Graph Representation

2021 연구실 동계 세미나

김 기 남

Vision & Display Systems Lab.

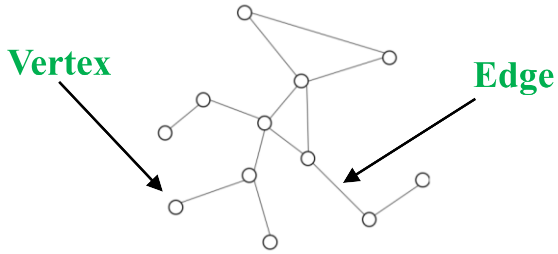
Dept. of Electronic Engineering, Sogang University

Outline

- GNN(Graph Neural Network)
- Graph Representation
- 개인 연구 주제
- References

GNN(Graph Neural Network)

- What is Graph data?
  - Graph data
    - Edge and Vertex

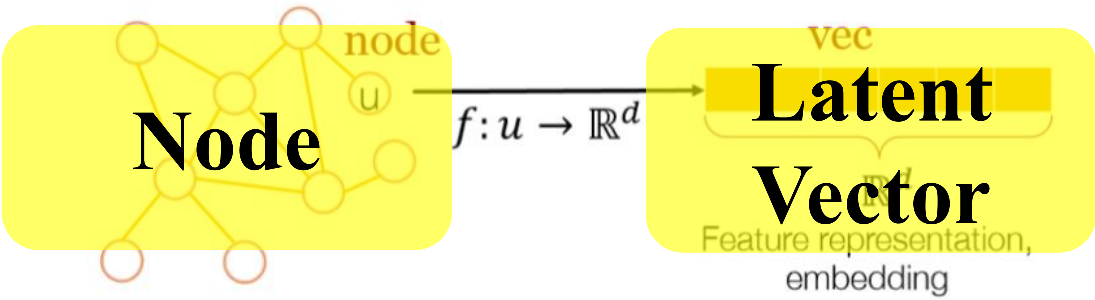


$$G = (V, E)$$

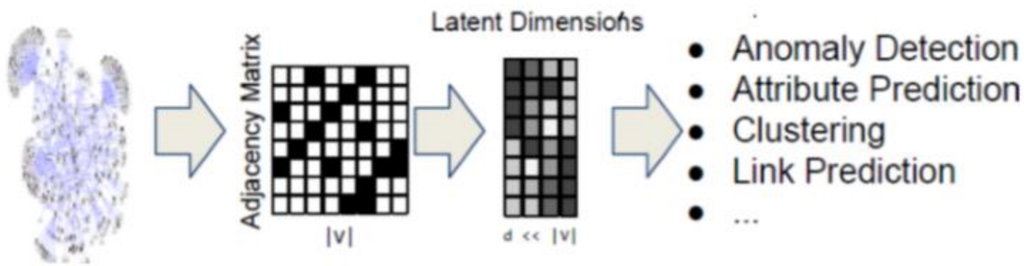
  - GNN's task
    - Node classification
    - Link prediction
    - Clustering
    - Anomaly detection

Graph Representation

- Graph representation?
  - Graph의 각 node를 feature로 표현하는 것
    - Graph를 벡터나 벡터의 집합으로 표현
    - Embedding을 통해 low-dimension space에 각 node를 mapping하여 낮은 차원의 latent matrix 생성



Node Embedding



Network Embedding

Graph Representation

- Why we need graph representation?
  - 기존의 Deep Learning 기법을 직접 적용하는데 한계가 존재
    - CNN, RNN 등을 위해 고안된 기존의 방법들은 대체로 Euclidean 데이터 타입(ex. Grid)
      - Grid의 경우 근접한 픽셀들끼리 연관이 있다는(Locality) 가정 가능
    - Graph 구조는 위상적 구조에 따라 구분되기 때문에 복잡한 데이터 타입
      - 이미지의 경우 회전, 왜곡 등을 통해 변형되지만, Graph의 위상적인 구조는 변하지 않음
    - Embedded Latent vector 를 통해 기존의 deep learning 기법 적용 가능
  - Low-Dimension 으로의 node feature mapping 을 통해 연산량 감소
    - Graph 의 Node 수가 많아지거나, Node feature 의 수가 많을 경우 연산량이 기하급수적으로 증가
    - 임베딩된 vector 끼리의 연산으로 간단하고 빠른 연산 가능

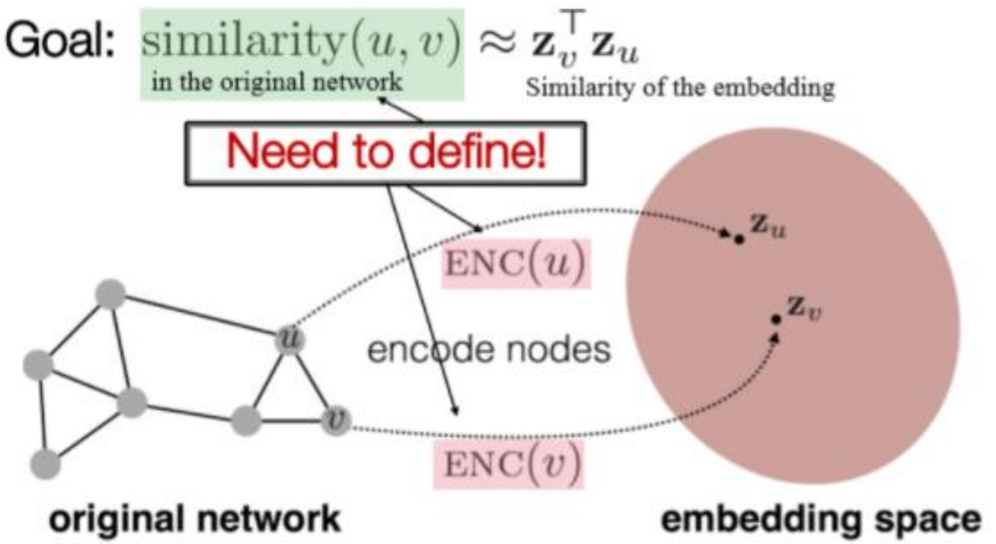
$$Z = f(X, A) = \underbrace{softmax(\hat{A}\,\underbrace{ReLU(\hat{A}XW^{0})}_{\text{Layer 1}}W^{1})}_{\text{Layer 2}}$$

