Exploring the Significance of Graphs in Image Processing and Computer Vision Applications

Wieke Prummel[1],

Under the direction of: Thierry Bouwmans[1], Anastasia Zakharova[1]

[1] Laboratoire Mathématiques, Image et Applications (MIA), La Rochelle Université, France

CANUM 2024

28 mai 2024 - Ile de Ré

Table of Contents

Context

Definitions
- Graph Definition
- From Video Data to Graphs
- Graph Convolution Network (GCN)

GraphIMOS
- Idea
- Overview
- Qualitative and Quantitative Results

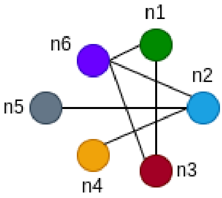
Conclusion and Perspectives

- A. Sandryhaila et al., “Discrete signal processing on graphs,” IEEE Trans. Signal Process., 2013.
- Shuman et al., “The emerging field of signal processing on graphs : Extending high-dimensional data analysis to networks and other irregular domains,” IEEE Signal Process., 2013.
- In "Graph Signal Processing : Overview, Challenges, and Applications" (*A. Ortega et al. Proceedings of the IEEE, 2017*) provide an inventory of the theory and different applications of graph signal processing.
- GraphMOD-Net(*J. Giraldo, T.Bouwmans, et al.), and GraphMOS (J. Giraldo, T.Bouwmans, et al., IEEE TPAMI, 2022), Frontiers of Computer Vision, 2021*) use semi-supervised learning posed as a graph signal reconstruction problem.

Graph

- A Graph, G(V,E), is defined through a set of nodes V $= \{1, \ldots, N\}$ and a set of edges E $= \{(i, j)\}$.
- $A \in \mathbb{R}^{NxN}$ is the adjacency matrix of G such that $A(i,j) = a_{i,j} \in \mathbb{R}^{+}$ is the weight connecting vertices i and j.
- A is symmetric for undirected graphs.
- Graph signal : y : V $\rightarrow \mathbb{R}$

Graph:



A:

	n1	n2	n3	n4	n5	n6
n1	0	0	1	0	0	1
n2	0	0	0	1	1	1
n3	1	0	0	0	0	1
n4	0	1	0	0	0	0
n5	0	1	0	0	0	0
n6	1	1	1	0	0	0

- Various systems consisting of entities and relationships between them can be represented as a graph.
- For example, videos are typically encoded as a succession of sequences of fixed-size 2-dimensional grids of pixels. On the other hand, representing data in a graph-structured way reveals valuable information that emerges from a higher-dimensional representation of these entities and their relationships, and would otherwise be lost.

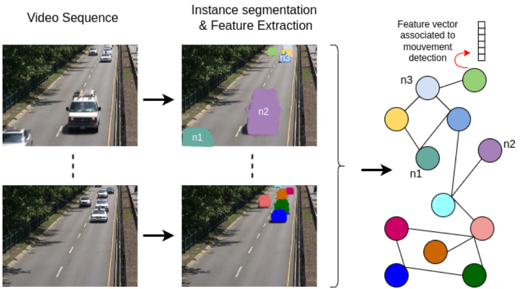


Figure – CDNet2014 image as fixed-sized 2-dimensional grids of pixels

The input of a CNN is a grid, thus the image is a function defined on a grid, such that

$$y_i = w_1 x_{i,1} + \ldots + w_j x_{i,j} \tag{1}$$

- Grid Operations (CNNs)
  - Aggregate pixel values and neighbors.
  - Constant neighbors and fixed order.
  - Same weights applied consistently.

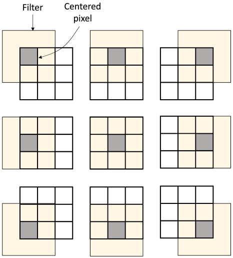


Figure – Illustration of filter passing when applying CNNs on images.

- Graph construction : identify pixel regions, associate a region to a node, feature extraction, calculate euclidean distance between the nodes, k-NN (Nearest Neighbors) to define the connections between two nodes.
- Graph Operations
  - Neighbors vary in number.
  - Arbitrary node ordering. Varying node degrees.
  - GNNs update node representations by aggregating neighboring node representations.

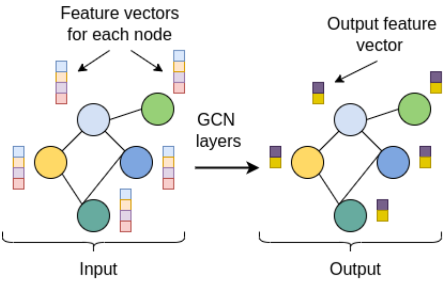


Figure – Figure illustrating node prediction with a GCN implementation.

GCN

- The adjacency matrix $A \in \mathbb{R}^{N \times N}$ is a popular choice of shift operator in GCNs.
- The diagonal degree matrix given by $\boldsymbol{D_{ii}} = \sum_{j=1}^{N} \boldsymbol{A_{ij}}$.
- The whole set of input features can be represented as $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]^{\mathrm{T}} \in \mathbb{R}^{N \times F}$, where F is the number of features.
- We define $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, where $\mathbf{I}$ is the identity matrix.
- The propagation rule of GCN is given by :

$$\mathbf{H}^{(l+1)} = \sigma(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(l)} \mathbf{W}^{(l)}), \tag{2}$$

  where $\tilde{\mathbf{D}}$ is the degree matrix of $\tilde{\mathbf{A}}$, $\mathbf{H}^{(l)}$ is the output matrix of layer l (with $\mathbf{H}^{(0)} = \mathbf{X}$), $\mathbf{W}^{(l)}$ is the matrix of trainable weights in layer l, and $\sigma(\cdot)$ is an activation function such as ReLU or softmax.

