Improvements in Vision Graph Neural Networks

Bartłomiej Wójcik[0009-0003-5096-0755], Arkadiusz Tomczyk[0000-0001-9840-6209]

Institute of Information Technology, Lodz University of Technology, al. Politechniki 8, 93-590 Lodz, Poland

Abstract

Vision Graph Neural Networks (ViG) have demonstrated superior performance in computer vision tasks compared to Vision Transformers (ViTs) and Convolutional Neural Networks (CNNs). ViG's adaptability to varying spatial relationships and irregular structures within images, coupled with its dynamic information aggregation, positions it as a robust solution for understanding of both fine-grained details and broader scene context. However, challenges such as vanishing gradient during training and the methods of defining edges need attention. In this work, we propose improvements to ViG, focusing on mitigating vanishing gradient issues, introducing novel edge generation strategies, and incorporating trainable edge weights.

Intorduction

Graph Neural Networks (GNNs) have emerged as a versatile and powerful tool across various domains, offering unique capabilities in modeling complex relational data. Originally developed for tasks involving graph-structured data such as social networks, molecular structures, and recommendation systems, GNNs have shown remarkable adaptability and efficacy. Their ability to capture intricate dependencies among entities represented as nodes and edges has led to significant advancements in diverse applications.

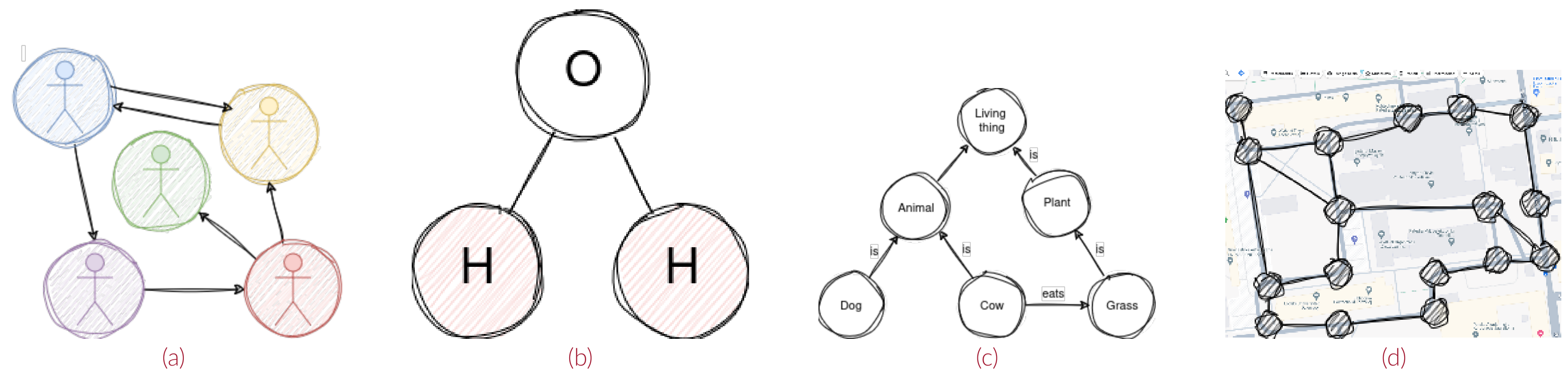

Figure 1. Applications of GNNs: (a) - social networks, (b) - molecular structures, (c) - knowledge graphs, (d) - road networks.

As GNNs gained popularity, researchers explored their potential beyond traditional graph data, extending their applicability to domains such as natural language processing, knowledge graphs, and even computer vision. While initially designed for non-Euclidean data, the inherent flexibility of GNNs enables them to handle structured data in various forms, including images.

