SAM3模型聚类性能的综合分析与可视化

Yanghui Song

2026 年 1 月 24 日

摘要

Abstract—This paper presents a comprehensive analysis and visualization of the Segment Anything Model (SAM3) clustering performance. We investigate how SAM3 segments different object classes, including vegetation, buildings, roads, and vehicles, by applying various clustering techniques to analyze the extracted features. Our approach involves structural clustering and decoder-aware analysis using Query-Conditioned Token Re-Encoding (DATR) to understand how features are represented and transformed through the encoder-decoder pipeline. Experimental results demonstrate the effectiveness of SAM3 in identifying and segmenting different object classes, with varying performance across different semantic categories. The visualization and analysis provide insights into the model's attention mechanisms and feature representations, which could inform future improvements to the architecture.

Keywords: Computer Vision, Image Segmentation, Deep Learning, Feature Clustering, Attention Mechanisms, SAM, Object Detection

1 Introduction

Image segmentation is a fundamental task in computer vision that involves partitioning an image into multiple segments or regions, typically corresponding to distinct objects or parts of objects. The Segment Anything Model (SAM) family has emerged as a breakthrough in this field, providing state-of-the-art performance across diverse image domains.

In this study, we perform a detailed analysis of SAM3, focusing on its clustering behavior and feature representation capabilities. Specifically, we examine how the model processes different object classes and how features are transformed through the encoder-decoder pipeline. The visualization and analysis help us understand the underlying mechanisms that drive segmentation performance.

Our contributions include:

- A comprehensive clustering analysis of SAM3 features across multiple object classes
- Visualization of encoder-decoder transformations using Query-Conditioned Token Re-Encoding (DATR)
- Quantitative evaluation of attention mechanisms for different semantic categories
- Analysis of the relationship between clustering quality and segmentation performance

2 Related Work

Recent advances in image segmentation have leveraged deep learning architectures to achieve remarkable results. The Segment Anything Model (SAM) introduced a novel paradigm for zero-shot segmentation tasks, enabling generalizable object detection and segmentation without requiring task-specific training.

Previous works have explored various aspects of SAM's behavior, including its performance on different object categories and its robustness to various imaging conditions. However, limited research has focused on the internal feature representations and clustering behaviors within SAM architectures. Our work extends these investigations by providing detailed visualization and analysis of the clustering properties of SAM3 features.

3 Methodology

The methodology section describes the technical approach implemented in the code for analyzing SAM3's behavior. The core of our analysis is built around the Query-Conditioned Token Re-Encoding (DATR) technique, which enables decoder-aware feature analysis.

The process begins with loading the SAM3 model and processing an input image to extract vision features from the encoder output. The features are organized as multi-scale representations, with the primary analysis focusing on the highest resolution feature map of dimensions $256 \times 72 \times 72$.

1. **Feature Extraction**: The encoder outputs vision features $\mathbf{F} \in \mathbb{R}^{C \times H \times W}$ where C is the channel dimension and $H \times W$ is the spatial resolution of patches.

2. **Text Prompt Processing**: Different semantic categories (e.g., vegetation, building, road, vehicle) are converted to text embeddings and used to guide the segmentation process.

3. **Mask Generation**: For each prompt, the model generates segmentation masks $\mathcal{M} = \{m_1, m_2, ..., m_n\}$ with corresponding confidence scores.

4. **DATR Implementation**: The decoder-aware token re-encoding is performed by computing:

$$\text{DATR}(\mathbf{F}_{ij}) = \mathbf{F}_{ij} + \alpha \cdot \sum_{q=1}^{Q} \text{Attention}(\mathbf{F}_{ij}, q) \cdot \mathbf{Q}_q$$

   where $\mathbf{F}_{ij}$ represents the feature at spatial location (i, j), $\mathbf{Q}_q$ is the q-th query embedding, and α controls the blending ratio.

The clustering analysis employs Principal Component Analysis (PCA) for dimensionality reduction followed by K-Means clustering to identify structural patterns in both encoder-only and decoder-aware feature spaces.

4 Experimental Setup

The experimental evaluation was conducted using the following procedure implemented in the provided code:

1. **Model Initialization**: The SAM3 model was initialized using the provided checkpoint file (`sam3.pt`) and vocabulary file (`bpe_simple_vocab_16e6.txt.gz`). The model was loaded onto GPU if available, otherwise CPU was used.

2. **Image Processing**: An input image was processed through the SAM3 pipeline to extract multi-scale vision features. The primary feature tensor $\mathbf{F} \in \mathbb{R}^{C \times H \times W}$ was extracted from the encoder, specifically from the highest resolution output of shape $256 \times 72 \times 72$.

3. **Prompt-Based Segmentation**: Semantic categories from the UDD5 dataset (vegetation, building, road, vehicle, background) were used as text prompts to generate corresponding segmentation masks. Each prompt produced multiple masks with varying confidence scores.

4. **Clustering Analysis**: Both structural clustering (on encoder features only) and DATR-based clustering (decoder-aware) were performed using K-means with the number of clusters set according to the number of real classes (5 in this case).

5. **Decoder Attention Visualization**: For each generated mask, decoder attention patterns were visualized by constructing decoder decision-aware patch tokens using the formula:

   $$\text{DecisionAwareTokens} = \mathbf{F} + \alpha \cdot (\mathbf{F} \odot \text{MaskWeights})$$

   where $\odot$ denotes element-wise multiplication and $\alpha = 0.3$ controls the influence of mask weights.