서강대학교 SOGANG UNIVERSITY

VDS LAB

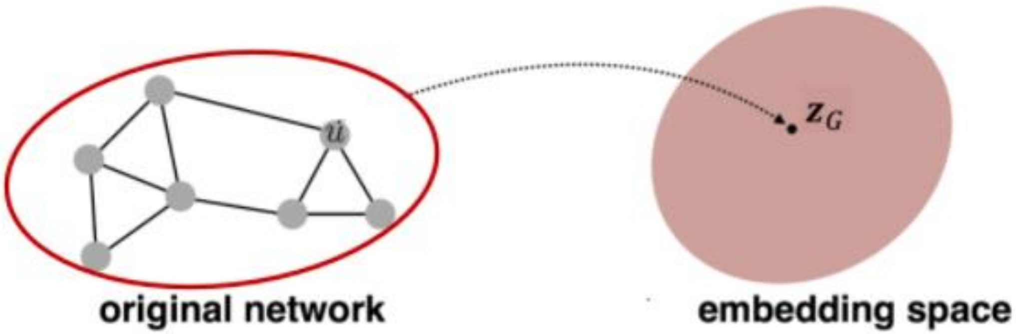
Graph Representation

- Challenge of Graph Representation
  - Embedding 이 Graph 의 성질을 잘 나타내야 함
    - GCN 전체 성능이 embedding 자체의 성능에 큰 영향을 받는 경우가 많음.
    - Graph 의 연결 상태, 주변 구조 표현
    - Original Graph 에서의 similarity 가 Embedding space 상에서도 유사하여야 함.



Graph Representation

- Graph Representation Method
  - Node Embedding
    - Random-walk[1]
    - Node2Vec[2]
    - Variational Graph Autoencoder[3]
    - Graph Projection – Reprojection structure[4]
  - Graph Embedding



Graph Embedding example

Graph Representation

- Node Embedding – Variational Graph Auto-Encoder[3]
  - Variational Auto-Encoder(VAE) 구조를 Graph에 적용
  - 입력으로 Node feature X와 Adjacency Matrix A를 사용하여 Node 의 분포를 학습
    - 2 layers 구조로 첫번째 layer 에서 μ를 추출하고, 두번째 layer에서 $log\sigma^2$을 추출하여 latent vector Z 생성
    - 생성된 Z를 inner product 하여 Adjacency Matrix 로 decoding

$$\mu = GCN_{\mu}(X, A) = \tilde{A}\bar{X}W_1$$

$$log\sigma^2 = GCN_{\sigma}(X, A) = \tilde{A}\bar{X}W_1$$

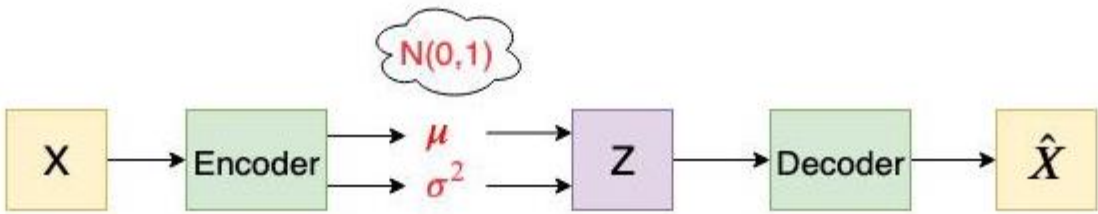
$$Z = \mu + \sigma * \epsilon$$

$$GCN(X, A) = \tilde{A}ReLU(\tilde{A}XW_0)W_1$$

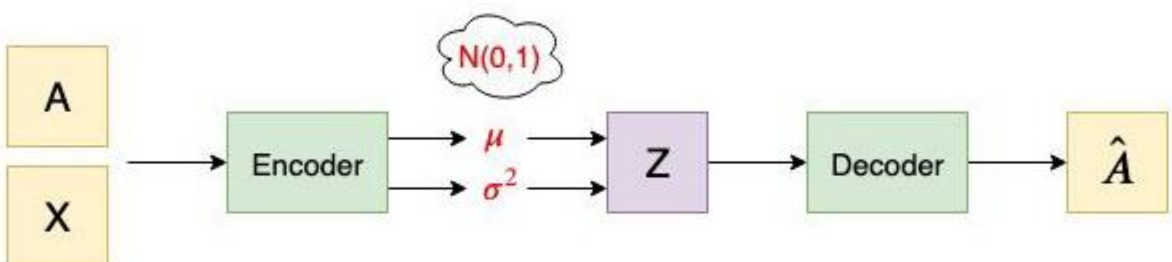
Graph Representation

- Node Embedding –Variational Graph Auto-Encoder
  - Graph Auto-Encoder
    - Variational Graph Auto-Encoder(VGAE) 와는 다르게 1 layer 구조로 latent vector Z를 GCN 을 통해 바로 추출

$$\hat{\mathbf{A}} = \sigma\left(\mathbf{Z}\mathbf{Z}^{\top}\right), \quad \text{with} \quad \mathbf{Z} = \mathrm{GCN}(\mathbf{X}, \mathbf{A})$$