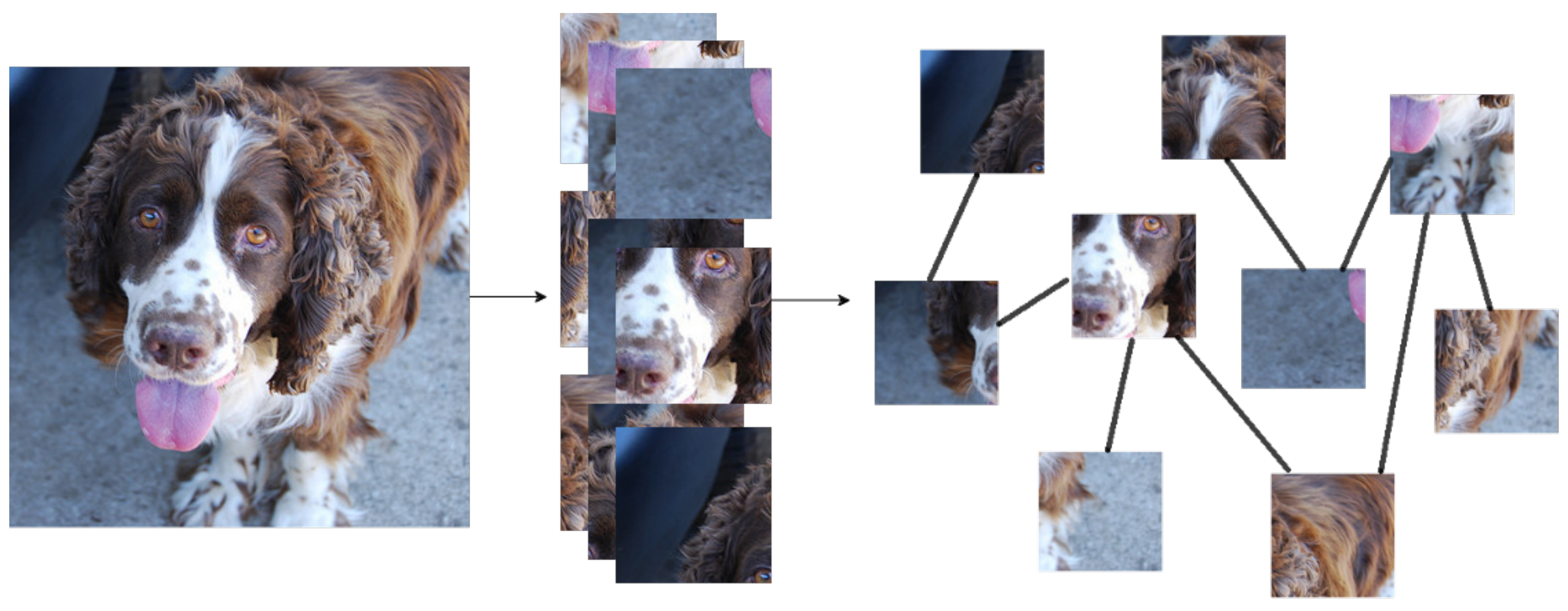

Figure 2. Image to graph conversion

Vision Graph Neural Networks[1] (ViG) emerge as powerful contenders for computer vision tasks, surpassing Vision Transformers[2] (ViTs) and Convolutional Neural Networks (CNNs) in flexible processing and seamless aggregation of global context. Graph Neural Networks (GNNs), designed to operate on graph-structured data, exhibit a remarkable ability to adapt to varying spatial relationships and irregular structures within images. Their flexibility enables the dynamic aggregation of information across nodes, facilitating effective propagation of context throughout the graph. Unlike the fixed receptive fields of CNNs, GNNs naturally handle complex structures of images. Comapring to ViT, they need not to considered a fully conntected graph and offer a variety of different graph convolutional operators.