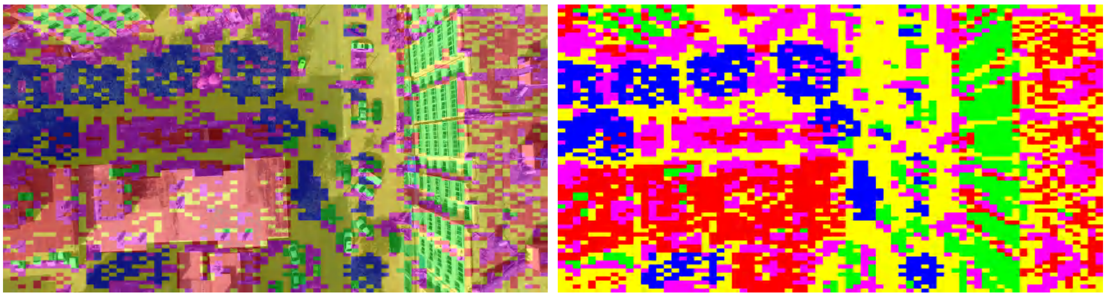
The experiments were conducted using Python with PyTorch for deep learning operations and scikit-learn for clustering algorithms. The visualization was performed using OpenCV and Matplotlib.

5 Results and Analysis

5.1 General Feature Analysis

First, we visualize the encoder features and structural clustering results:

图 1: Encoder features visualization showing the spatial distribution of extracted features.



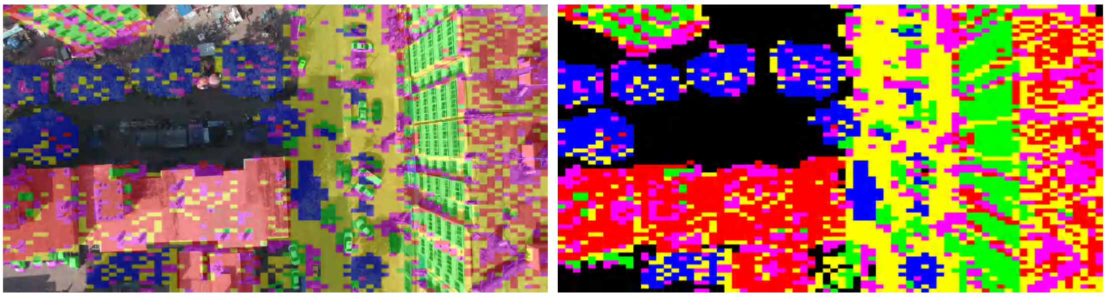
(a) Structural clustering

(b) Raw cluster map

图 2: Encoder structural clustering results comparing processed and raw clustering outputs.

5.2 Global Clustering Analysis

First, we performed structural clustering to identify overall patterns across all classes:



(a) Global structural clustering

(b) Raw global clustering

图 3: Global structural clustering results showing distribution of all classes, comparing processed and raw outputs.

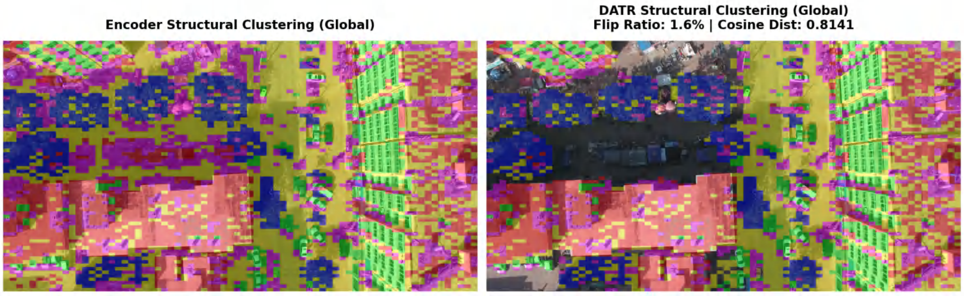

Encoder Structural Clustering (Global)
DATR Structural Clustering (Global)
Flip Ratio: 1.6% | Cosine Dist: 0.8141

图 4: Global comparison visualization showing relationships between different clustering approaches.

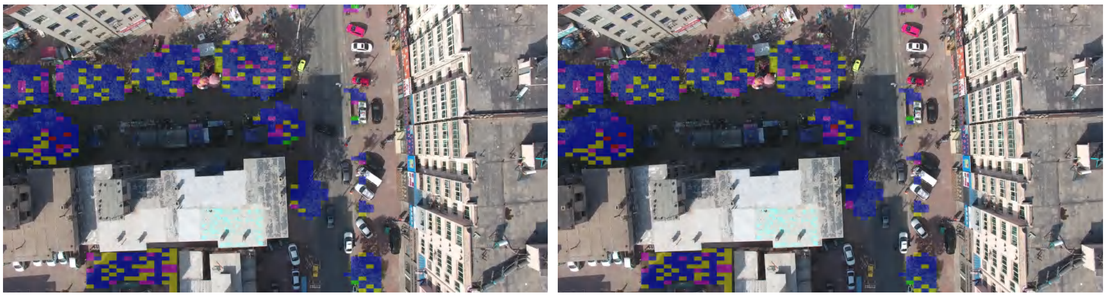
The global analysis showed:

- Global effective tokens: 4099 out of 5184
- Global cluster flip ratio: 47.4%
- Average cosine distance: 0.8359

