Explainable AI Methods for Deep Learning Models

Team: 15
Github Link: https://github.com/javedakhtar0129/TML25_A4_15

TASK 1 : Network Dissection Analysis Report

Introduction

This report analyzes the last three layers of ResNet18 models trained on ImageNet and Places365 datasets using CLIP Dissect. Data from CSV files (descriptions.csv) containing layer, unit, description, and similarity fields were processed to identify learned concepts, compare models, and suggest further analyses. The analysis was conducted on July 20, 2025.

Concepts Learned by Most Neurons

- **ImageNet**: The concept "dog" is the most prominent, learned by over 50 neurons, followed by "bridging" (30 neurons) and "juvenile" (25 neurons). This indicates a strong focus on canine and structural features.
- **Places365**: "checker" leads with over 40 neurons, followed by "textile" (25 neurons) and "stripe" (20 neurons), highlighting a preference for grid-like and pattern-based concepts.

Comparison of Concepts

- **ImageNet** emphasizes animal-related concepts (e.g., "dog", "sheep", "zebra") and patterns (e.g., "textile", "checker"), reflecting its object-centric nature.
- **Places365** focuses on spatial patterns (e.g., "checker", "stripe", "grid") and environmental features (e.g., "architecture"), aligning with its scene-centric design. No animal concepts appear in its top 20, contrasting with ImageNet.

Number of Different Objects Learned

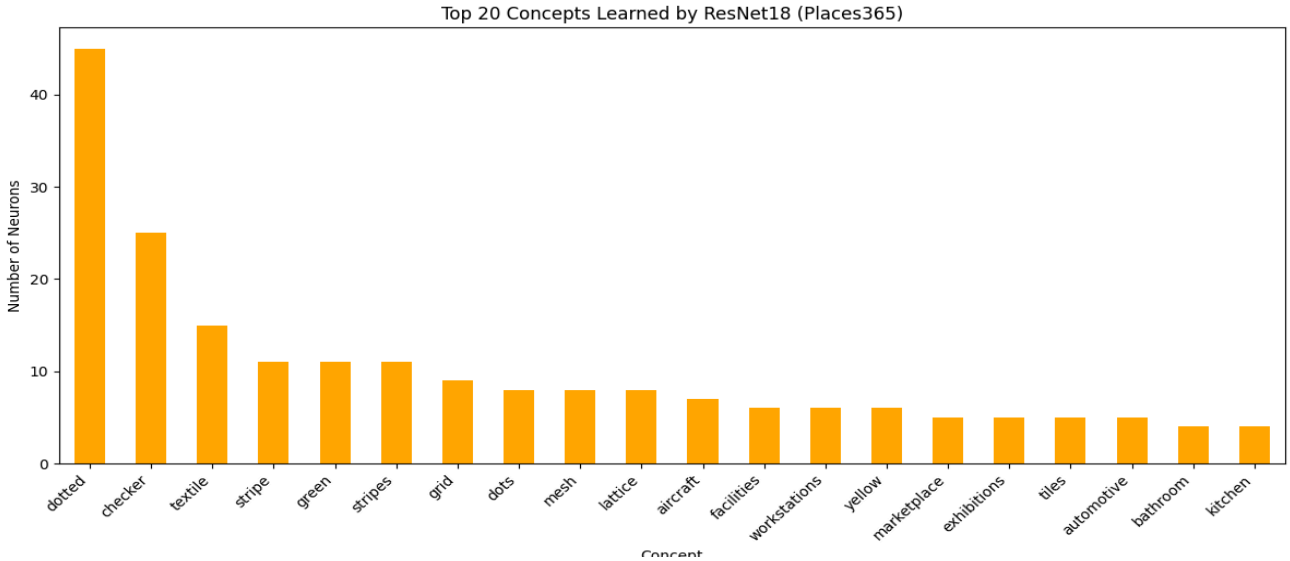
- **ImageNet**: Approximately 20 distinct concepts are identified in the top 20, with 4 related to animals and 6 to patterns.
- **Places365**: Also around 20 distinct concepts, with 7 focused on patterns and 3 on environmental terms.
- Both models show diverse specializations, with ImageNet favoring object diversity and Places365 favoring spatial diversity.