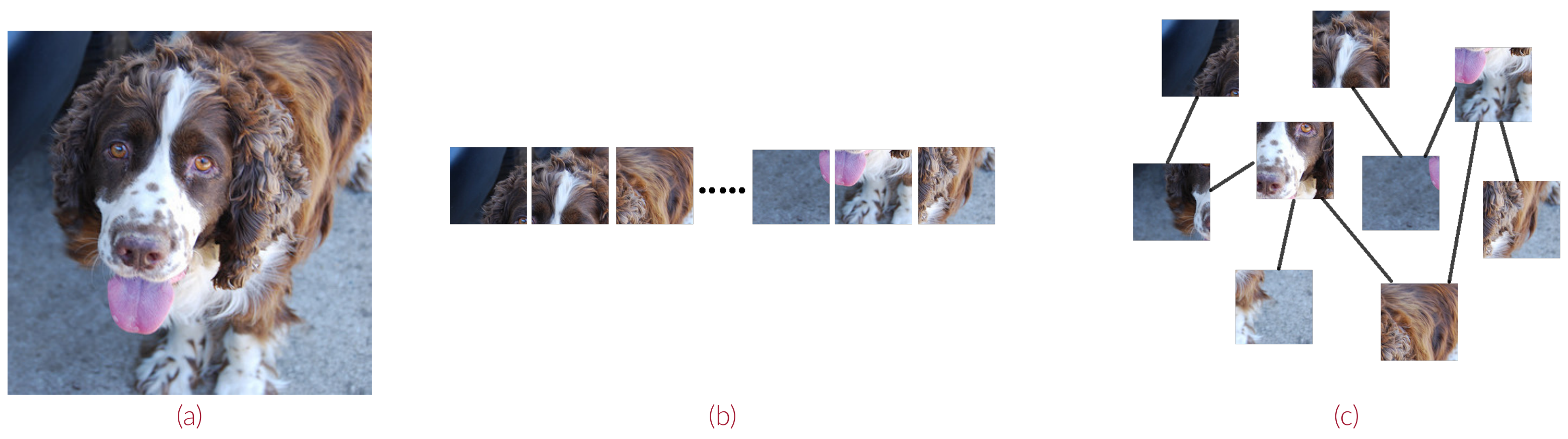

Figure 3. Comparison of input image representation: (a) - when using CNNs, we are constrained by the grid of pixels in the image, (b) - with the ViT architecture all patches are arranged in a sequence, which is further processed by transformer encoder layers, (c) - ViG allows for arbitrary dependencies to be set between patches.

Convolution operations in CNNs extract local features from grid-like data, such as images, by applying filters across the input. Attention mechanisms in Transformers capture global dependencies in sequential data by allowing elements to focus on relevant information. Message passing in GNNs facilitates information exchange between nodes in graph structures, enabling the modeling of complex relational data.

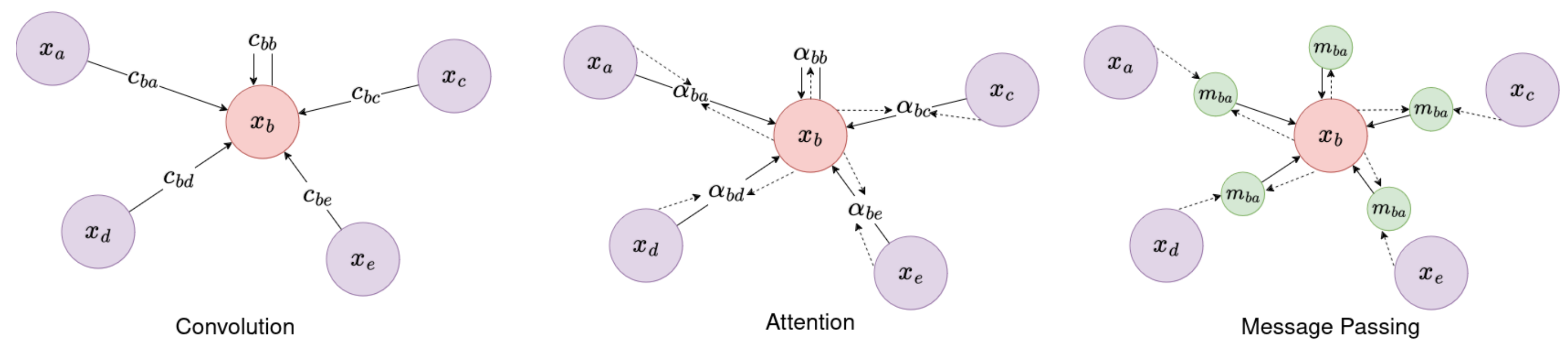

Figure 4. Dataflows in CNN, ViT and ViG models.

