DENSITY-ADAPTIVE GRAPH LAPLACIAN CONSTRUCTION FOR ROBUST SPECTRAL CLUSTERING

GROUP -C11

TEAM MATES :

- JOVIKA N B CB.SC.U4AIE24223
- P POOJITHA CB.SC.U4AIE24237
- POTNURU VARUN CB.SC.U4AIE24244
- RAMU JAHNA BINDU CB.SC.U4AIE24249

INTRODUCTION

- Spectral clustering groups data by analyzing the eigenvalues and eigenvectors of a graph Laplacian.
- Standard spectral clustering uses a fixed global similarity scale, which assumes uniform data density.
- Real-world datasets, such as satellite images, often exhibit heterogeneous densities and complex structures.
- Fixed-scale graph construction can lead to unstable Laplacians and sensitivity to parameter choice.
- This project focuses on constructing a density-adaptive graph Laplacian using local neighborhood information.
- Density adaptation improves the stability of the Laplacian spectrum and the robustness of spectral clustering.
- Experiments on satellite image data demonstrate the effectiveness of density-adaptive graph construction.

LITERATURE REVIEW

Core Idea	Methodology	Limitations Identified	Paper Title	Authors & Year	Relevance to Our Project
Formalized spectral clustering using graph Laplacians and eigenvectors	Construct affinity matrix using global Gaussian kernel, normalize Laplacian, cluster eigenvectors using K-means	Uses a single global similarity scale (σ), sensitive to parameter choice, assumes uniform data density	On Spectral Clustering: Analysis and an Algorithm	Ng, Jordan, Weiss (2001)	Provides the theoretical foundation for spectral clustering and motivates why improving graph construction is necessary
Theoretical and intuitive understanding of spectral clustering	• Formulated clustering as a graph partitioning problem• Defined unnormalized and normalized graph Laplacians• Analyzed spectral properties and eigenvector behavior• Derived spectral clustering via graph cuts, random walks, and perturbation theory	• Highlights strong dependence on similarity graph construction• Does not propose adaptive similarity scaling• Notes lack of theory on optimal graph design	A Tutorial on Spectral Clustering	Ulrike von Luxburg (2007)	Provides the theoretical foundation for graph Laplacians and motivates operator-level analysis of similarity graphs
Sensitivity of spectral clustering to global scale parameter σ and inability to handle multi-scale data	• Introduced local (density-adaptive) scaling• Removed dependence on a single global σ• Improved clustering in multi-scale and cluttered data	• Defined local scale σ_i using k-th nearest neighbor distance• Constructed adaptive affinity: $A_{ij} = \exp(-\lVert x_i - x_j \rVert^2 / (\sigma_i \sigma_j))$• Built normalized Laplacian and analyzed eigenvectors	Self-Tuning Spectral Clustering	Zelnik-Manor & Perona (2004)	• Focuses mainly on clustering quality• Limited operator-level spectral analysis• Does not deeply study eigenvalue stability

PROBLEM STATEMENT

- Standard spectral clustering methods construct similarity graphs using a fixed global scale parameter, implicitly assuming uniform data density across the dataset. This assumption often fails for real-world datasets such as satellite imagery, where data points exhibit heterogeneous densities and varying local structures.

- As a result, fixed-scale graph Laplacians can lead to unstable spectral representations, sensitivity to parameter selection, and degraded clustering performance. There is a need for a principled graph construction approach that adapts to local data density in order to improve the robustness and stability of spectral clustering.

Project vision and mission

PROJECT VISION

To explore how graph construction influences the spectral properties of data beyond standard clustering pipelines.

To demonstrate that incorporating local density information leads to more robust and stable spectral representations.

To bridge theoretical concepts of graph Laplacians with practical applications in real-world image datasets.

To move beyond syllabus-level implementation by focusing on operator-level design and analysis.

PROJECT MISSION

- To construct a density-adaptive graph Laplacian using local neighborhood information.
- To analyze and compare the spectral behavior of fixed-scale and density-adaptive Laplacians.
- To evaluate robustness through eigenvalue spectra and spectral embeddings rather than only clustering accuracy.
- To apply the proposed approach to satellite image data and visually interpret the resulting spectral structures.
- To build a mathematically grounded, explainable, and defensible spectral clustering framework.