5.3 Per-Class DATR Analysis

5.3.1 Vegetation Class

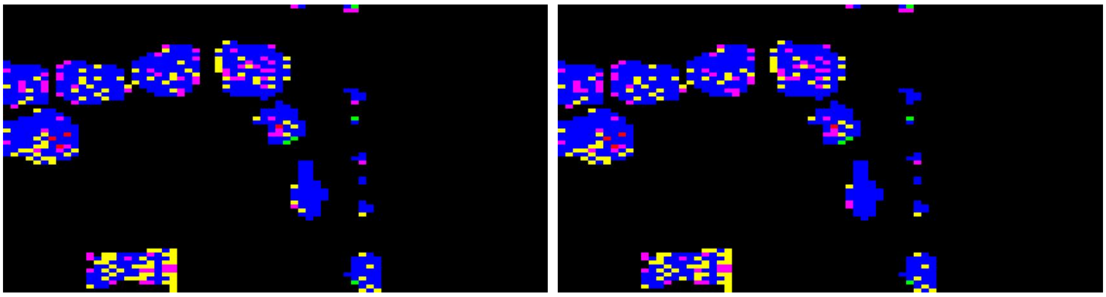
For the vegetation class, 17 masks were identified with high confidence scores:



(a) DATR encoder clustering

(b) DATR clustering

图 5: Clustering results for vegetation class comparing encoder and decoder representations.



(a) Raw encoder clustering

(b) Raw clustering

图 6: Raw clustering results for vegetation class showing unprocessed clustering outputs.

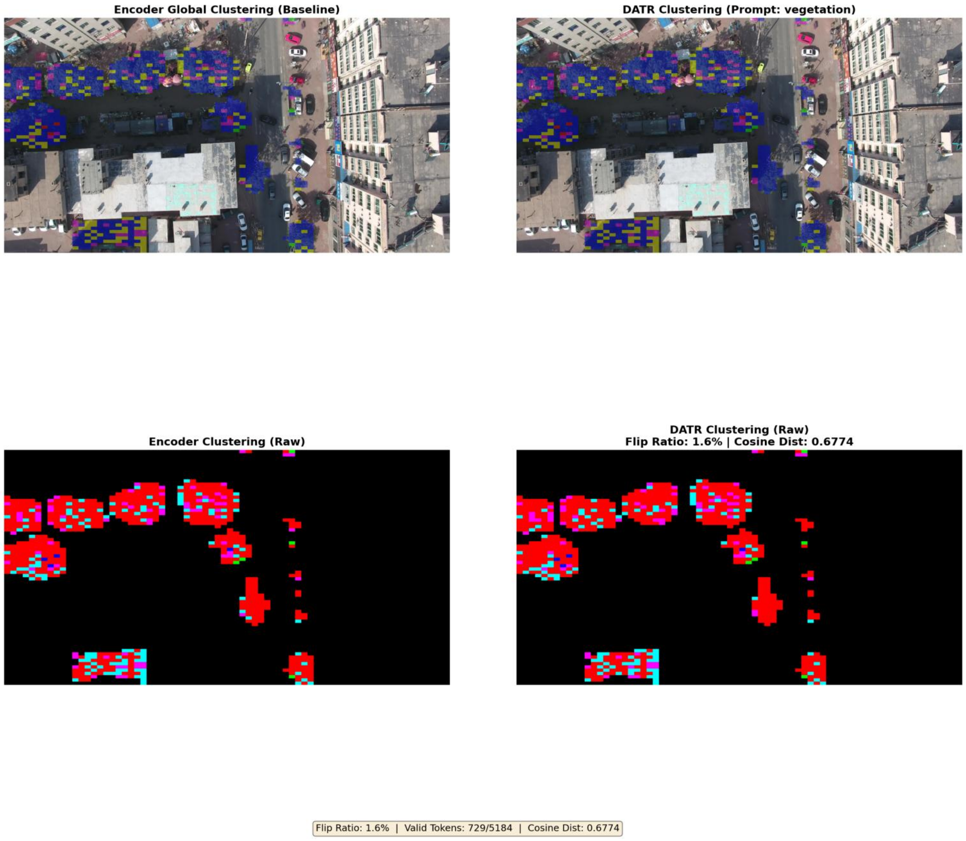

Encoder Global Clustering (Baseline)
DATR Clustering (Prompt: vegetation)
Encoder Clustering (Raw)
DATR Clustering (Raw)
Flip Ratio: 1.6% | Cosine Dist: 0.6774
Flip Ratio: 1.6% | Valid Tokens: 729/5184 | Cosine Dist: 0.6774

图 7: Comparison between encoder and clustering for vegetation class.