Improvments

We demonstrate our improvements in effectiveness of ViG model on image classification task. To ensure the comparability of the experiments, in all of them we use similar architecture (Figures 6 and 7): the same CNN block to convert the image into patches, the same number of graph convolutions, global pooling and linear classifier. The proposed novelties in ViG's architecture include: **alternative static edge creation strategies, residual connections and trainable edge weights**. We demonstrate our improvements in effectiveness of ViG model on image classification task on the Imagenette[a] dataset, featuring a subset of 10 easily distinguishable classes from ImageNet, containing: tenches, English springers, cassette players, chain saws, churches, French horns, garbage trucks, gas pumps, golf balls, and parachutes.

Method

In contrast to the computationally demanding approach of generating edges based on the K nearest neighbors[1], we propose alternative strategies involving neighbor and complete versions, which provide compelling advantages in the context of graph construction for vision tasks (Figure 5).

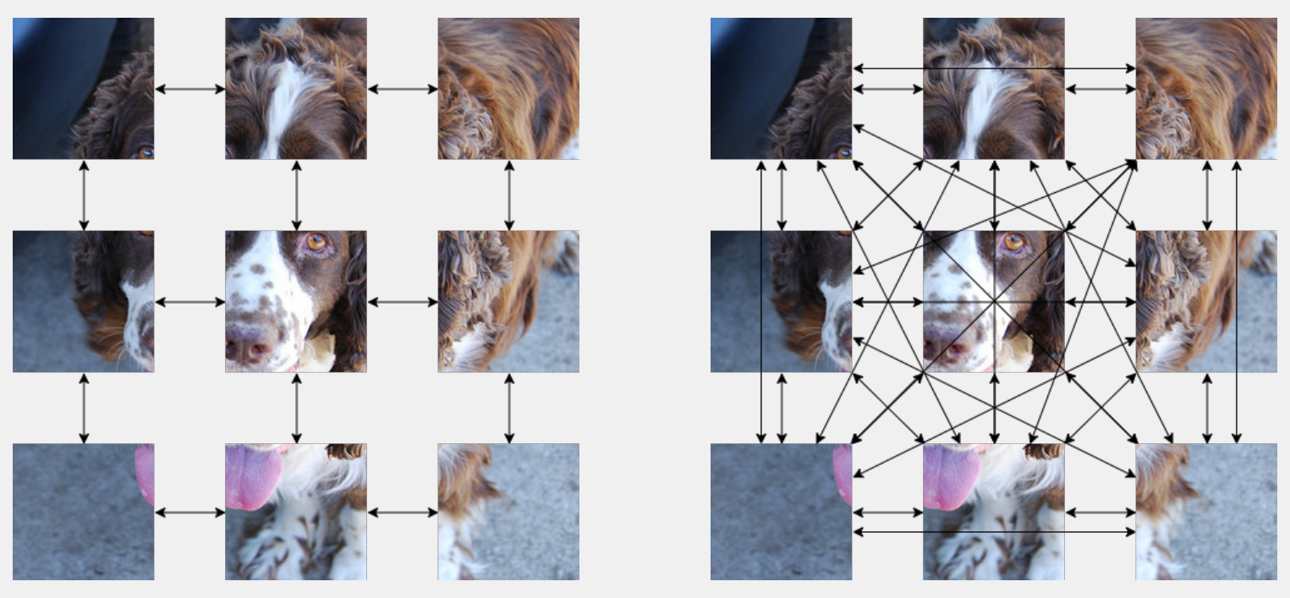

Figure 5. Static edge creation: (a) - neighbour edges, (b) - complete edges.

Rather than fixing on a specific number of neighbors basing on proximity, the generation of neighbor edges offers a more stable solution. Nodes establish connections based on their inherent spatial relationships. Moreover, although it is more expensive computationally, the incorporation of complete edges augments the graph with a global perspective allowing nodes to be linked.