MATHEMATICAL FRAMEWORK OF DENSITY-ADAPTIVE SPECTRAL CLUSTERING

PROBLEM DEFINITION AND NOTATION

Dataset Representation

Let

$$\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$$

where:

N = total number of data samples (images)

$xi \in \mathbb{R}^d xi$=feature vector of the ith image

$d = 2048$ (ResNet50 embedding dimension)

Meaning

Each satellite image is converted into a point in a high-dimensional feature space.

Data Representation

Equation:

$$\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$$

Explanation:

•$\mathcal{X}$→ the entire dataset

•N→ total number of images (2000 in your case)

•Each x_i →one image represented as numbers

Next equation:

$$x_i \in \mathbb{R}^d$$

•This means each image is a vector

•$d = 2048$(from ResNet50)

•So each image is a point in 2048-dimensional space

📌 Why this matters

Clustering happens on numbers, not images.

Clustering Objective

Equation:

$$\mathcal{C} = \{C_1, C_2, \ldots, C_K\}$$

Explanation:

•$\mathcal{C}$→ final clustering result

•K→ number of clusters (5)

•C_k →the k-th cluster

Cluster properties:

$$C_k \subseteq \mathcal{X}$$

•Each cluster contains some images from the dataset

$$\bigcup_{k=1}^{K} C_k = \mathcal{X}$$

•Every image must belong to some cluster

$$C_i \cap C_j = \emptyset \ (i \neq j)$$

•No image belongs to two clusters

Graph Representation

Equation:

$$G = (V, E)$$

Explanation:

- G → graph
- V → vertices (nodes)
- E → edges (connections)

Nodes:

$$V = \{1, 2, \ldots, N\}$$

- Each node represents **one image**

Distance:

$$d_{ij} = \| x_i - x_j \|_2$$

- Euclidean distance between image i and image j
- Smaller distance → more similar images

Why graph?

Because clusters are **connectivity-based**, not centroid-based.

k-Nearest Neighbor Graph
Equation:

$$\mathcal{N}_k(i)$$

•Set of **k closest images** to image i

Why kNN?

•Avoids connecting unrelated images
•Preserves local structure
•Makes graph sparse and stable

Affinity Matrix
Equation:

$$W \in \mathbb{R}^{N \times N}$$

•W_{ij} →similarity between image $i$and image j
•Larger value → more similar

Global Sigma (baseline)

$$W_{ij} = \exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right)$$

•Same $\sigma$for all images
•Fails when data density varies

Density-Adaptive (Self-Tuning) Affinity Local scale:

$$\sigma_i = d(x_i, x_{i,k})$$

•Distance from image i to its **k-th nearest neighbor**

•Represents **local density**

- Dense region → small σ_i
- Sparse region → large σ_i

Final similarity:

$$W_{ij} = \exp\left(-\frac{d_{ij}^2}{\sigma_i \sigma_j}\right)$$

THIS is the heart of your project

•Similarity adapts to **local density**

•Prevents over-connection and under-connection

Degree Matrix

Equation:

$$D_{ii} = \sum_{j=1}^{N} W_{ij}$$

•Total connection strength of node i

$$D = \mathrm{diag}(D_{11}, D_{22}, \dots)$$

Why needed?

•Used for normalization

•Balances influence of dense nodes

Normalized Graph Laplacian
Equation:

$$L = D^{-1/2} W D^{-1/2}$$

Meaning
•Removes bias caused by node degree
•Ensures fair clustering across densities
This matrix encodes **global structure** of the graph.

Eigenvalue Problem
Equation:

$$Lu = \lambda u$$

•u→ eigenvector
•λ→ eigenvalue
Why eigenvectors?
•They reveal **connected components**
•Smoothest directions on the graph

Spectral Embedding
Select top K eigenvectors:

$$U = [u_1, u_2, \ldots, u_K]$$

Each row corresponds to **one image**.

Normalize rows:

$$y_i = \frac{u_i}{\| u_i \|_2}$$

Why normalize?
•Removes scale differences
•Improves separation

Final Clustering

- Apply K-Means on $\{y_i\}$
- Output: cluster labels

Important

K-Means works **because spectral space is already clean**.