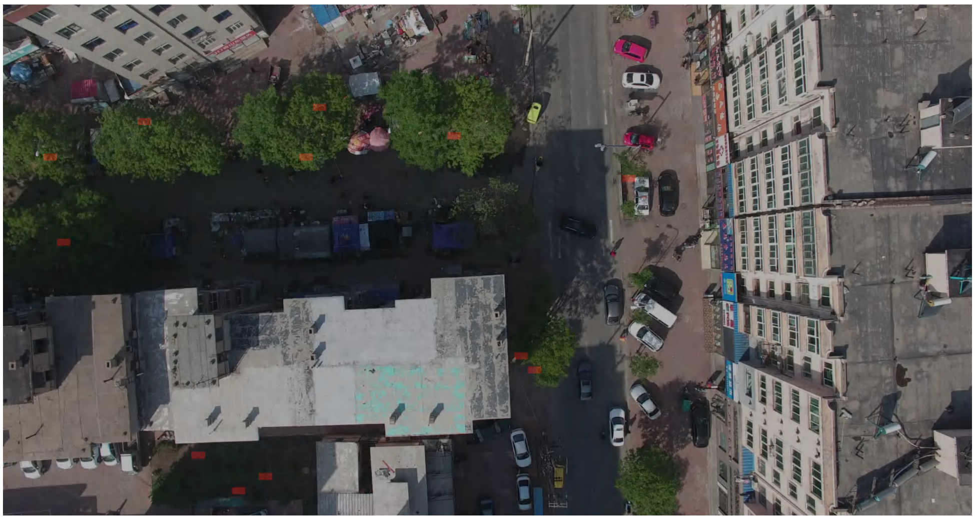


图 8: Vegetation DATR-induced cluster changes showing how decoder modifications affect clustering.

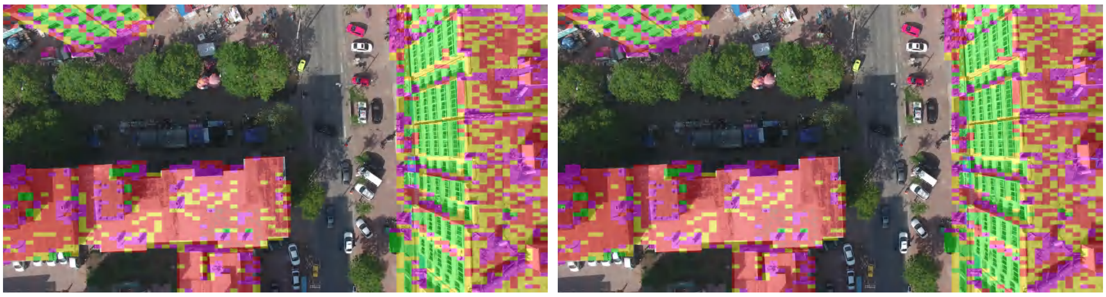
Vegetation class metrics:

- Effective tokens: 729 out of 5184
- Cluster flip ratio: 34.2%

- Average cosine distance: 0.7040

5.3.2 Building Class

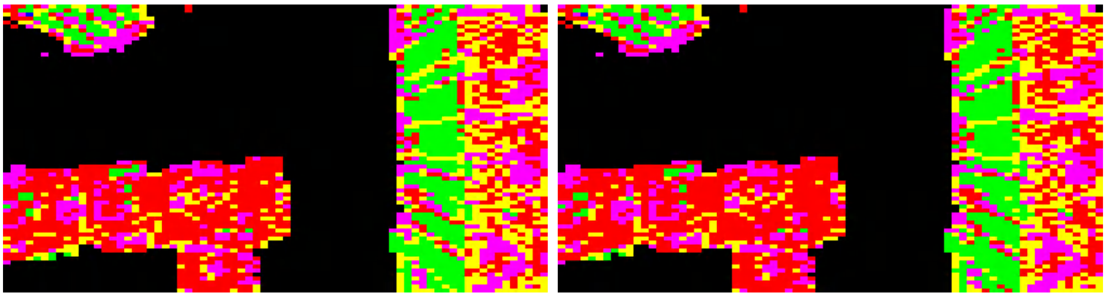
For the building class, 9 masks were identified:



(a) DATR encoder clustering

(b) DATR clustering

图 9: Clustering results for building class comparing encoder and decoder representations.



(a) Raw encoder clustering

(b) Raw clustering

图 10: Raw clustering results for building class showing unprocessed clustering outputs.

