Image Analysis and Object Detection Using Deep Learning

Billal MOKHTARI
(Billal.MOKHTARI@etu.uca.fr)

Master 2 International in Computer Science

Supervisor: Prof. Vincent BARRA
Email: Vincent.BARRA@uca.fr

ISIMA - Clermont Auvergne University
2024

Outline

Context

- The project is founded by ANR (Agence Nationale de la Recherche).
- Detect objects and background contexts in a dataset of 177 images taken from IOM (International Organization for Migration) using panoptic segmentation methods.
- Graph Analysis using statistical methods such as Confidence Intervals.
- Extract important features from the graph.
- Use Graph based Neural Networks to cluster the nodes.
- Image and Text Semantic Analysis based on Siamese Networks.

PointRend

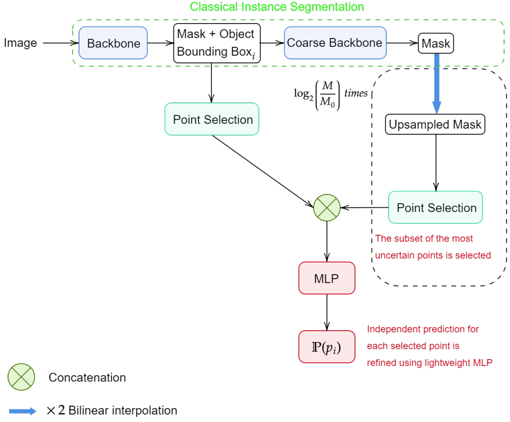


Figure: High Level Architecture of PointRend Approach

Graph Convolutional Networks

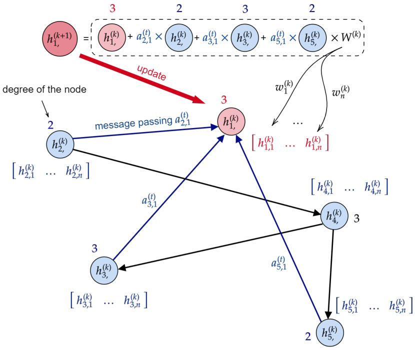


Figure: Graph Convolutional Networks $G = (V, E)$

- $H^{(k+1)} = \sigma\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(k)}W^{(k)}\right)$
- $\widetilde{A} = A + I_m$
- $\widetilde{D_{ii}} = \sum_j \tilde{A}_{ij}$
- $H^{(k)}$: Node features at time k
- A : Adjacency matrix of the Graph G
- $\widetilde{A}$: Adjacency matrix of the Graph G, including the self relationships
- $\widetilde{D}$: Diagonal matrix of degree of each node in G
- σ : Certain activation function
- I_m : Identity matrix of size $m \times m$

Overview

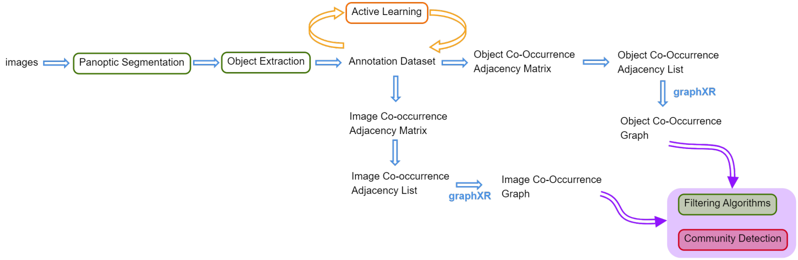


Figure: Overview of our method. The green boxes mean that the tasks are implemented. The orange means that it may be better if it will be implemented. The red one means that it is not yet implemented

Active Learning

No algorithm can be 100% accurate. That says, there exist **false positives (FP)** or **false negatives (FN)**. The solution consists of