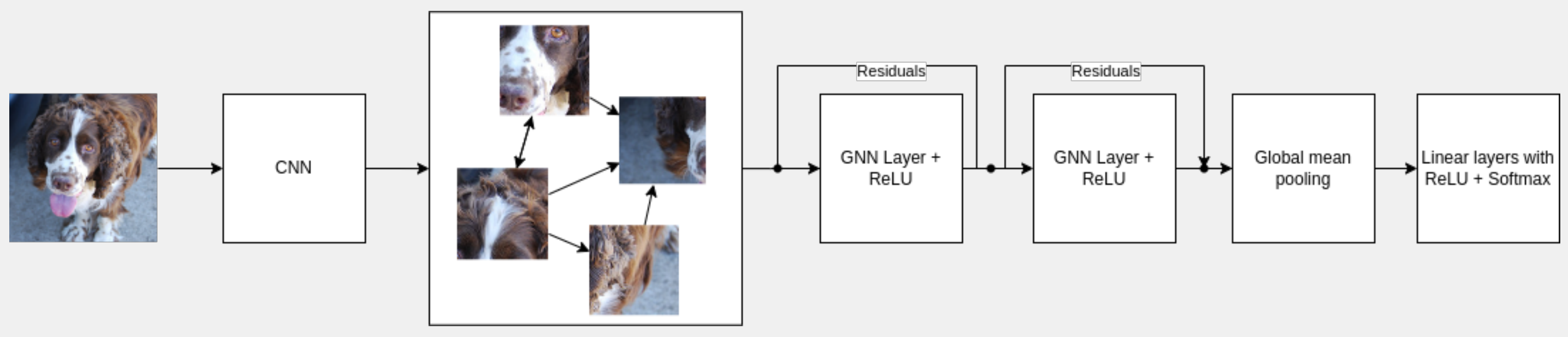

Figure 6. ViG architecture with residual connections.

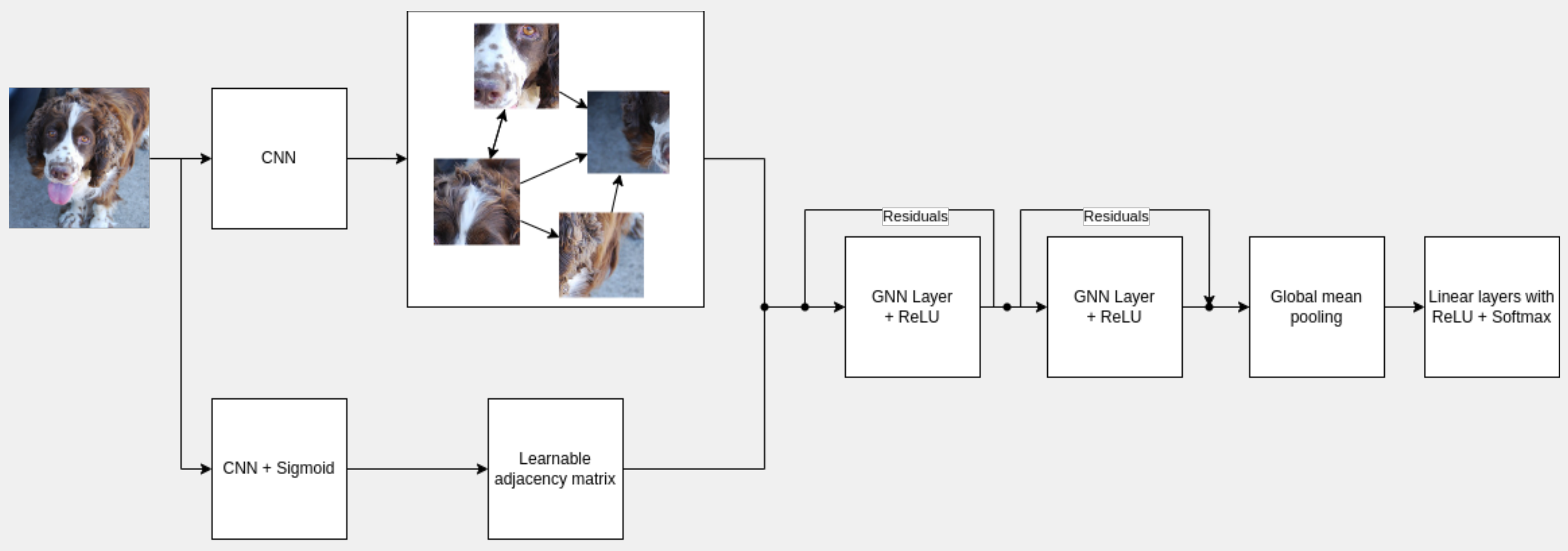

Figure 7. ViG architecture with trainable edge weights.