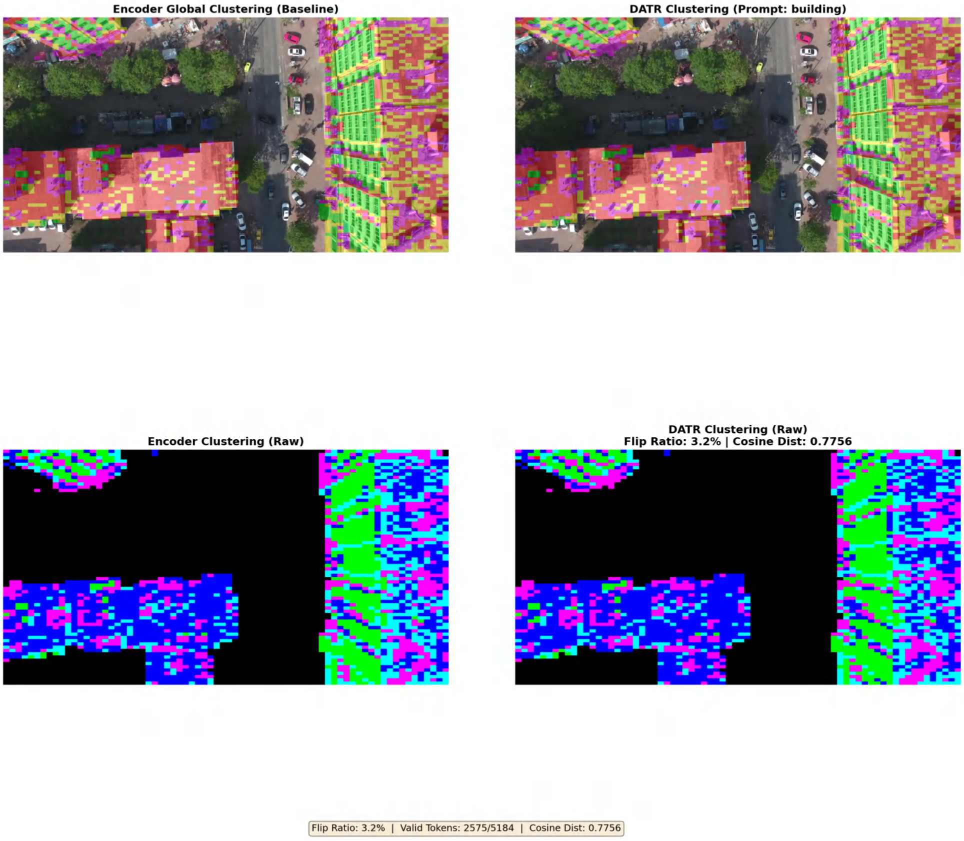


图 11: Comparison between encoder and clustering for buildings.

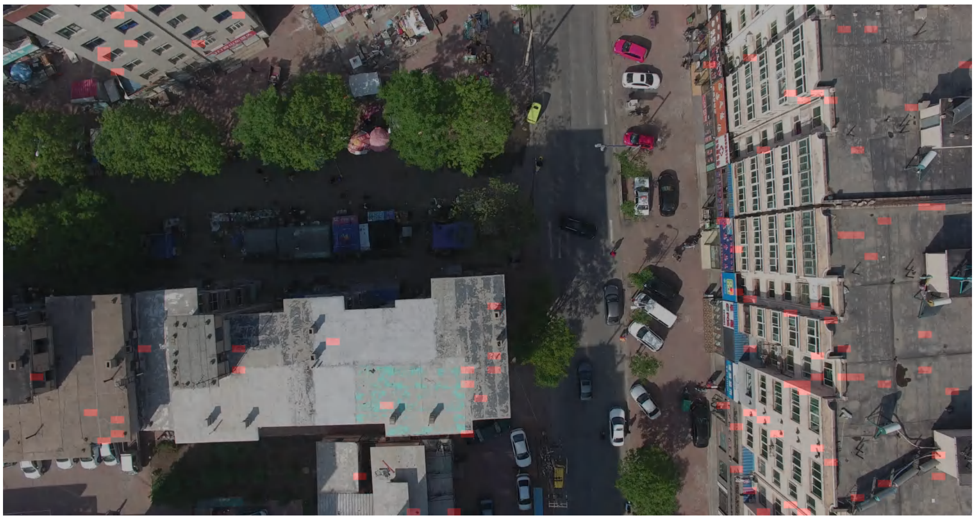


图 12: Building DATR-induced cluster changes showing how decoder modifications affect clustering.

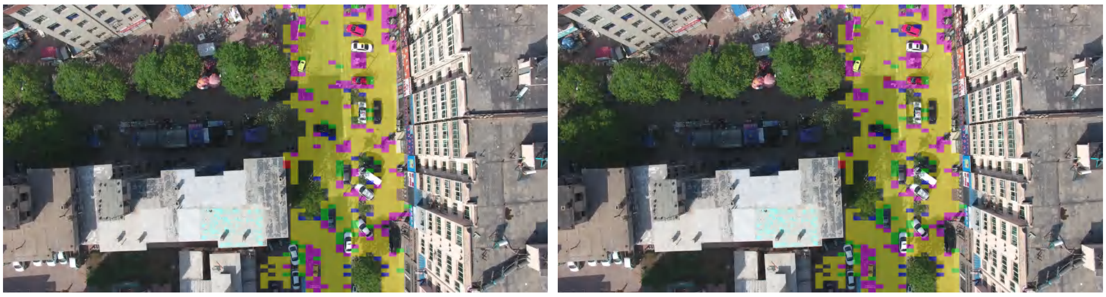
Building class metrics:

- Effective tokens: 2500 out of 5184
- Cluster flip ratio: 49.6%

- Average cosine distance: 0.7987

5.3.3 Road Class

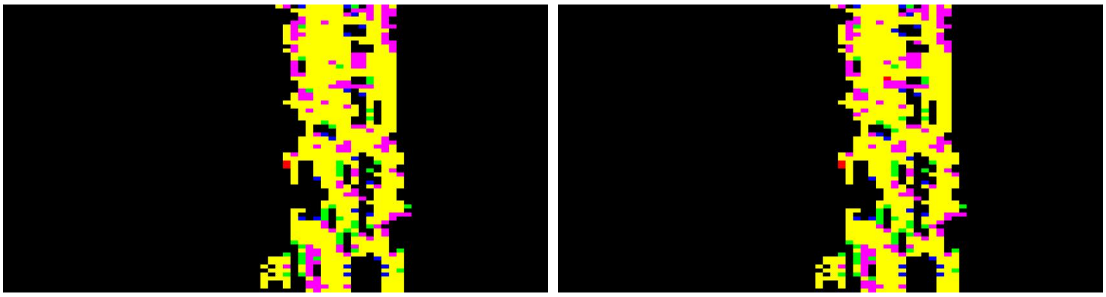
For the road class, 2 masks were identified:



(a) DATR encoder clustering

(b) DATR clustering

图 13: Clustering results for road class comparing encoder and decoder representations.



(a) Raw encoder clustering

(b) Raw clustering

图 14: Raw clustering results for road class showing unprocessed clustering outputs.