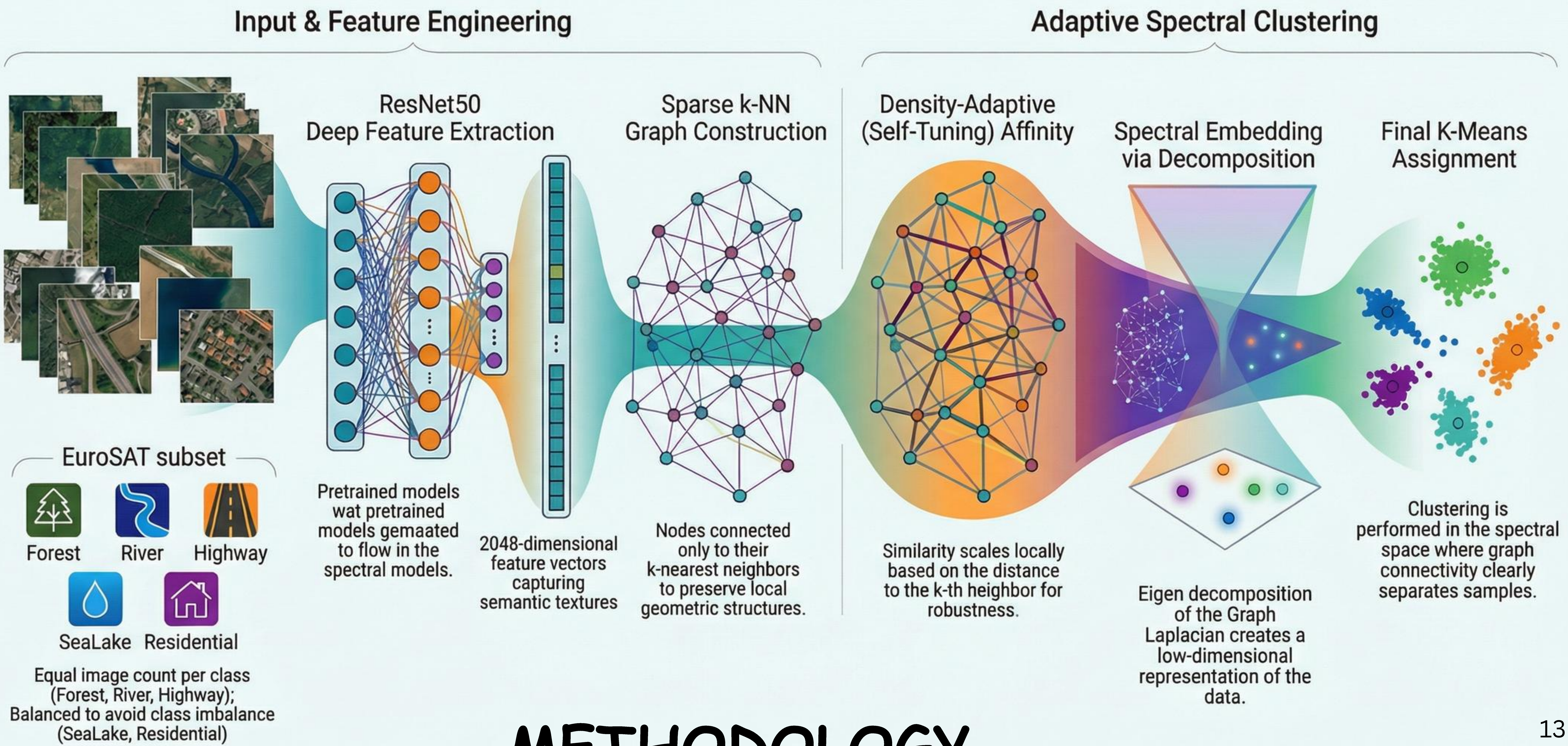
Evaluation Metrics

Accuracy:

$$\mathrm{ACC} = \frac{1}{N}\max_{\pi}\sum \mathbf{1}\left(y_i = \pi(t_i)\right)$$