VAE architecture



VGAE architecture

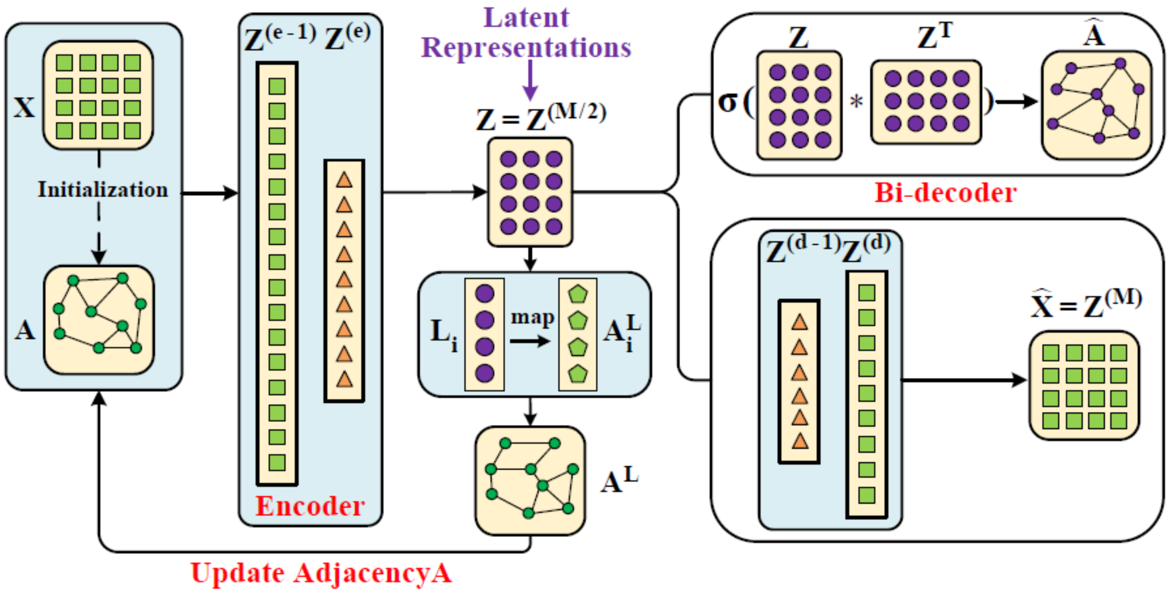
Graph Representation

- Node Embedding –Variational Graph Auto-Encoder
  - Experimental Result
    - Input feature 를 함께 사용하였을 때 성능이 훨씬 높아지는 것을 확인

Method	Cora		Citeseer		Pubmed	
	AUC	AP	AUC	AP	AUC	AP
SC [5]	84.6 ± 0.01	88.5 ± 0.00	80.5 ± 0.01	85.0 ± 0.01	84.2 ± 0.02	87.8 ± 0.01
DW [6]	83.1 ± 0.01	85.0 ± 0.00	80.5 ± 0.02	83.6 ± 0.01	84.4 ± 0.00	84.1 ± 0.00
GAE*	84.3 ± 0.02	88.1 ± 0.01	78.7 ± 0.02	84.1 ± 0.02	82.2 ± 0.01	87.4 ± 0.00
VGAE*	84.0 ± 0.02	87.7 ± 0.01	78.9 ± 0.03	84.1 ± 0.02	82.7 ± 0.01	87.5 ± 0.01
GAE	91.0 ± 0.02	92.0 ± 0.03	89.5 ± 0.04	89.9 ± 0.05	$\mathbf{96.4} \pm 0.00$	$\mathbf{96.5} \pm 0.00$
VGAE	$\mathbf{91.4} \pm 0.01$	$\mathbf{92.6} \pm 0.01$	$\mathbf{90.8} \pm 0.02$	$\mathbf{92.0} \pm 0.02$	94.4 ± 0.02	94.7 ± 0.02

Graph Representation

- Graph Convolutional Auto-encoder with Bi-decoder and Adaptive-sharing Adjacency (BAGE)[5]
  - VGAE 의 Decoder 에서 Adjacency Matrix와 Node feature X까지 decoding 하도록 변형
  - Adaptive-sharing Adjacency Matrix 적용



Graph Representation

- Graph Convolutional Auto-encoder with Bi-decoder and Adaptive-sharing Adjacency (BAGE)[5]
  - Node feature 까지 decoding 하게 학습함으로써 정확한 Latent vector 추정
    - Overfitting 방지
  - Adaptive-sharing Adjacency Matrix
    - Latent vector 로부터 Adjacency matrix 를 추정하여 update 하여 사용
      - Prior Adjacency Matrix 가 정확하지 않은 연결 관계를 나타내는 등의 오염된 부분이 있을 때 이를 optimization 할 수 있음
      - 네트워크가 prior adjacency information 에 대해 높은 의존도를 갖지 않도록 학습 가능
    - Adaptive-sharing Adjacency Matrix 는 Backpropagation 하여 추정하지 않음
      - Backpropagation 을 통한 adjacency matrix 추정은 오히려 meaningless 한 matrix 생성
      - 추정된 각 node latent vector 의 distance 에 Lagrange equation 과 KKT(Karush-Kuhn-Tucket) condition 을 적용하여 adjacency matrix 추정

$$\min_{\mathbf{A}} \sum_{i,j=1}^{n} \left(\|\mathbf{z}_i - \mathbf{z}_j\|_2^2 \, a_{ij} + \gamma_i a_{ij}^2 \right)$$

$$\text{s.t.} \quad \mathbf{a}_i^T \mathbf{1} = 1, 0 \leq \mathbf{a}_i \leq \mathbf{1}.$$