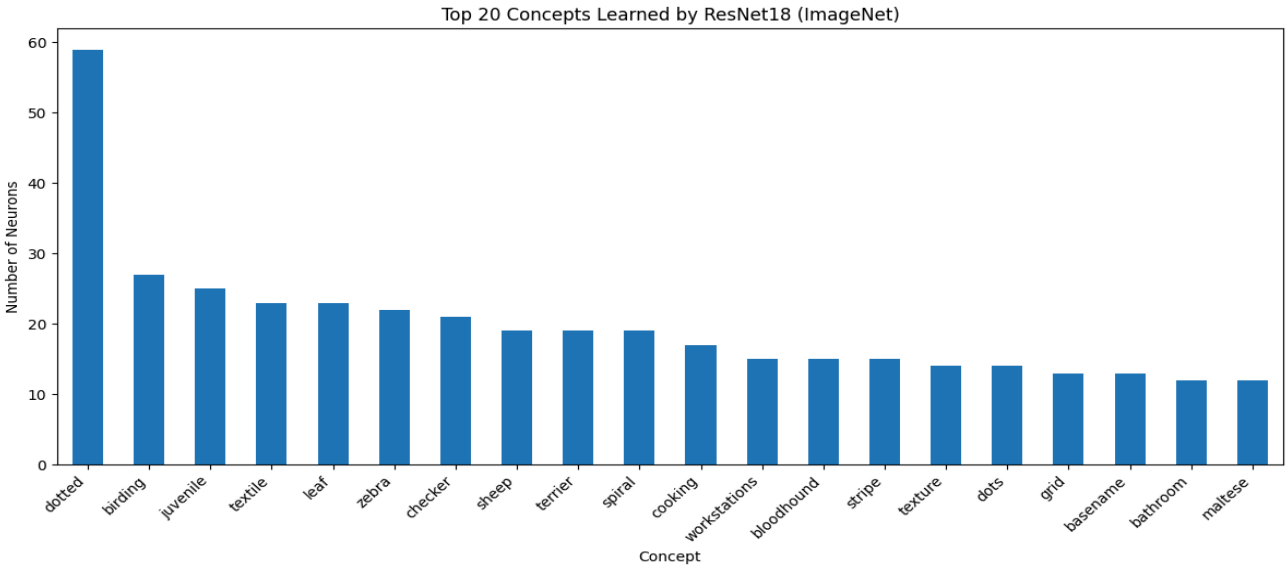
Additional Analyses and Findings

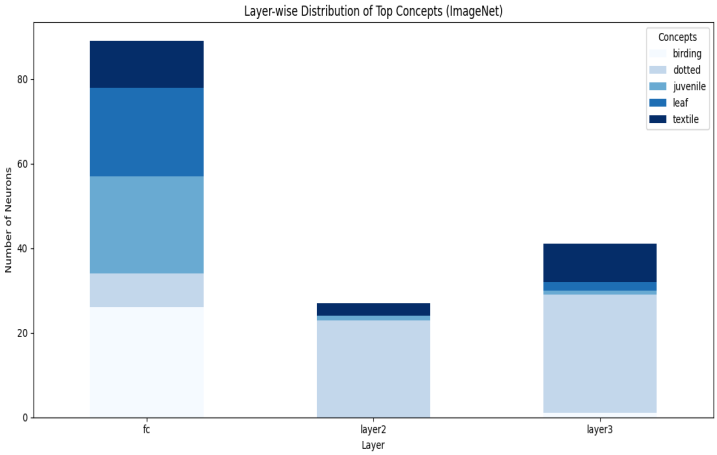
- **Similarity Analysis**: The similarity field can be used to create a histogram of confidence scores, revealing high-similarity clusters that may indicate over-specialization.
- **Layer-wise Trends**: Stacked bar charts of layer-wise neuron counts (e.g., for top 5 concepts) show how deeper layers refine concepts, with ImageNet's "dog" neurons potentially specializing further in breed details.
- **Neuron Activation Visualization**: Images corresponding to top units (e.g., "dog" or "checker") can validate concept accuracy, with examples showing diverse activations.