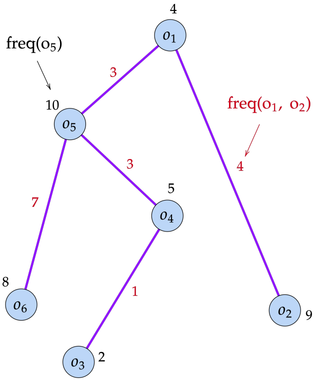
- Active Learning approach.
- The user will have access to the dataset X and associated labels $\widehat{Y}$.
- The user associates new labels Y^* manually to the objects as a ground truth.
- We want that $FN + FP \approx 0$
- **Objective:** $Y^* = \arg\min_{\widehat{Y}} (FN + FP)$

Object Co-Occurrence Graph

Let $G_{obj} = (V_{obj}, E_{obj})$ an **object co-occurrence graph**.

- $V_{obj} = \{o_1, \ldots, o_{|V_{obj}|}\}$ a set of objects o_j.
- Let $\mathcal{T}_{obj} = \{\text{Thing}, \text{Stuff}\}$
- $\deg(o_j) = |\mathcal{N}(o_j)| \in \mathcal{D}_{obj} \subset \mathbb{N}$ the degree of node o_j
- $\text{freq}(o_j) = \sum_{i=1}^{m} \mathbb{1}_{o_j \in x^{(i)}} \in \mathcal{F}_{obj} \subset \mathbb{R}$ the frequency of occurrence of object o_j in the whole dataset $X \in \mathbb{R}^{m \times n_w \times n_h \times n_c}$ where m is the number of examples, $n_w \times n_h \times n_c$ is the shape of each image $x^{(i)}$, $n_c = 3 + |V_{obj}|$ in which 3 refers to the number of channels in RGB and $vertV_{obj}|$ means that for every channel is assigned to a certain object and $\mathbb{1}_{\text{condition}} = \begin{cases} 1 & \text{if} \quad \text{condition} \\ 0 & \text{otherwise} \end{cases}$.
- **Closeness Centrality Measure:** $C(o_j) = \frac{1}{\sum_{i=1}^{|V_{obj}|} d(o_i, o_j)}$ where $d(x, y) \in \mathcal{C}_{obj} \subset]0; 1]$ calculates the length of shortest path between x and y.
- Each object node $o_j \in \mathcal{T}_{obj} \times \mathcal{F}_{obj} \times \mathcal{D}_{obj} \times \mathcal{C}_{obj}$
- $e_{ij}^{obj} = \text{freq}(o_i, o_j) \in E_{obj}$ with $\text{freq}(o_i, o_j) = \sum_{k=1}^{m} \mathbb{1}_{o_i \in x^{(k)} \wedge o_j \in x^{(k)}}$