Graph Representation

- Graph Convolutional Auto-encoder with Bi-decoder and Adaptive-sharing Adjacency (BAGE)[5]
  - Experimental Results

Methods		Ours	Ours-NA	CAN	GAE	SAE	RWL-AN	k-means
Cora-P	ACC	**48.49**	42.64	30.13	47.14	\	\	\
	NMI	**32.09**	19.87	10.79	27.24			
Citeseer-P	ACC	**48.21**	34.54	21.17	38.94	\	\	\
	NMI	**21.71**	9.99	10.56	13.11			
COIL	ACC	**68.69**	66.62	68.52	\	66.75	63.66	65.53
	NMI	81.20	79.99	**81.79**		76.24	71.26	79.11
FEI	ACC	42.01	43.66	**45.14**	\	36.45	35.10	38.84
	NMI	73.40	74.02	**74.48**		70.35	69.17	72.28
IMM	ACC	**64.96**	63.47	46.00	\	54.46	49.91	57.71
	NMI	**80.82**	74.27	71.49		75.74	72.89	77.92
YALEB	ACC	**58.31**	51.94	15.31	\	46.94	53.58	51.78
	NMI	**74.99**	69.50	29.03		64.72	66.80	61.38

Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

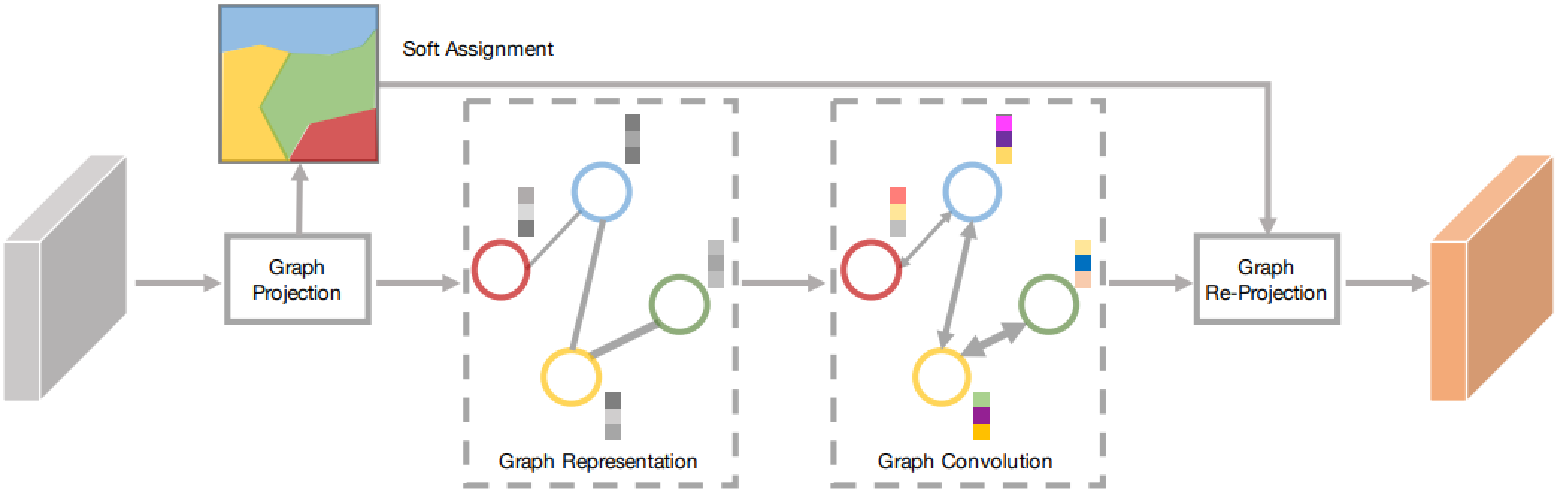
  Beyond Grids: Learning Graph Representations for Visual Recognition[4]

  - 이전의 GCN은 Graph 구조의 data 를 위한 구조
    - ex) Semi-supervised clustering, Unsupervised clustering …
  - 이미지와 같이 Pixel 단위로 이루어져 있는 data 에 대해 GCN 구조 적용이 어려움
  - Pixel로 구성되어 있는 data를 Graph space 로 projection 하고, 이 data 를 GCN을 통한 계산 후 다시 pixel domain 으로 reprojection
  - 기존 네트워크 대비 semantic segmentation, object detection, object instance segmentation 에서 SOTA 달성
    - Local region 의 information 만 활용하였던 기존의 CNN models와 달리 global region 에 대한 information 활용 가능

Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Beyond Grids: Learning Graph Representations for Visual Recognition



Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Beyond Grids: Learning Graph Representations for Visual Recognition

  - Graph Projection
    - Pixel-to-vertex assignment
      - 비슷한 Feature 를 가진 pixel 들을 하나의 vertex 로 assign

$$q_{ij}^{k}=\frac{\exp\left(-\|(x_{ij}-w_k)/\sigma_k\|_2^2/2\right)}{\sum_k \exp\left(-\|(x_{ij}-w_k)/\sigma_k\|_2^2/2\right)} \qquad \sum_k q_{ij}^{k}=1$$