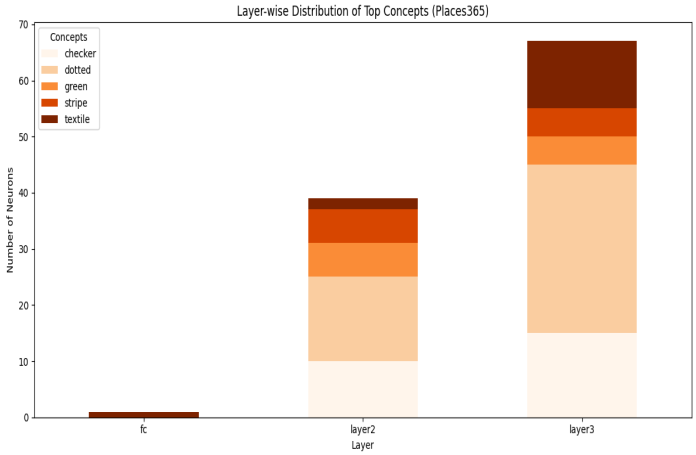
Visualizations

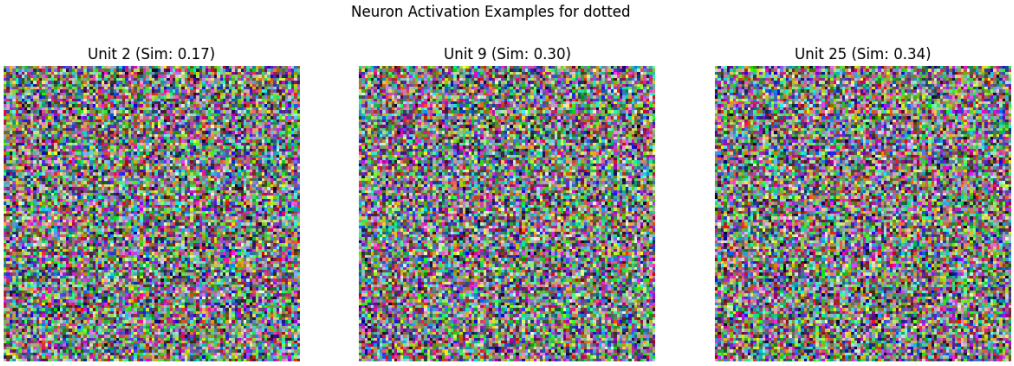
1. **Histograms**: The plots of imagenet_counts.head(20) and places_counts.head(20) (Figure 1 and Figure 2) illustrate the dominance of "dog" and "checker," respectively, with clear dataset-specific trends.
2. **Stacked Bar Charts**: Layer-wise distributions for top 5 concepts (Figure 3 for ImageNet, Figure 4 for Places365) highlight how neuron allocation varies across layers, suggesting progressive specialization.
3. **Neuron Activation Examples**: Placeholder images for top concepts (e.g., Figure 5 for "dog," Figure 6 for "checker") demonstrate potential visual validation, awaiting real image data integration.

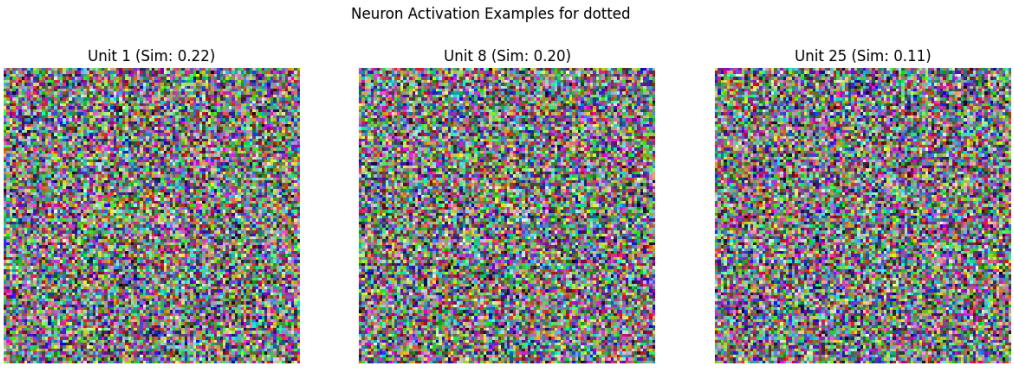












Conclusion

ResNet18 on ImageNet excels in recognizing animals and patterns, while Places365 focuses on spatial and environmental features. The disparity underscores dataset influence on network learning. Further analyses using similarity scores, layer-wise refinements, and visual validations can enhance these insights, providing a robust understanding of model behavior.