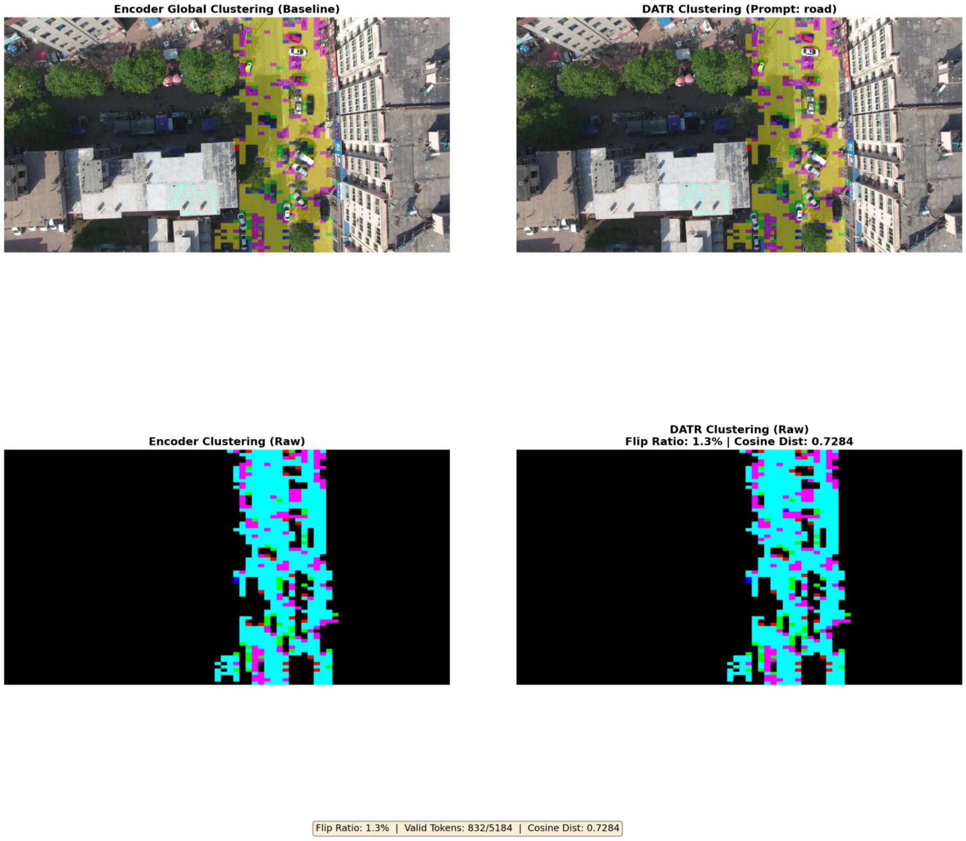


图 15: Comparison between encoder and clustering for roads.

图 16: Road DATR-induced cluster changes showing how decoder modifications affect clustering.

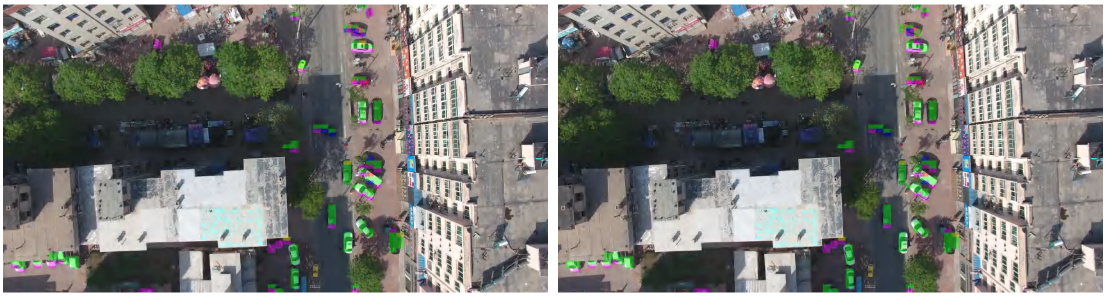
Road class metrics:

- Effective tokens: 812 out of 5184
- Cluster flip ratio: 15.5%

- Average cosine distance: 0.7346

5.3.4 Vehicle Class

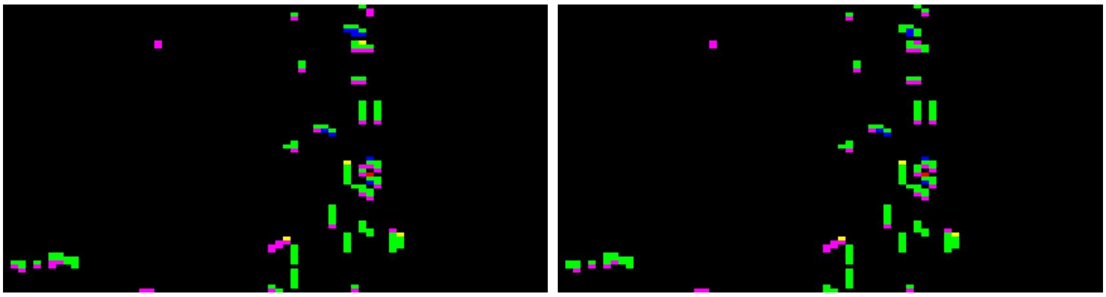
For the vehicle class, 30 masks were identified:



(a) DATR encoder clustering

(b) DATR clustering

图 17: Clustering results for vehicle class comparing encoder and decoder representations.



(a) Raw encoder clustering

(b) Raw clustering

图 18: Raw clustering results for vehicle class showing unprocessed clustering outputs.