Object Co-Occurrence Graph



$G_{obj} = (V_{obj}, E_{obj})$

Figure: Illustration of Object Co-Occurrence Graph

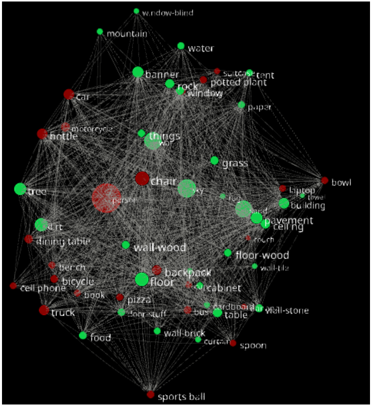


Figure: Object Co-Occurrence Graph using **GraphXR**

Panoptic Segmentation

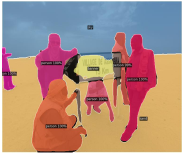


Figure: Image 1

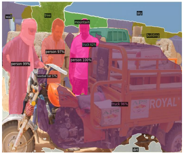


Figure: Image 2

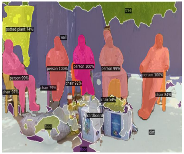


Figure: Image 3

Figure: Image 4

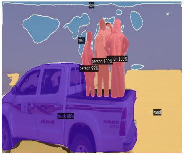


Figure: Image 5

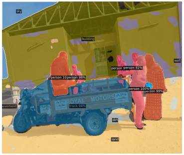


Figure: Image 6

Figure: Image 7

Figure: Image 8

Image Analysis
Frequency

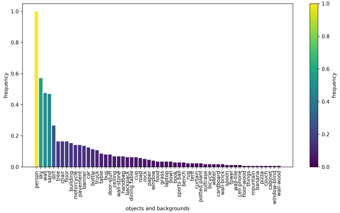


Figure: Object frquency

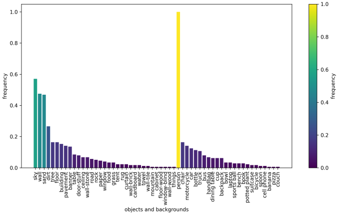


Figure: Object frequency ordered by type (thing or stuff)

	overall	object	background
mean	17.087	16.458	17.545
standard deviation	29.707	35.160	25.600
min	1.000	1.000	1.000
Q1 (25%)	3.000	3.000	3.000
median (50%)	6.000	6.000	7.000
Q3 (75%)	19.000	15.250	24.000
max	177.000	177.000	101.000

Table: Frequency Statistics

Image Analysis
Degree

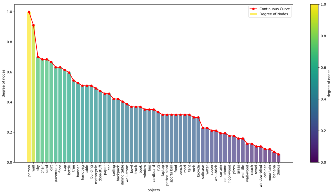


Figure: Node degrees

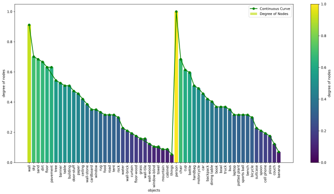


Figure: Node degrees grouped by type

	overall	object	background
mean	21.210	27.603	27.981
standard deviation	11.942	12.974	11.916
min	3.000	4.000	3.000
Q1 (25%)	12.000	18.000	19.000
median (50%)	20.000	24.000	29.000
Q3 (75%)	29.000	34.250	36.000
max	57.000	57.000	52.000

Table: Degree Statistics

Image Analysis
Closeness Centrality

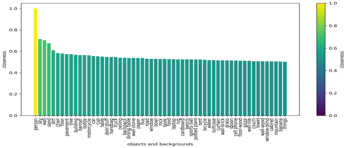


Figure: Closeness Centrality of each node in the graph G_{obj}

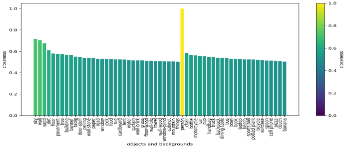


Figure: Closeness Centrality of each node in the graph G_{obj} grouped by type

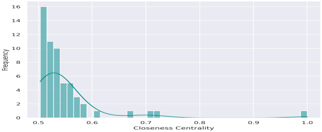


Figure: Closeness Centrality distribution

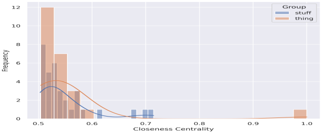


Figure: Closeness Centrality distribution for each type in $\mathcal{T}_{obj}$

Object and Background Importance

Table: Top 10 Objects and Backgrounds According to Different Metrics

Frequency		**Degree**		**Closeness Centrality**	
Object	**Background**	**Object**	**Background**	**Object**	**Background**
person	sky	**person**	wall	**person**	sky
chair	wall	**chair**	sky	**chair**	wall
motorcycle	**sand**	cup	**sand**	bottle	**sand**
car	**dirt**	bottle	**dirt**	motorcycle	**dirt**
bottle	tree	handbag	floor	car	floor
truck	floor	motorcycle	pavement	cup	pavement
bus	building	car	tree	handbag	tree
handbag	pavement	backpack	banner	truck	building
dining table	banner	dining table	table	backpack	banner

Object and Background Importance

- **{person, chair, sand, dirt}** are frequent, with highest degrees and highest closeness centrality.
- **bus** is frequent but does not have the highest degree neither the highest closeness.
- **table** is among a list of 10 objects with highest degree but it is not frequent neither has the highest closeness centrality.