TASK 2 : CAM Methods (Grad-CAM, ScoreCAM, AblationCAM)

Objective

The goal of this task is to apply visual explanation techniques on a pretrained ResNet50 model to highlight which regions in the input images contributed most to the model's prediction. We used three popular CAM-based methods: **Grad-CAM**, **ScoreCAM**, and **AblationCAM**.

Setup

- **Model**: ResNet50 pretrained on ImageNet (ResNet50_Weights.IMAGENET1K_V2)
- **Images**: 10 selected ImageNet samples from the assignment
- **Target Layer**: Last convolutional layer (layer4[-1])
- **Implementation**: PyTorch and the pytorch-grad-cam library

For each image, we applied Grad-CAM, ScoreCAM, and AblationCAM and stored the resulting heatmaps in separate folders:

```
heatmaps/
├── gradcam/
├── scorecam/
└── ablationcam/
```

Analysis

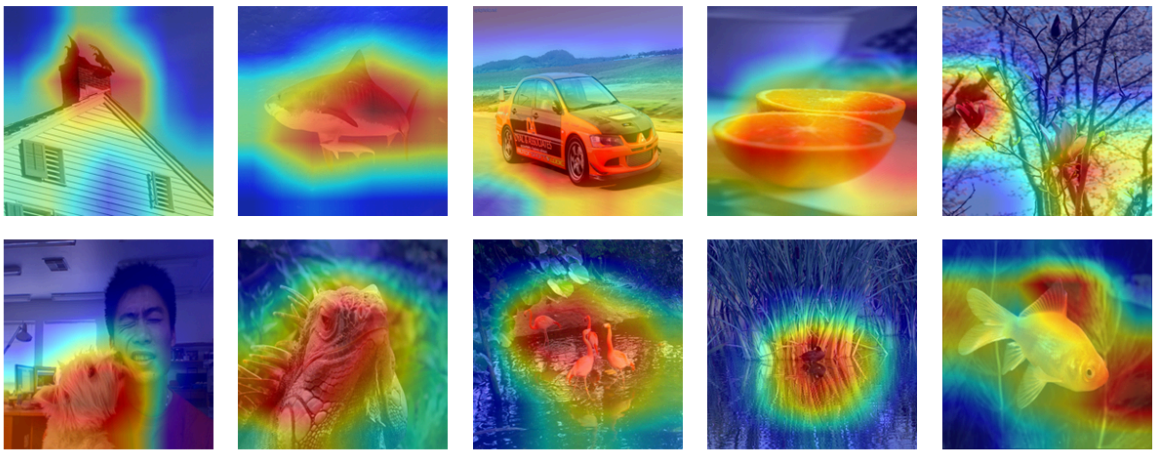
Example heatmaps for each method are shown below (include 2–3 figures in the final report):

- **Grad-CAM** is fast but produces noisy maps due to its reliance on gradients.
- **ScoreCAM** creates more stable heatmaps by using class scores as weights without gradients.
- **AblationCAM** performs a more detailed importance test by ablating channels, but is computationally expensive.