    - Encoding
      - 각 vertex 에서 assign 된 pixel information 을 latent vector 로 encoding

$$z_k=\frac{z'_k}{\|z'_k\|_2}, \quad z'_k=\frac{1}{\sum_{ij} q_{ij}^{k}}\sum_{ij} q_{ij}^{k}\left(x_{ij}-w_k\right)/\sigma_k$$

    - Adjacency Matrix
      - Encoding 된 latent vector Z에서 Adjacency Matrix 추출

$$\mathcal{A}=Z^{T}Z$$

Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Beyond Grids: Learning Graph Representations for Visual Recognition

  - Graph Convolution Network

    - 일반적인 GCN 적용

$$\tilde{Z} = f(\mathcal{A}Z^T W_g)$$

  - Graph Reprojection

    - Vertex-to-pixel

      - Graph projection 시에 구하였던 assignment matrix Q를 이용하여 reprojection

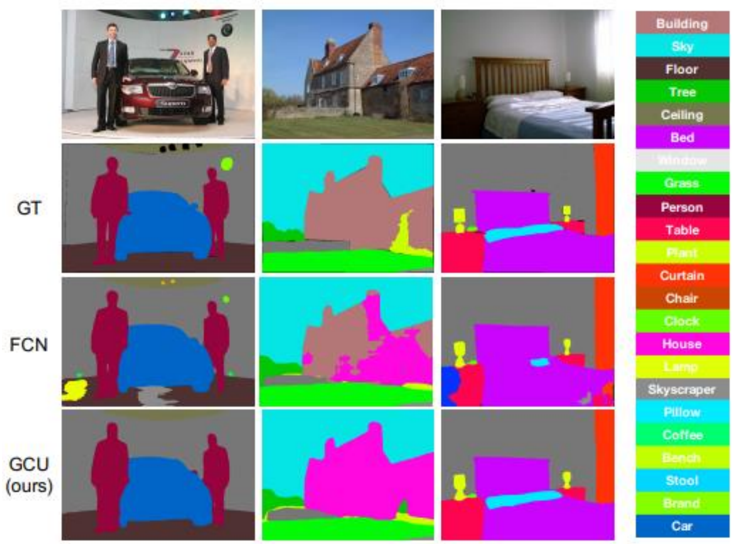
$$\tilde{X} = Q\tilde{Z}^T$$

Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Beyond Grids: Learning Graph Representations for Visual Recognition

  - Experimental Results
    - Semantic segmentation



Backbone	Method	PixAcc%	mIoU%
VGG16 [42]	FCN-8s [12]	71.32	29.39
	SegNet [41]	71.00	21.64
	DilatedNet [17]	73.55	32.31
	CascadeNet [37]	74.52	34.90
Res50 [38]	Dilated FCN	76.51	35.60
	PSPNet [13]	80.76	**42.78**
	EncNet [14]	79.73	41.11
	GCU (ours)	79.51	**42.60**
Res101 [38]	RefineNet [19]	-	40.20
	PSPNet [13]	81.39	43.29
	EncNet [14]	81.69	**44.65**
	GCU (ours)	81.19	**44.81**

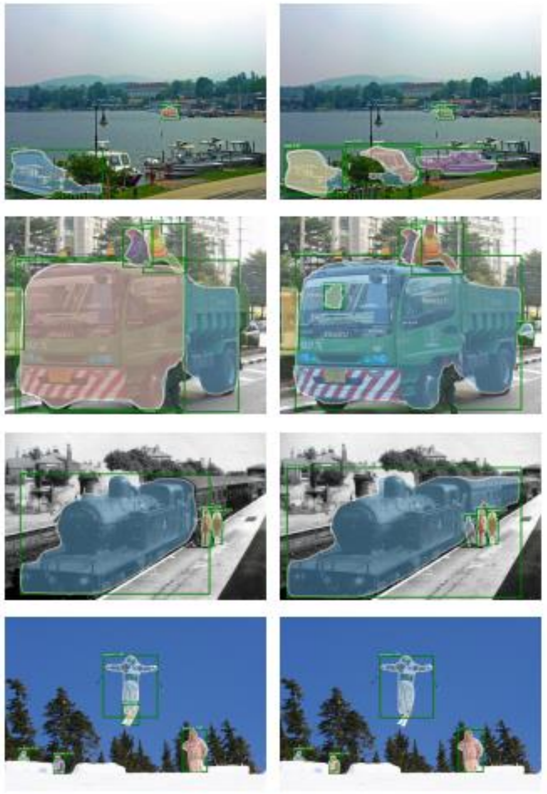
Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Beyond Grids: Learning Graph Representations for Visual Recognition

  - Experimental Results
    - Object instance segmentation

Backbone	Method	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	AP^{seg}	AP^{seg}_{50}	AP^{seg}_{75}
ResNet 50 [38]	Mask RCNN [15, 20]	38.0	59.6	41.0	34.6	56.4	36.5
	Mask RCNN + NL [20]	**39.0**	61.1	41.9	**35.5**	58.0	37.4
	Mask RCNN(Detectron) [15, 44]	37.7	59.2	40.9	33.9	55.8	35.8
	Mask RCNN(Detectron) + GCU	**38.7**	60.5	41.7	34.7	57.2	36.5
ResNet 101 [38]	Mask RCNN [15, 20]	39.5	61.3	42.9	36.0	58.1	38.3
	Mask RCNN + NL [20]	**40.8**	63.1	44.5	**37.1**	59.9	39.2
	Mask RCNN(Detectron) [15, 44]	40.0	61.8	43.7	35.9	58.3	38.0
	Mask RCNN(Detectron) + GCU	**41.1**	63.2	44.9	**36.9**	59.8	39.0