Filtering Algorithm based on Confidence Intervals

Table: Confidence Intervals using $\alpha = 1\%$

Variable	Lower Bound	Upper Bound
Stuff Frequency	5.34	29.75
Thing Frequency	-3.69	36.61
Stuff Degree	34.11	80.02
Thing Degree	26.64	89.61
Stuff Closeness	0.52	0.57
Thing Closeness	0.50	0.61

Image Clustering Based on Node and Edge Message Passing using Graph Neural Networks I

Notation

Table: Notation

Symbol	Description
X	Dataset of images
$G_h^{(k)} = (V_h, E_h)$	Primal graph representation at k-th layer of the GCN
$G_e^{(k)} = (V_e, E_e)$	Dual graph representation at k-th layer of the GCN
m	Number of images in X
n_w	Image width
n_h	Image height
n_c	Number of channels
n_r	Number of arcs in the image representation graph
l_k	Dimension of each arc in the image-based representation graph (l_0 is the number of exist objects)
d_k	Dimension of each node in the image-based representation graph
L	Number of GCN layers
A_e	Relationship graph representation-based adjacency matrix

Image Clustering Based on Node and Edge Message Passing using Graph Neural Networks II

Notation

$A_h^{(k)}$	Adjacency matrix of image representation graph at k-th layer
$A_h^{(k)<j>}$	j-th layer of matrix $A_h^{(k)}$, $1 \leq j \leq l_k$
$\phi_h^{(k)}$	Activation function at the k-th layer of the neural network that will process G_h
$\phi_e^{(k)}$	Activation function at the k-th layer of the neural network that will process G_e
$\psi_j(A)$	Parametric aggregator across the j-th dimension of the matrix A.

$$\psi_j \quad \begin{array}{rcl} A & \longrightarrow & \psi_j(A) \\ \mathbb{E}^{\times_{i=1}^{N} N_i} & \longrightarrow & \mathbb{E}^{\times_{i=1}^{j-1} N_i \times_{i=j+1}^{N} N_i} \end{array}$$

$h_i^{(k)}$	Representation of image i in the k-th layer of the GCN
$f^{(k)}$	Transformation applied on the images to get embeddings at k-th iteration
$e_{ij}^{(k)} \in \mathbb{R}^{l_k}$	Representation of the relationship between images i and j in the k-th layer of the GCN
$a_h^{(k)}{}_{i,j} \in \mathbb{R}^{l_k}$	Coefficient at the i-th row and j-th column of $A_h^{(k)}$

Image Clustering Based on Node-Edge Duality Message Passing using Graph Attention Networks I
Method

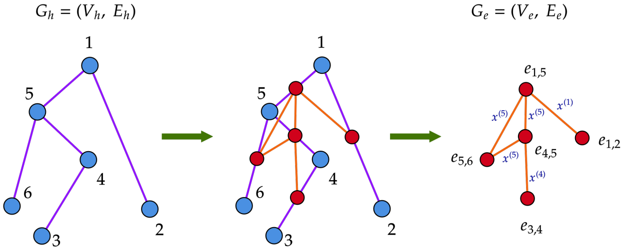


Figure: Primal to Dual Transformation

- **Goal:** find the reduction R such that $G_h = (V_h, E_h) \xrightarrow{R} \quad G_e = (V_e, E_e)$
- $V_e = E_h$ and $(e_{ij}, e_{kl}) = x^{(p)} \in V_e$ if $\{i,j\} \cap \{k,l\} = p \neq \emptyset$

Image Clustering Based on Node-Edge Duality Message Passing using Graph Attention Networks I
Method