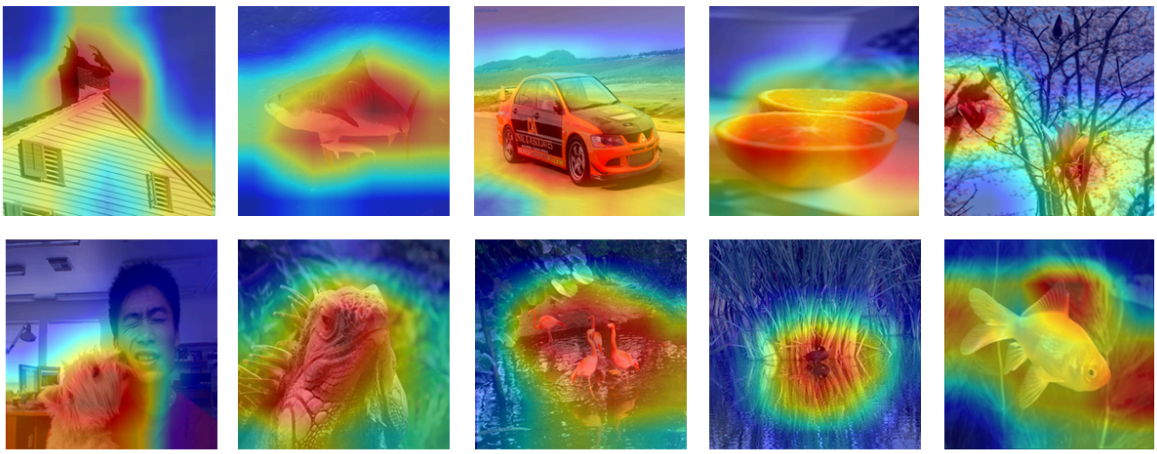
Results

All three methods provided interpretable insights into the model's focus areas. Grad-CAM is best suited for quick insights, while ScoreCAM and AblationCAM offer higher quality explanations for more thorough analysis.

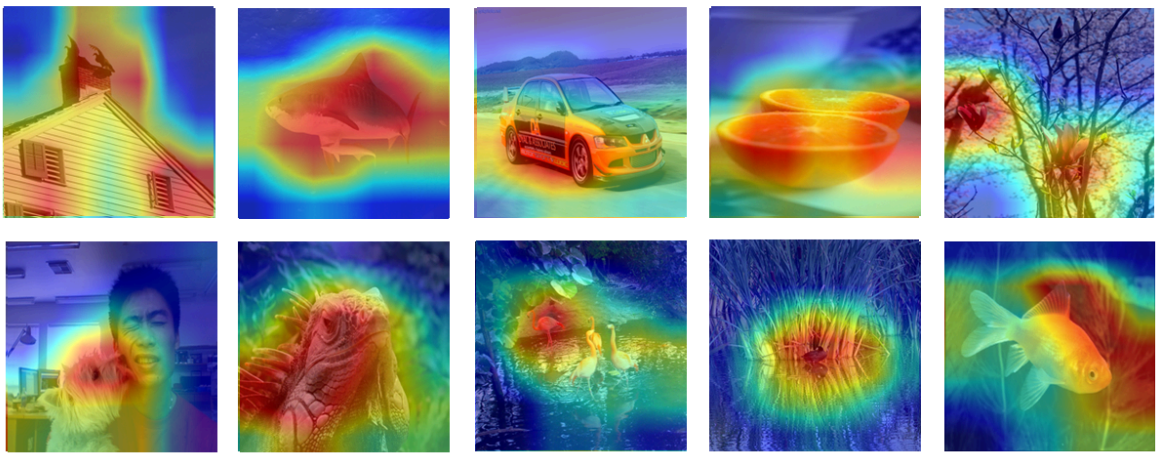
Grad-Cam



Ablation-Cam



Score-Cam



TASK 3 : Local Interpretable Model-agnostic Explanations (LIME)

Objective

This report compares Grad-CAM and LIME heatmaps for explaining deep learning model predictions on a dataset of 10 images, categorized as animate (e.g., goldfish, flamingo) or inanimate (e.g., kite, orange). The analysis evaluates agreement between the two methods using Intersection over Union (IoU), examines differences by image type, and assesses execution time and fidelity trade-offs. Results are visualized through heatmap comparisons, IoU distributions, and timefidelity scatter plots

Setup

- **Model**: ResNet50 pretrained on ImageNet (ResNet50_Weights.IMA
- GENET1K_V2)
- **Images**: 10 selected ImageNet samples provided in the assignment