- Measures best label matching

Mapping with Precision: The Density-Adaptive Spectral Clustering Workflow



-METHODOLOGY-

EXPERIMENTAL SETUP

Dataset Used: EuroSAT

- Selected Classes: Forest, River, Sea Lake, Residential, Highway
- Images per Class: 400
- Total Images: 2000
- Image Preprocessing:
  - Images resized to 224 × 224
  - Pixel values normalized
- Feature Extraction:
  - Pretrained ResNet50 used
  - Output feature size: 2048
  - Each image → 2048-dimensional vector

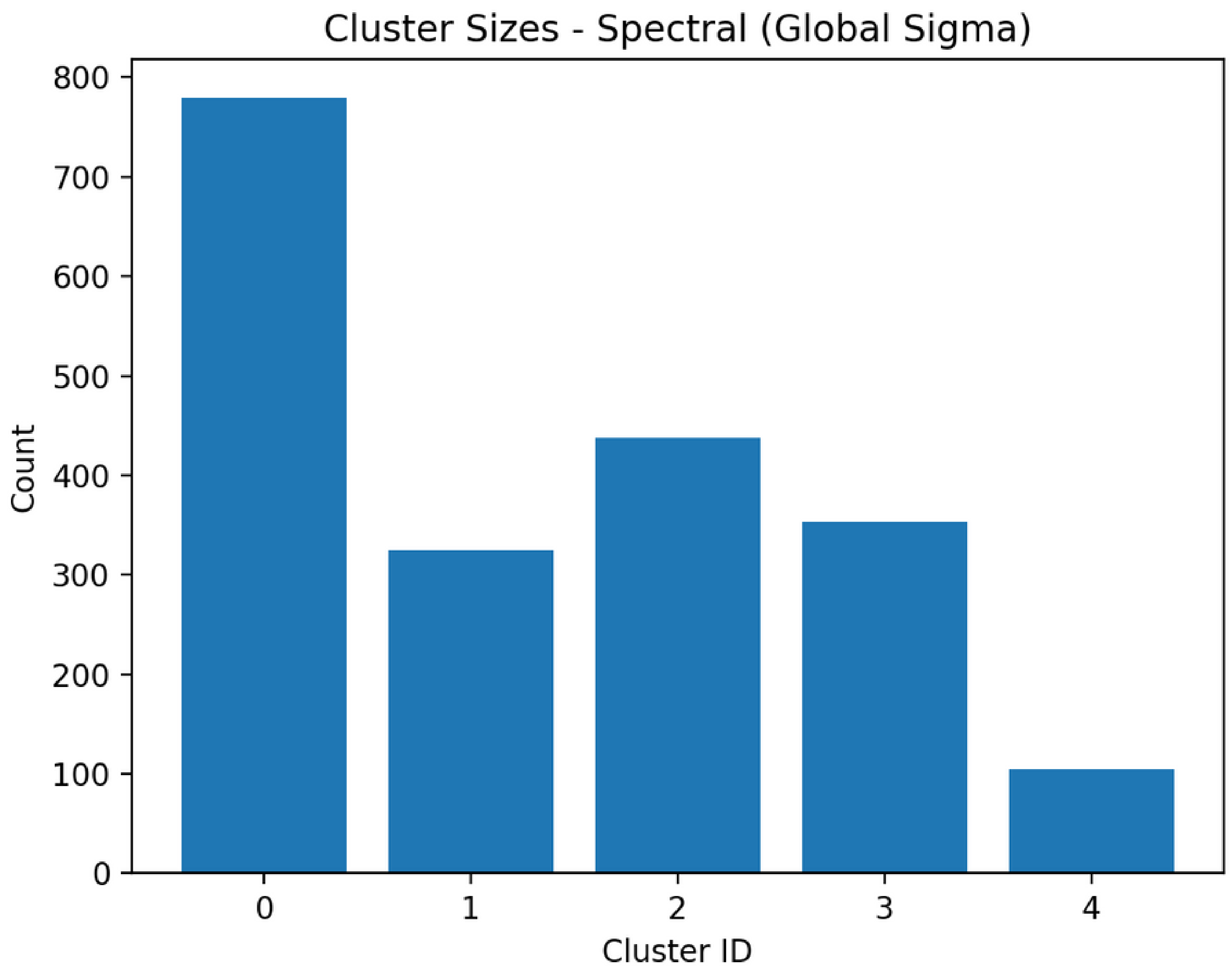
Graph Construction:

- Each image treated as a node
  - k-Nearest Neighbors (k-NN) graph used
  - Distance metric: Euclidean distance
- Clustering Methods Compared:
  - K-Means on ResNet features
  - Spectral clustering with global sigma
  - Self-tuning (density-adaptive) spectral clustering

Implementation Tools:

- Python
  - PyTorch
  - NumPy
  - Scikit-learn
- Evaluation Metrics:
  - Clustering Accuracy (ACC)
  - Normalized Mutual Information (NMI)
  - Adjusted Rand Index (ARI
- Qualitative Evaluation:
  - Cluster-wise image montages
  - Cluster purity analysis

RESULTS AND ANALYSIS



Main Observation:

Cluster 0 is very large (~780)

This means:

Global sigma spectral clustering merged multiple types of images into one cluster.

So cluster 0 is not "one clean class", it's a mixed cluster.

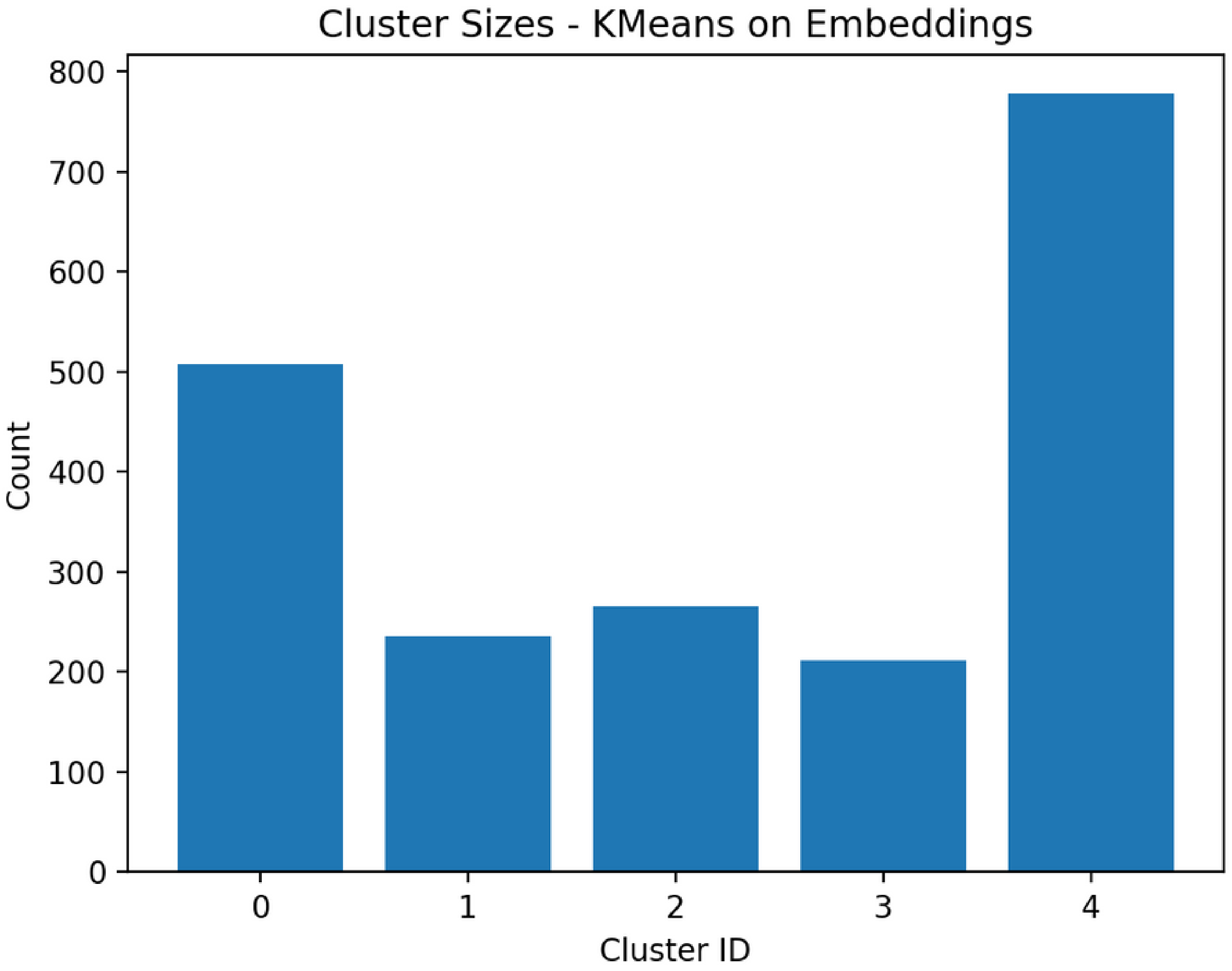
Cluster 4 is very small (~105)