Kipf, Welling et al. *Semi-supervised classification with graph convolutional networks*, [2016].

GCN

- The GCN propagation rule :

$$\mathbf{H}^{(l+1)} = \sigma(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}), \tag{3}$$

- The learned weights in a GCN are stored in the matrix $\mathbf{W}$.
- They are used to update the aggregated feature vectors.
- The dimensionality of $\mathbf{W}$ does not depend on the number of nodes in the graph.
- This allows the graph convolutional layer to handle graphs of varying sizes, as long as the feature vectors at each node have consistent dimensions.

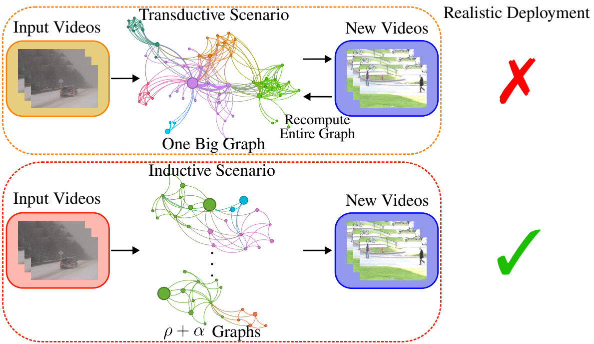


Figure – Transductive vs. Inductive Learning

- ρ represent training and validation graphs and α represent testing graphs.
- Transductive learning assumes the access to the entire training and testing data for evaluation (Giraldo et al. "Graph Moving Object Segmentation", [2022]).
- Inductive learning uses completely unseen data for the testing phase (Krizhevsky et al. "Imagenet classification with deep convolutional neural networks", [2017]).

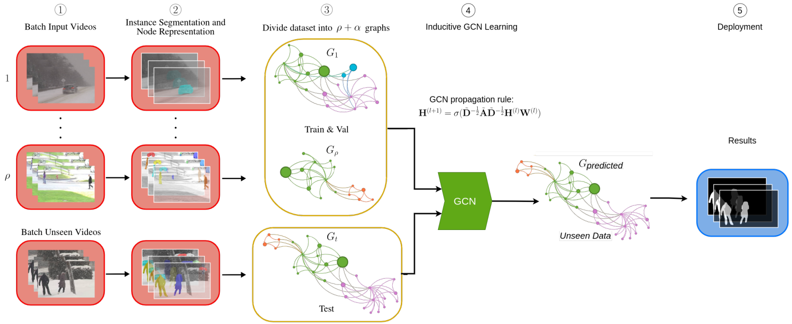
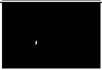
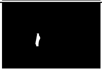
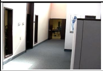
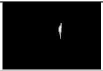
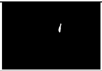


Figure – Pipeline : After instance segmentation and node feature extraction, the dataset is divided into graphs for training-validation and into testing graphs. The algorithm then classifies nodes in the graphs as either moving or static objects.

Instance segmentation algorithm : He et al. *"Mask R-CNN"*, Proceedings of the IEEE international conference on computer vision, [2017].

Inductive Learning Methods

CDNet 2014	Original	Ground Truth	FgSegNet	GraphMOD	GraphIMOS
Pedestrians					
Tramstop					
Cubicle					

Lim et al., "Learning multi-scale features for foreground segmentation", [2020].
Giraldo et al., "Graph CNN for moving object detection in complex environments from unseen videos", [2021].

The evaluation metrics, namely F-Measure, precision, and recall, are specified as follows :

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \ \text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$
$$\text{F-measure} = 2\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{4}$$

TP, FN, and FP denote the true positives, false negatives, and false positives, correspondingly.

Method	BSL	BWT	IOM	LFR	PTZ	THL	CJI	SHW	DBA	Overall
Transductive Learning Methods										
GraphMOS	0.9398	0.8294	0.3607	0.5538	0.7599	0.7292	0.7005	**0.9653**	0.7334	0.7302
GraphMOD-Net (Original)	0.9550	0.8390	**0.5540**	0.5210	**0.7700**	0.6820	0.7200	0.9420	0.8510	0.7593
ConvNet-GT	**0.9813**	**0.9264**	−−	**0.9612**	−−	**0.8543**	**0.9020**	0.9454	**0.8845**	**0.9221**
ConvNet-IUTIS	0.9647	0.8849	−−	0.8273	−−	0.7559	0.8013	0.8590	0.7923	0.8408
Inductive Learning Methods										
FgSegNet	0.5641	0.2789	0.3325	0.2115	0.1400	0.3584	0.2815	0.3809	0.2067	0.3061
GraphMOD-Net (Modified)	0.6474	0.6268	0.5243	0.5337	0.5899	0.5484	0.4926	0.6587	**0.6254**	0.5831
GraphIMOS (Ours)	**0.7003**	**0.6377**	**0.5284**	**0.5478**	**0.5932**	**0.6453**	**0.6700**	**0.6807**	0.5868	**0.6211**

Average F-Measure comparisons on CDNet2014.

M. Braham, M. Van Droogenbroeck, *"Deep Background Subtrac- tion with Scene-Specific Convolutional Neural Networks"*, (IWSSIP), [2016].

Conclusion and Perspectives

- Solving inductive learning problems is more challenging than solving transductive problems.
- We propose a method that combines Graph Signal Processing with GNN techniques in an inductive way.
- GraphIMOS offers improved performance and a better trade-off between performance and practical deployment.
- Perspectives :
  - Optimizing the choice of training, validation and testing graphs by for example studying graph homophily.
  - Using deep features instead of hand crafted features.

Thank You

https ://sites.google.com/view/wiekeprummel/
wieke.prummel@univ-lr.fr