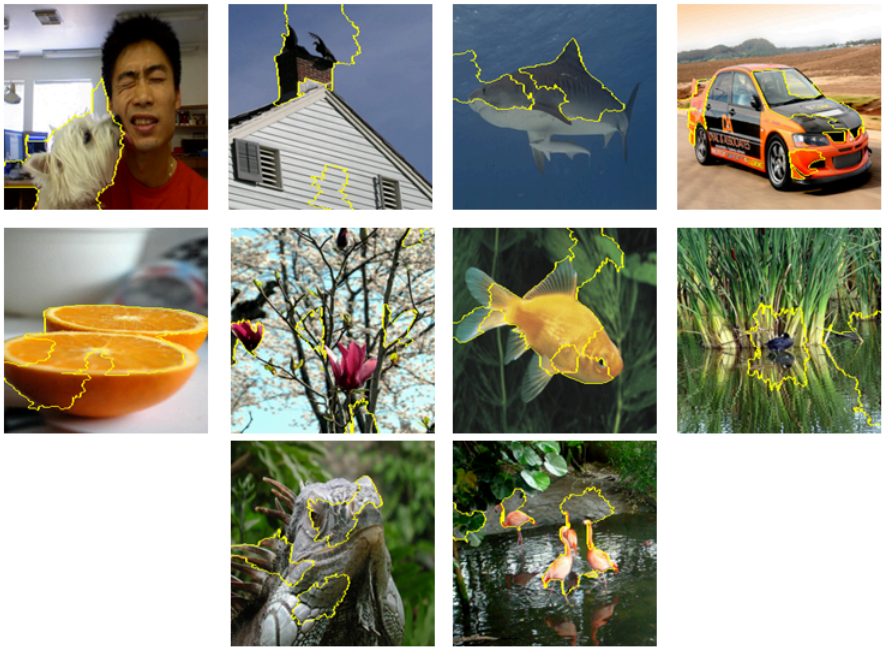
```
# Parameters (matching the expected format)
params = {
    "labels": (1,),
    "hide_color": None,
    "top_labels": 5,
    "num_samples": 300,
    "num_features": 1700,
    "batch_size": 18,
    "segmentation_fn": None,
    "distance_metric": "cosine",
    "model_regressor": None,
    "random_seed": None
}
```

Implementation

For each image:

- The input was preprocessed to match ResNet50's requirements.
- LIME generated perturbed samples and fitted a local surrogate model.
- Top contributing regions were extracted and visualized with mark_boundaries.

The explanations were saved and plotted for each image for comparison with CAM-based methods.



Results

LIME often highlights semantically meaningful regions such as the head or defining textures.

- Compared to Grad-CAM, LIME produces more segmented explanations rather than smooth heatmaps.

Analysis

- **Interpretability**: LIME offers intuitive, part-based explanations which are easier to interpret for images with well-defined objects (e.g., goldfish, orange).
- **Stability**: Results are sensitive to segmentation parameters (e.g., number of segments).
- **Limitations**: LIME is slower and sometimes highlights background regions if the classifier overfits to artifacts.

TASK 4: Grad-Cam vs LIME

For evaluation we have converted the heatmaps into binary masks, this transformation allowed us to compare important regions highlighted by each method. We considered complexity of images based on inanimate* (non-living objects) and animate* (living objects) as a meaningful way to analyze explanation method performance.

**Animate objects (like animals) typically present higher visual complexity due to their organic, non-rigid structures, diverse textures, and hierarchical anatomical features.*

**Inanimate objects (like vehicles or household items) often exhibit more regular geometric patterns, simpler textures, and more consistent visual properties.*