This means:

Global sigma created a tiny cluster containing a very specific type of images (usually very similar ones).

Cluster 0 ≈ 780 images
Cluster 1 ≈ 325 images
Cluster 2 ≈ 440 images
Cluster 3 ≈ 350 images
Cluster 4 ≈ 105 images

RESULTS AND ANALYSIS



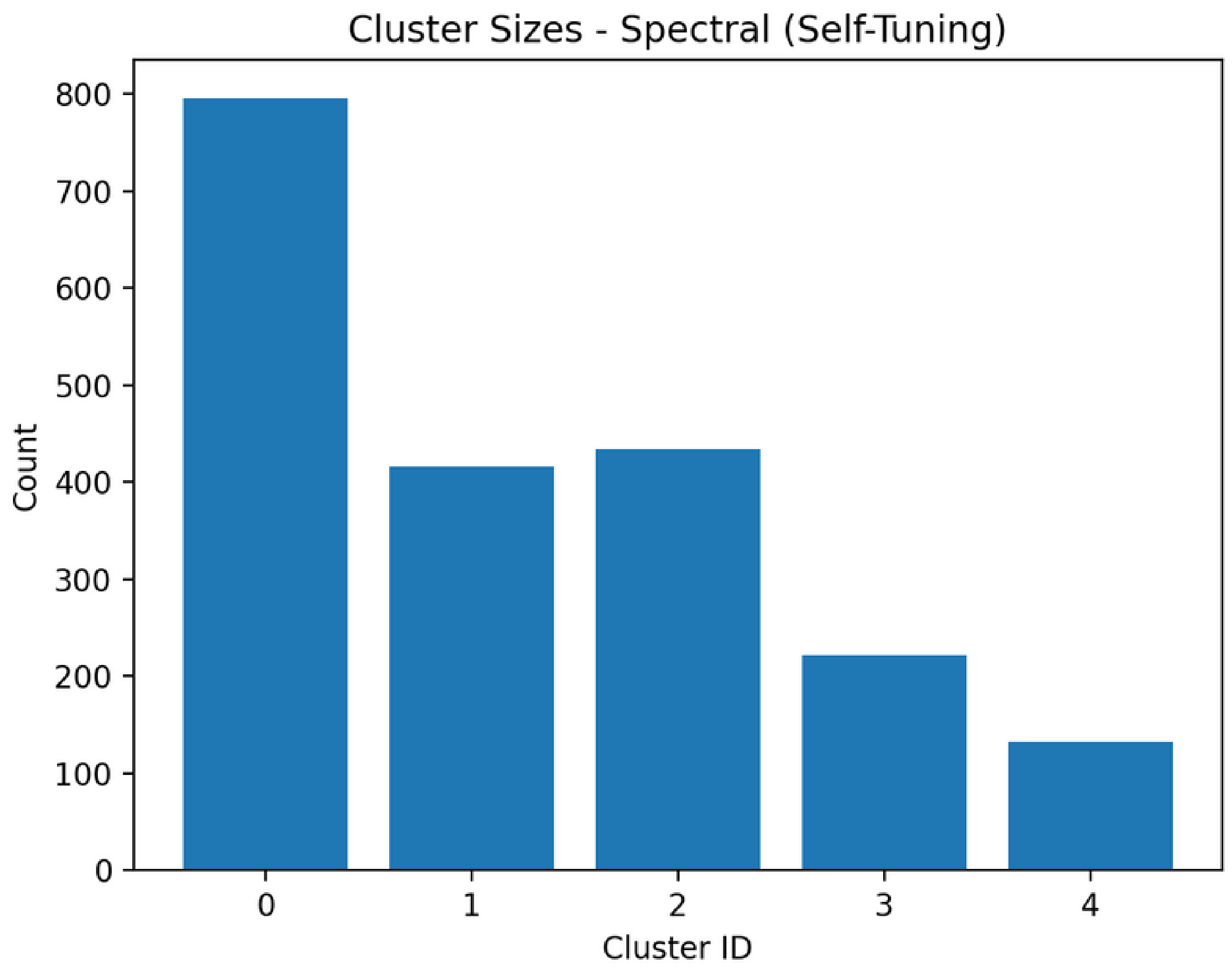
Why cluster 4 is very big?
Because KMeans groups based on distance in the embedding space.
So if many images look "similar" in the ResNet feature space, KMeans will push them into the same cluster.