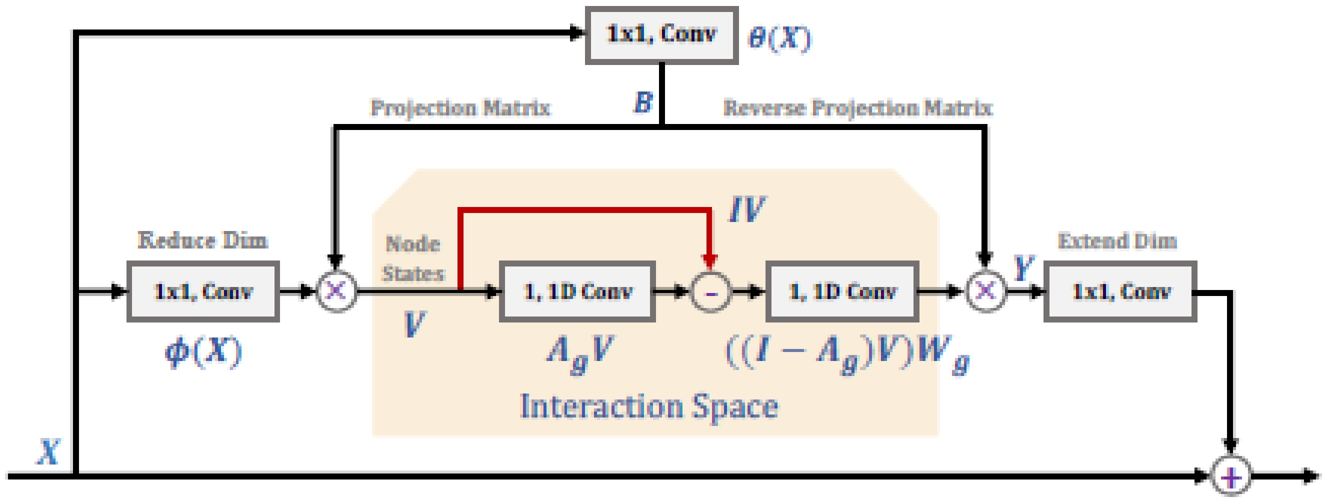


Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Graph-Based Global Reasoning Networks [6]

  - Pixel to Vertex Projection 방법을 CNN을 적용
    - 1x1 Conv 을 통해 projection matrix 추정
      - 추정된 projection matrix 를 이용하여 projection 및 reprojection 진행

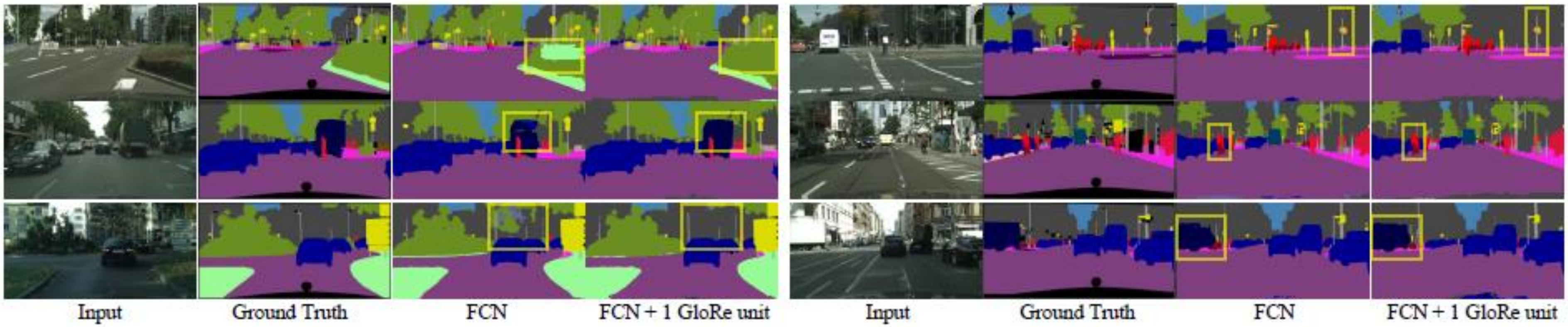


Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Graph-Based Global Reasoning Networks [6]