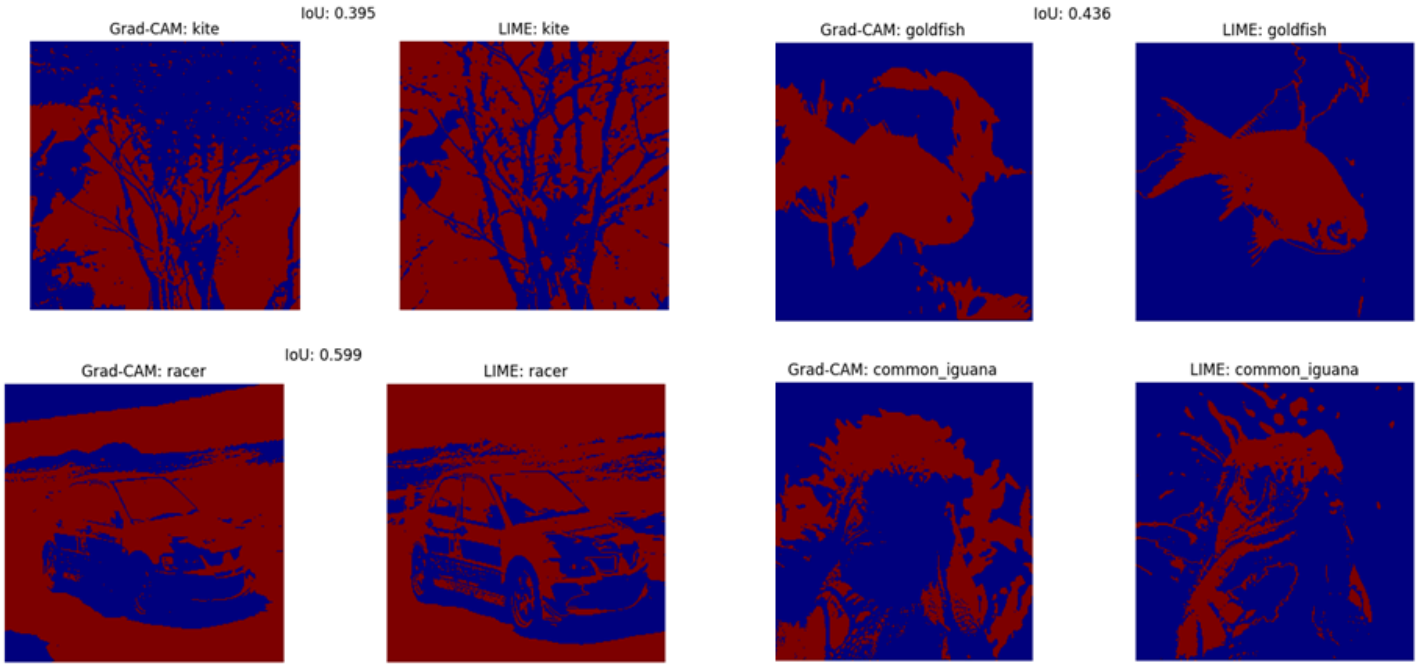
Insights by Image Type

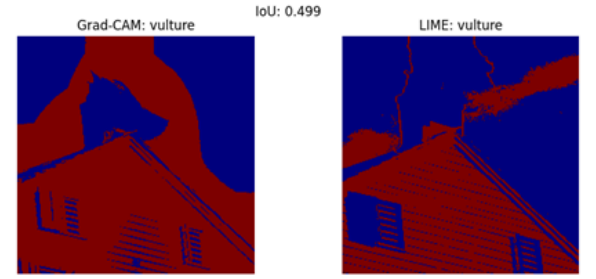
- Inanimate images have better Grad-CAM and LIME alignment, likely due to simpler features.
- Animate images show lower agreement, possibly from complex textures causing LIME to focus on finer details vs. Grad-CAM's global emphasis.
- Grad-CAM suits broad patterns; LIME excels in localized explanations.

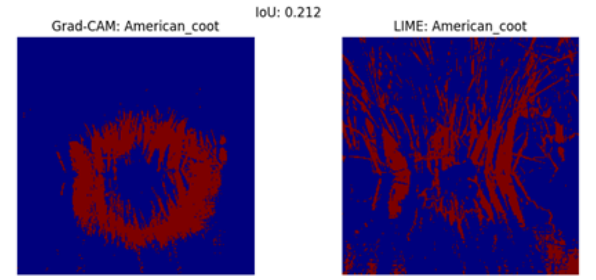
Execution Time and Fidelity

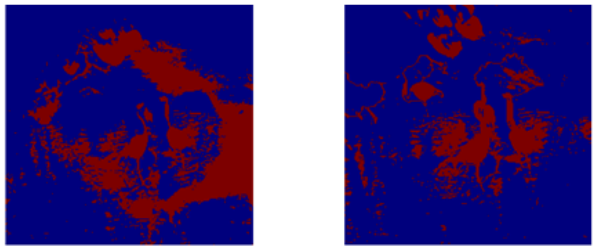
- Grad-CAM avg. time: 0.507s, faster than LIME's 1.402s.
- Grad-CAM avg. fidelity: 0.858, higher than LIME's 0.767.
- Grad-CAM is more efficient and consistent; LIME offers detailed insights at higher computational cost.