One of the methods of solving vanishing gradient problem in classical CNNs was the usage of residual connections. It enabled also the creation of deeper architectures. Inspired by this success residual connections were added also in GNNs ([3]).

Results

All our experiments were conducted using three different graph convolutional layers: GraphSAGE (SAmple and aggreGatE) ([4]), Graph Attention Networks (GAT) ([5]) and Graph Transformer ([6]), each model consisted of two GNN layers. ViG and ViT models were reproduced as in the original publications. CNN model consisted of two classic convolutional layers instead of GNN layers. All results are the average of the three trials. Moreover, each group of trials was initialized with the same set of seeds. Every experiment was trained for a maximum of 100 epochs with early stopping on validation accuracy with patience of 20 epochs. Then, the epoch with the best validation accuracy was used for testing. As Imagenette only contains a train and validation dataset, the train dataset of Imagenette was split with fixed seed into train (8500 samples) and validation (969 samples) datasets, and the original validation dataset was used as the test (3925 samples) dataset.

Table 1. Results of experiments (C - complete edges, N - neighbor edges, R - residuals, TE - trainable edges).

Model	Accuracy	Model	Accuracy
Ours SAGE C-R-TE	0.871 ± 0.003	Ours Trans C	0.848 ± 0.006
Ours Trans C-R-TE	0.867 ± 0.006	Ours GAT N	0.845 ± 0.013
Ours GAT N-R	0.866 ± 0.008	Ours GAT C	0.845 ± 0.009
Ours GAT C-R	0.865 ± 0.009	ViG	0.840 ± 0.004
Ours GAT C-R-TE	0.863 ± 0.005	CNN	0.832 ± 0.021
Ours SAGE C-R	0.860 ± 0.004	Ours Trans N	0.825 ± 0.017
Ours SAGE N-R	0.857 ± 0.012	Ours SAGE N	0.825 ± 0.007
Ours Trans C-R	0.856 ± 0.003	Ours SAGE C	0.822 ± 0.009
Ours Trans N-R	0.852 ± 0.011	ViT	0.746 ± 0.005

Conclusions

The results presented in Table 1 reveal several notable findings. Firstly, our proposed modifications to the ViG model, particularly those incorporating complete trainable edges (C-R-TE), have led to significant improvements in accuracy compared to traditional convolutional neural networks (CNN) and the Vision Transformer (ViT) on the Imagenette dataset. This suggests that leveraging graph-based structures and integrating them into convolutional architectures can effectively enhance performance in image classification tasks. Furthermore, the performance of different graph convolutional layers varied, but all the layers indicated the efficacy of this approach in capturing global graph structures for image feature extraction.

References

[1] Han, K., Wang, Y., Guo, J., Tang, Y., and Wu, E. Vision gnn: An image is worth graph of nodes. *Advances in Neural Information Processing Systems*, 35:8291–8303, 2022.

[2] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. URL https://arxiv.org/abs/2010.11929.

[3] Li, G., Müller, M., Thabet, A. K., and Ghanem, B. Can GCNs go as deep as CNNs? *CoRR*, abs/1904.03751, 2019. URL http://arxiv.org/abs/1904.03751.

[4] Hamilton, W. L., Ying, R., and Leskovec, J. Inductive representation learning on large graphs. *CoRR*, abs/1706.02216, 2017. URL http://arxiv.org/abs/1706.02216.

[5] Brody, S., Alon, U., and Yahav, E. How attentive are graph attention networks? *CoRR*, abs/2105.14491, 2021. URL https://arxiv.org/abs/2105.14491.

[6] Shi, Y., Huang, Z., Wang, W., Zhong, H., Feng, S., and Sun, Y. Masked label prediction: Unified massage passing model for semi-supervised classification. *CoRR*, abs/2009.03509, 2020. URL https://arxiv.org/abs/2009.03509.

[a] https://github.com/fastai/imagenette