  - Experimental Results

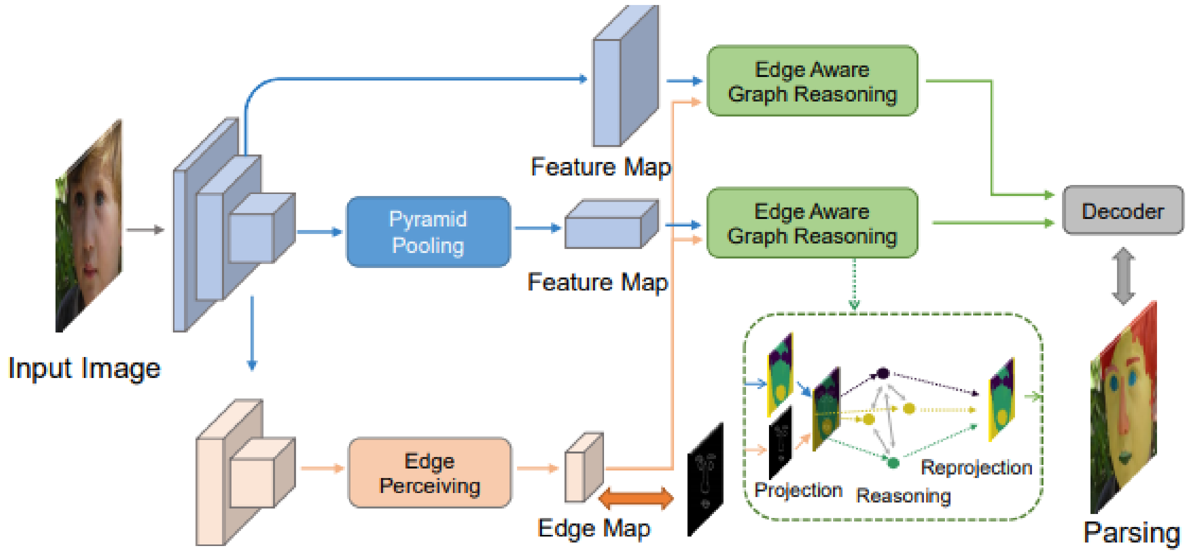


Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Edge-aware Graph Representation Learning and Reasoning for Face Parsing [7]

  - Face Parsing 에 GCN 적용
  - Backbone feature 에서 face parsing feature map 과 edge 를 각자 추출하고, 이를 Attention mechanism 을 적용한 graph projection 알고리즘에 적용하여 성능 향상

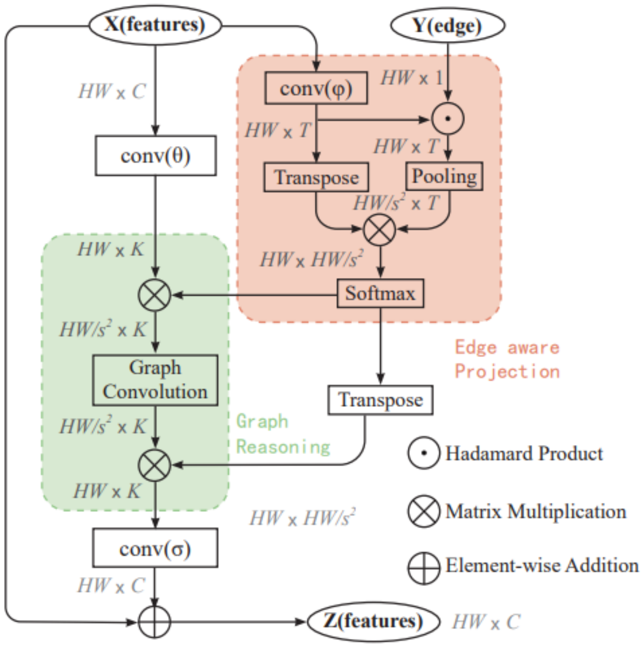


Graph Representation

- Node Embedding – Graph Projection – Reprojection structure

  Edge-aware Graph Representation Learning and Reasoning for Face Parsing [7]

  - Edge-aware Graph Reasoning module
    - Edge pixels 에 더 큰 weight 적용



Graph Representation

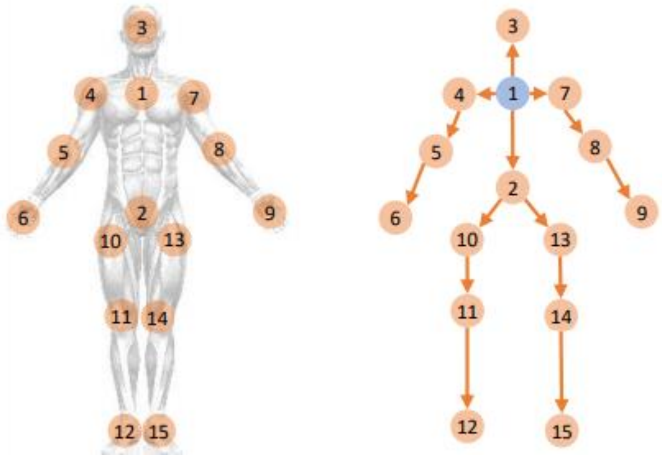
- Node Embedding – Graph Projection – Reprojection structure

  Edge-aware Graph Representation Learning and Reasoning for Face Parsing [7]

  - Experimental Results

개인 연구 주제

- Pose Refinement with GCN
  - Human pose estimation 은 대표적인 graph structure 의 data
    - CNN 기반의 Pose estimation model 에서 추정된 Keypoints 를 Node feature 로 하여 GCN 적용 후 성능 향상을 목표로 함.
      - 인접한 Keypoints 정보를 aggregation 하여 잘못 추정된 Keypoints 위치 수정