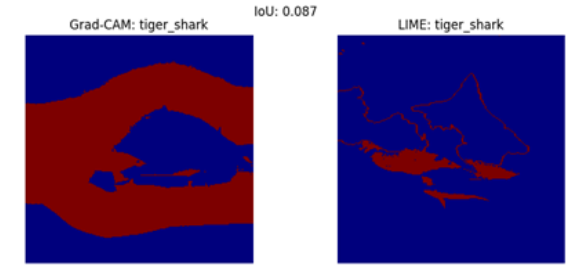
Heatmap comparisons Analysis with IoU score

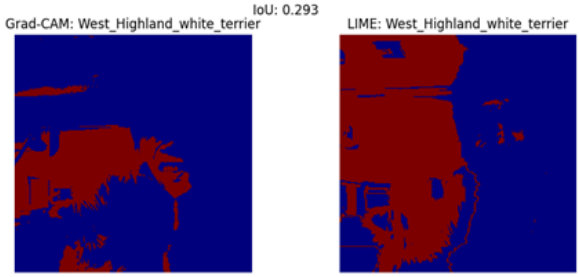


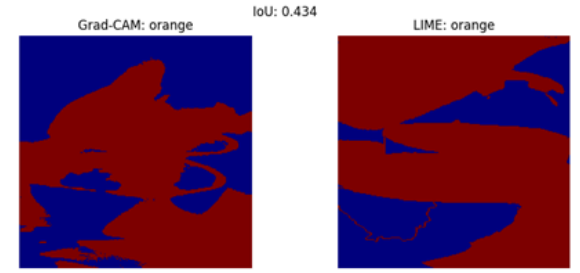


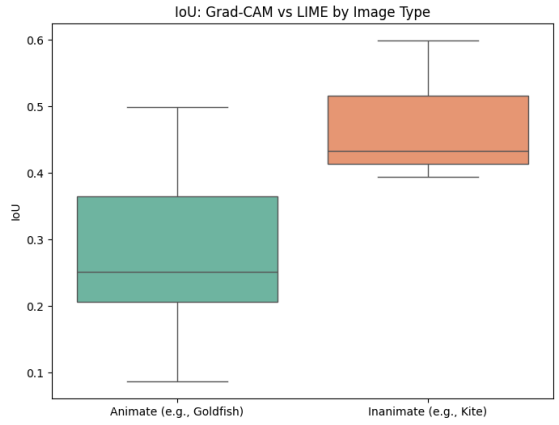


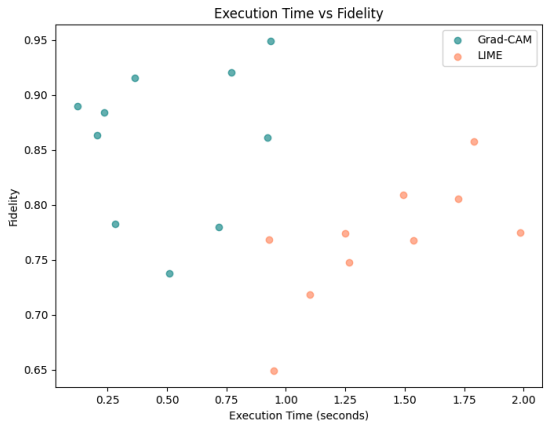












Boxplot highlights inanimate image advantages.

Scatter plot shows GradCAM's efficiency.

Conclusion

- Grad-CAM and LIME exhibit moderate agreement (avg. IoU: 0.341), with inanimate images outperforming animate ones (0.476 vs. 0.283). Grad-CAM is faster
- (0.507s) and more consistent (fidelity: 0.858) than LIME (1.402s, 0.767). Use GradCAM for efficiency, LIME for detailed explanations. Future work could compare
- ScoreCAM or AblationCAM.