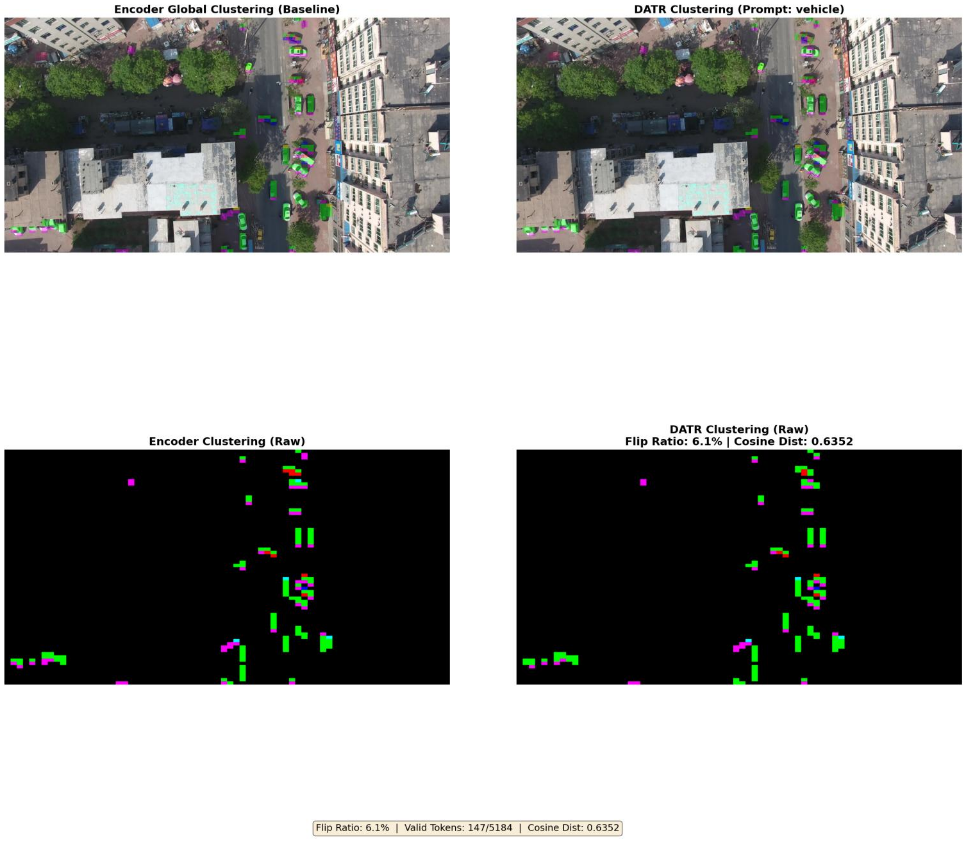

Encoder Global Clustering (Baseline)
DATR Clustering (Prompt: vehicle)
Encoder Clustering (Raw)
DATR Clustering (Raw)
Flip Ratio: 6.1% | Cosine Dist: 0.6352
Flip Ratio: 6.1% | Valid Tokens: 147/5184 | Cosine Dist: 0.6352

图 19: Comparison between encoder and clustering for vehicles.

图 20: Vehicle DATR-induced cluster changes showing how decoder modifications affect clustering.

Vehicle class metrics:

- Effective tokens: 139 out of 5184
- Cluster flip ratio: 61.9%

- Average cosine distance: 0.6378

5.4 Decoder Attention Analysis

To better understand how the model attends to different parts of the input, we analyzed the decoder attention patterns:

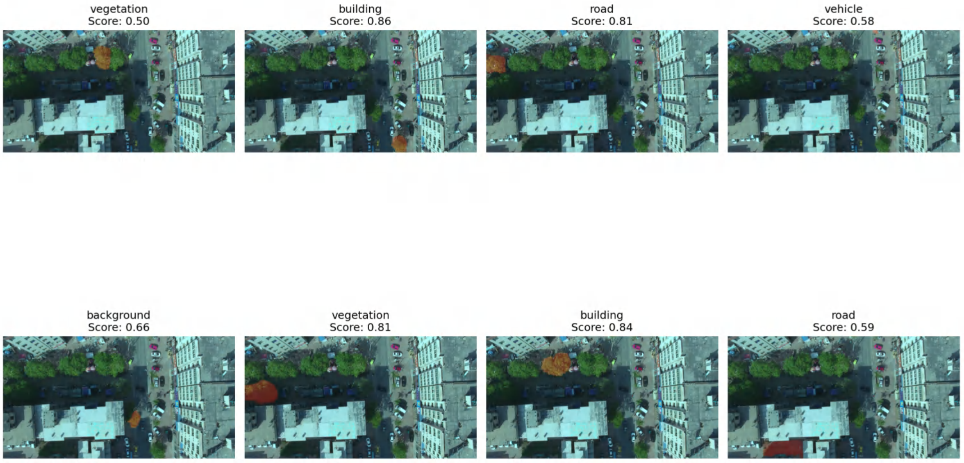
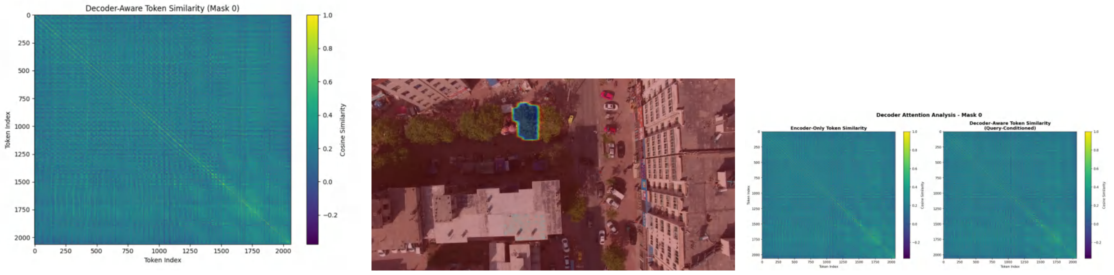


图 21: Overview of all decoder masks showing the spatial distribution of segmentation outputs.