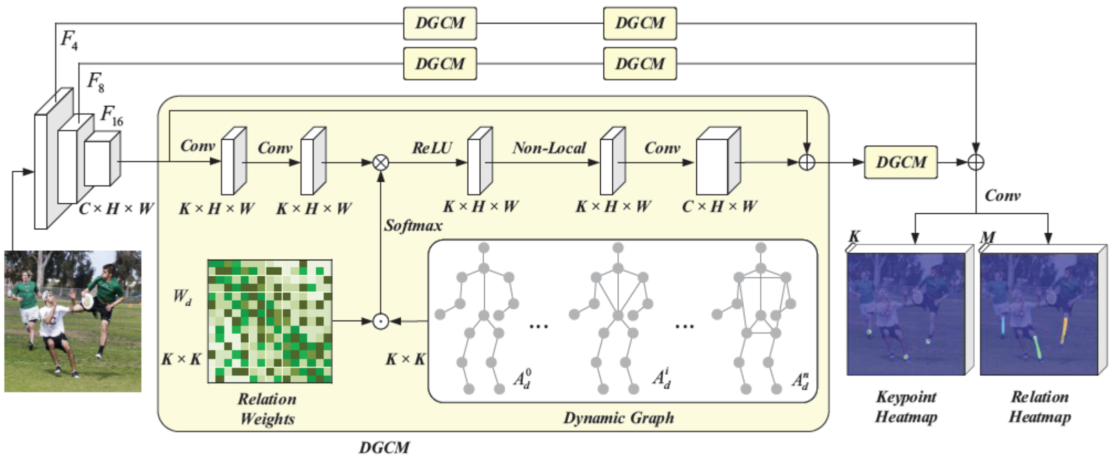


개인 연구 주제

- Pose Refinement with GCN
  - Graph Projection – Reprojection 구조 적용
    - Pixel 단위로 표현되어 있는 human pose heatmap 을 Graph space 로 projection
    - Reprojection 시 Ground Truth heatmap 과의 L2 loss를 통해 [5]와 같이 Node feature reconstruction 까지 학습

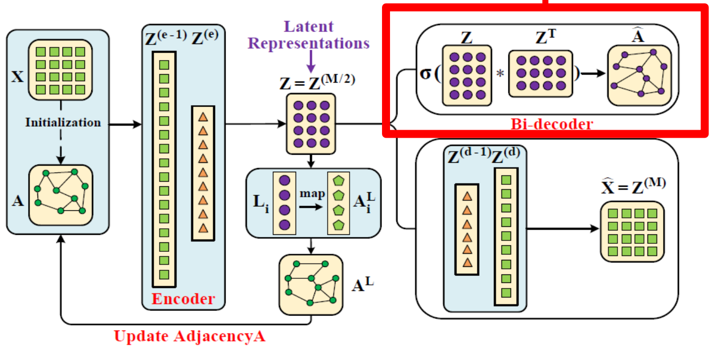
개인 연구 주제

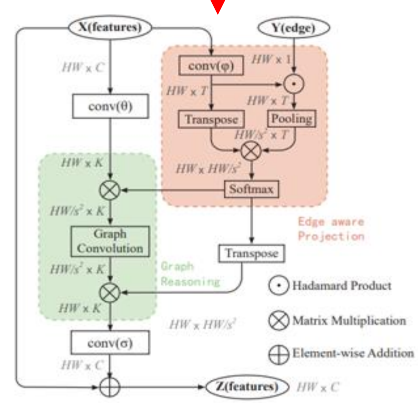
- Pose Refinement with GCN
  - Graph Auto-Encoder 를 통한 Adaptive adjacency matrix 적용
    - Human pose 는 prior adjacency matrix 가 존재하지만 pose estimation 시 prior adjacency matrix 가 아닌 dynamic adjacency matrix 를 적용하였을 때 성능 향상 확인(DGCN[8])
      - Edge 는 실제 물리적인 연결 뿐만 아니라 연결되지 않은 Edge 를 통해서도 Keypoints 에 영향을 줄 수 있음.
      - 실제 inference 시 image 내에 존재하지 않는 Keypoints 가 있어 기존의 prior adjacency matrix 는 [5] 에서 말한 Polluted adjacency matrix 라고 할 수 있음.



개인 연구 주제

- Pose Refinement with GCN
  - Adaptive adjacency matrix 적용
    - GAE 를 통한 latent vector 로 adaptive adjacency matrix 를 적용하여 실제 image 내에 존재하는 human object 에 맞는 adjacency matrix 구축
    - Bi-decoding 시 Adjacency matrix 의 GT는 prior adjacency matrix 에서 GT 상에 존재하지 않는 point 에 해당되는 값만 0으로 가정
  - 추정된 Adaptive adjacency matrix 기반 Attention mechanism 적용
    - Node 간의 feature 를 aggregation 하기 적합하게 추정된 adaptive adjacency matrix를 통해 node feature 의 weighted summation 진행





References

[1] Perozzi, B., Al-Rfou, R., & Skiena, S. (2014, August). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 701-710). ACM.

[2] Grover, A., & Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855-864). ACM.

[3] T. N. Kipf and M. Welling, Variational graph auto-encoders, *Proc. NIPS Workshop Bayesian Deep Learn.*, pp. 1-3, 2016.

[4] Y. Li and A. Gupta. Beyond grids: Learning graph representations for visual recognition. In *NeurIPS*, pages 9245–9255, 2018.

[5] Rui Zhang, Yunxing Zhang, Xuelong Li, Graph Convolutional Auto-encoder with Bi-decoder and Adaptive-sharing Adjacency, arXiv, 2020.

[6] Chen, Y., Rohrbach, M., Yan, Z., Yan, S., Feng, J., & Kalantidis, Y., Graph based global reasoning networks. In CVPR, 2019.

[7] G. Te, Y. Liu, W. Hu, H. Shi, and T. Mei, "Edge-aware graph representation learning and reasoning for face parsing," in European Conference on Computer Vision. Springer, Cham, 2020, pp. 258–274.

[8] Qiu, Z.; Qiu, K.; Fu, J.; and Fu, D. 2020. DGCN: Dynamic Graph Convolutional Network for Efficient Multi-Person Pose Estimation. In AAAI, 11924–11931, 2020.

Thank you