That's why cluster 4 became huge:
it is mixing multiple types of images, not just one class.

What this indicates about K Means ?
K Means creates imbalanced clusters
Means K Means is weaker for this dataset compared to spectral methods
It can't separate complex classes like River vs Highway vs Residential properly

Cluster 0 ≈ 508 images
Cluster 1 ≈ 236 images
Cluster 2 ≈ 266 images
Cluster 3 ≈ 212 images
Cluster 4 ≈ 778 images

RESULTS AND ANALYSIS



Why cluster sizes are different?
Because self-tuning does not try to keep equal sizes.
It groups images based on similarity + local density

Small cluster (like Cluster 4):
Means:
those images are very similar to each other
they form a tight group
Example: SeaLake images (pure water) often become a small but clean cluster.

Big cluster (like Cluster 0) :
Means:
The images are more diverse / mixed
Example: River type can overlap with forest edges, highways, and residential boundaries → so it becomes large.

What this plot tells about self-tuning quality ?
Compared to KMeans, self-tuning:
It makes more meaningful groups .
It creates compact pure clusters (small ones) .
It improves clustering results overall .

Cluster 0 ≈ 795 images
Cluster 1 ≈ 416 images
Cluster 2 ≈ 434 images
Cluster 3 ≈ 222 images
Cluster 4 ≈ 133 images
Total = 2000 images

RESULTS AND ANALYSIS

Method	Cluster_ID	Cluster_Size	Majority_Class	Majority_Count	Purity_%
KMeans on Embeddings	0	508	Forest	385	75.79
KMeans on Embeddings	1	236	Residential	201	85.17
KMeans on Embeddings	2	266	SeaLake	265	99.62
KMeans on Embeddings	3	212	Residential	188	88.68
KMeans on Embeddings	4	778	River	381	48.97
Spectral (Global Sigma)	0	779	River	386	49.55
Spectral (Global Sigma)	1	325	Residential	325	100
Spectral (Global Sigma)	2	438	Forest	392	89.5
Spectral (Global Sigma)	3	353	SeaLake	349	98.87
Spectral (Global Sigma)	4	105	Residential	69	65.71
Spectral (Self-Tuning)	0	795	River	392	49.31
Spectral (Self-Tuning)	1	416	Residential	394	94.71
Spectral (Self-Tuning)	2	434	Forest	393	90.55
Spectral (Self-Tuning)	3	222	SeaLake	218	98.2
Spectral (Self-Tuning)	4	133	SeaLake	133	100

CONCLUSION

This work presented a deep feature-based clustering framework for EuroSAT satellite images using ResNet50 embeddings and spectral clustering. A comparison between KMeans, global sigma spectral clustering, and self-tuning density-adaptive spectral clustering was performed. The results show that the proposed self-tuning approach improves clustering consistency and interpretability by adapting similarity measures to local data density.

Future Work

Future improvements can include:

- testing on all EuroSAT classes for scalability
- using Vision Transformers (ViT) embeddings for stronger features
- using approximate nearest neighbor methods (FAISS) for faster graph building
- experimenting with different graph kernels and normalization techniques
- replacing KMeans with more robust spectral embedding clustering (e.g., GMM)

Thank you!