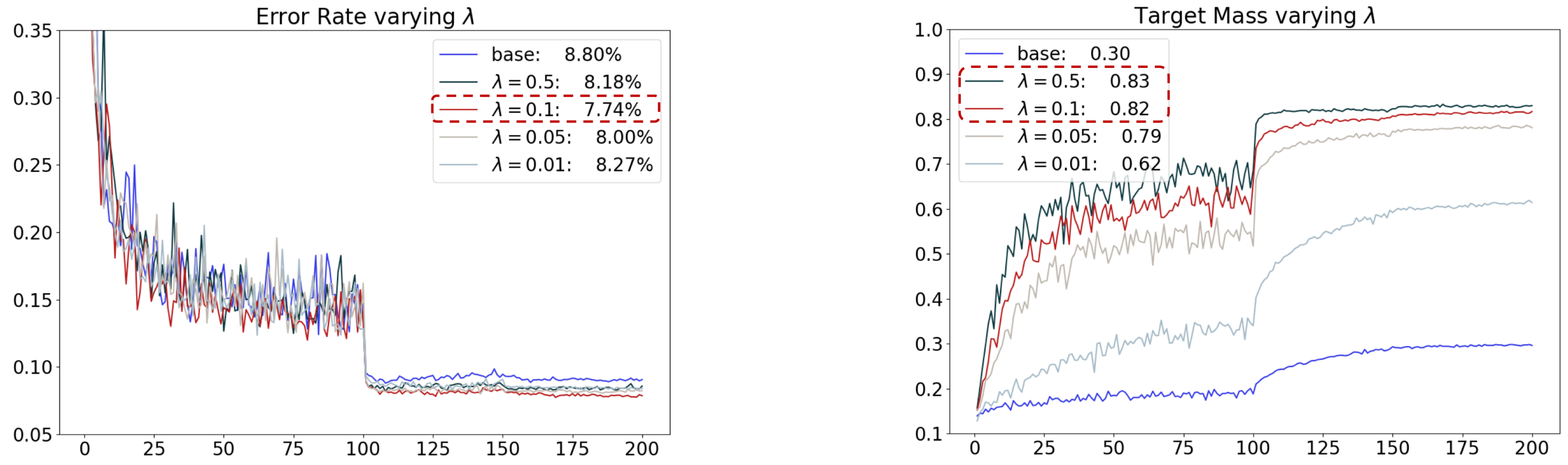



Figure 5. Classification error rates and target mass with varying loss balancing factor λ.

# Mini Batch Training Results

| CIFAR10 | ResNet 20 | ResNet 56 | ResNet 110 |
|---|---|---|---|
| base CNN | $91.34 \pm 0.27$ | $92.24 \pm 0.48$ | $92.64 \pm 0.59$ |
| Affinity Sup | $92.03 \pm 0.21$ | $92.90 \pm 0.35$ | $93.42 \pm 0.38$ |
| **CIFAR100** | **ResNet 20** | **ResNet 56** | **ResNet 110** |
| base CNN | $66.51 \pm 0.46$ | $68.36 \pm 0.68$ | $69.12 \pm 0.63$ |
| Affinity Sup | $67.27 \pm 0.31$ | $\mathbf{69.79 \pm 0.59}$ | $\mathbf{70.5 \pm 0.60}$ |
| **Tiny Imagenet** | **ResNet 18** | **ResNet 50** | **ResNet 101** |
| base CNN | $48.35 \pm 0.27$ | $49.86 \pm 0.80$ | $50.72 \pm 0.82$ |
| Affinity Sup | $\mathbf{49.30 \pm 0.21}$ | $\mathbf{51.04 \pm 0.68}$ | $\mathbf{51.82 \pm 0.71}$ |

Table 3. Affinity supervision results in mini-batch training. CIFAR results are reported over 10 runs and tiny ImageNet over 5 runs