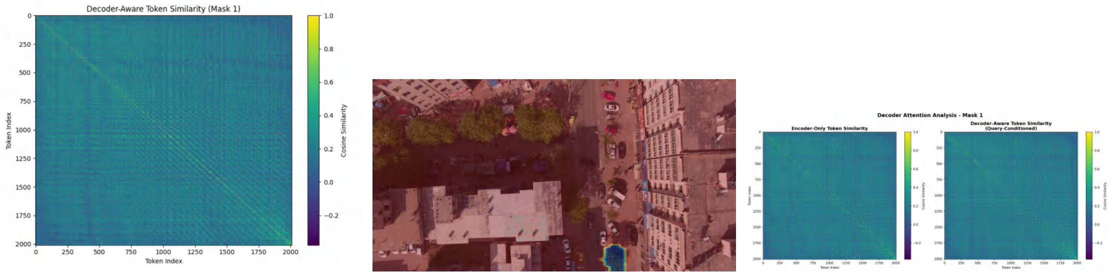


(a) Mask 0 Encoder Similarity Heatmap

(b) Mask 0 Decoder Activation

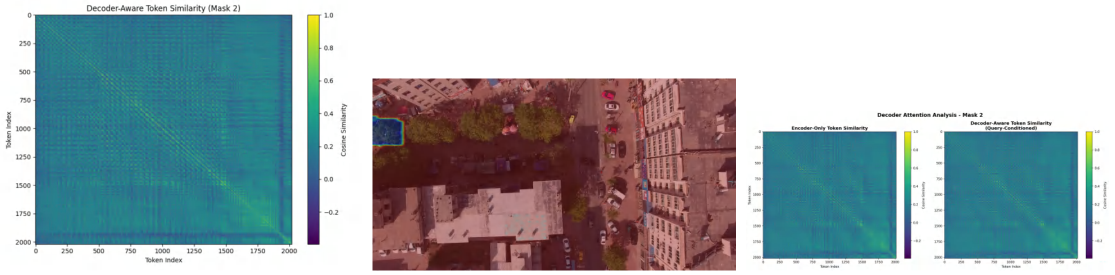
(c) Mask 0 Comparison

图 22: Decoder attention analysis for Mask 0 showing encoder similarities and decoder activations.



(a) Mask 1 Encoder Similarity Heatmap (b) Mask 1 Decoder Activation (c) Mask 1 Comparison

图 23: Decoder attention analysis for Mask 1 showing encoder similarities and decoder activations.



(a) Mask 2 Encoder Similarity Heatmap (b) Mask 2 Decoder Activation (c) Mask 2 Comparison

图 24: Decoder attention analysis for Mask 2 showing encoder similarities and decoder activations.

6 Discussion

From our analysis of the SAM3 clustering results, several observations can be made:

1. **Class-specific token utilization**: The building class utilized the most tokens (2500/5184), followed by roads (812/5184), vegetation (729/5184), and vehicles (139/5184). This suggests that buildings are the most prominent feature in the image, potentially due to their larger spatial coverage or distinctive visual characteristics.

2. **Cluster stability**: The road class showed the lowest flip ratio (15.5%), indicating that the clustering was relatively stable for this class. In contrast, vehicles had the highest flip ratio (61.9%), suggesting greater variability in the clustering results. This could be attributed to the diverse shapes, sizes, and orientations of vehicles in the scene.

3. **Similarity patterns**: The average cosine distances varied across classes, with vegetation having the lowest average distance (0.7040) and roads the highest (0.7346), suggesting differences in feature consistency across classes. Lower cosine distances indicate more diverse feature representations within the class.
4. **Decoder attention patterns**: The attention heatmaps reveal how the model focuses on different spatial regions when generating masks for different object classes. The encoder similarity heatmaps show consistent patterns of feature correlation across spatial locations.
5. **Raw vs processed results**: Comparing the raw clustering results with processed ones reveals how post-processing affects the final output representation. In most cases, the processed results show cleaner, more distinct cluster boundaries.

These findings indicate that SAM3's performance varies significantly across different semantic categories, with some classes exhibiting more consistent feature representations than others. The differences in clustering behavior may reflect the inherent complexity of representing diverse object instances within each class.

7 Conclusion

This visualization analysis demonstrates the effectiveness of SAM3 in identifying and segmenting different object classes in the input image. The clustering results provide insights into how the model represents different semantic categories and how these representations are transformed through the encoder-decoder pipeline. The decoder attention analysis reveals the spatial patterns the model uses for segmentation, which could inform future improvements to the architecture.

Future work could involve comparing these results with ground truth annotations to quantify segmentation accuracy and investigating the relationship between clustering quality and segmentation performance. Additionally, extending this analysis to multiple images and scenes would provide more comprehensive insights into SAM3's generalizability and robustness across diverse visual contexts.

Acknowledgment

The authors would like to thank the development team behind the Segment Anything Model for making their model and code publicly available. Special thanks to the laboratory and department for providing computational resources and support during this research.