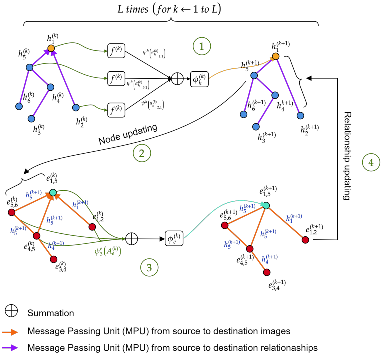


Figure: Method overview

- $H^{(k+1)} = \phi_h^{(k)}\left(\psi_3^h\left(A_h^{(k)}\right) f^{(k)}\left(H^{(k)}\right)\right)$
- $E^{(k+1)} = \phi_e^{(k)}\left(\psi_3^e\left(A_e^{(k+1)}\right) E^{(k)} W^{(k)}\right) \in \mathbb{R}^{n_r \times l_{k+1}}$
- $\psi_3^h(A_h^{(k)}) = W_h * \sum_{j=1}^{3} A_h^{(k)<j>}$
- $\psi_3^e(A_e^{(k)}) = W_e * \sum_{j=1}^{3} A_e^{(k)<j>}$
- $H^{(0)} = X \in \mathbb{R}^{m \times n_w \times n_h \times n_c}$
- $E^{(0)} \in \{0,\ 1\}^{n_r \times l_0}$
- $A_h^{(k)} \in \mathbb{R}^{m \times m \times l_k}, A_e^{(k)} \in \mathbb{R}^{n_r \times n_r \times d_{k+1}}$
- $A_h^{(0)} \in \{0,\ 1\}^{m \times m \times l_0}$
- $W_h \in \mathbb{R}^{m \times m}, W_e \in \mathbb{R}^{n_r \times n_r}$

Image Clustering Based on Node-Edge Duality Message Passing using Graph Attention Networks
Algorithm

Algorithm Image clustering based on message passing between nodes and multi-valued edges using GCNs

Data: $G_h = (V_h, E_h)$
Result: communities $\{c_1, \ldots, c_C\}$
procedure $\textsc{dualGraph}(G_h = (V_h, E_h))$

$V_e \leftarrow E_h$

$(e_{ij}, e_{kl}) \leftarrow x^{(p)}$ if $\{i,j\} \cap \{k,l\} = p \neq \emptyset$ with $p \in \{i,j,k,l\}$

return $G_e = (V_e, E_e)$

end procedure

$G_e = (V_e, E_e) \leftarrow \textit{dualGraph}\,(G_h = (V_h, E_h))$
for $k \leftarrow 0$ *to* L **do**

Compute $H^{(k+1)}$

$E_e \leftarrow H^{(k+1)}$ ▷ Update the dual graph edges with primal graph nodes at step $k+1$

Compute $E^{(k+1)}$

$E_h \longleftarrow E^{(k+1)}$ ▷ Update the primal graph edges with dual graph nodes at step $k+1$

end

$\{c_1, \ldots, c_C\} \leftarrow$ clustering$(V_h^{(L)})$

return $\{c_1, \ldots, c_C\}$

Image Clustering Based on Node-Edge Duality Message Passing using Graph Attention Networks

Clustering

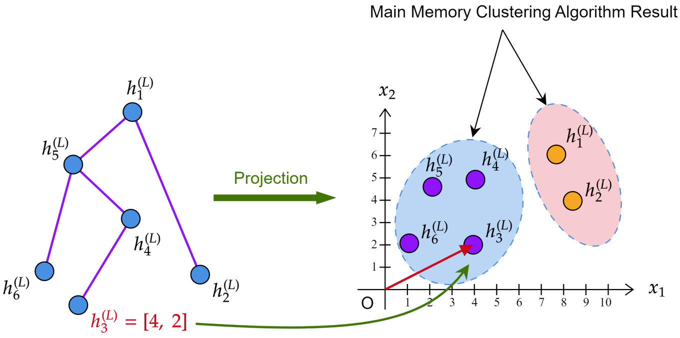


Figure: Projection of the last layer L on a d_L-plan (multi-dimensional plan with d_L dimensions)

Thank you for your attention !