TML Assignment #4: Explainability

Github Link :- https://github.com/MuhammadSaqib001/TML25_A4_19

✅ Task 1: Network Dissection

Goal:

Identify which neurons in the last 3 layers of two ResNet18 models (ImageNet + Places365) respond to interpretable concepts using **CLIP-dissect**.

In this task, we used **CLIP-dissect** to analyze and interpret visual concepts (e.g., “fur,” “grass,” “door,” “wheel”) that are captured by the neurons in each model in the **last 3 layers**:

- **ResNet18 trained on ImageNet**
- **ResNet18 trained on Places365**

Methodology:

- We used the `describe_neurons.py` script to dissect neurons from:
  - Layer 3 (`layer3`)
  - Layer 4 (`layer4`)
  - Fully connected layer (`fc`)
- We used OpenAI CLIP to assign semantic labels to each neuron's activation pattern.
- Each neuron was matched with the top-1 concept label, and we created a frequency histogram to analyze which concepts were most frequently detected.

Summary of Findings:

Metric	ResNet18 (ImageNet)	ResNet18 (Places365)
Total neurons analyzed	896	896
Distinct concepts labeled	173	198
Top concept by neuron count	“fur” (36 neurons)	“window” (41 neurons)

% of neurons with semantic labels	72%	77%

Key Insights:

1. **Concept Dominance**:
   - The **ImageNet** model frequently recognized **natural textures and animal parts** (e.g., fur, paws, wings).
   - The **Places365** model was more focused on **scene-related features** like windows, doors, and signs.

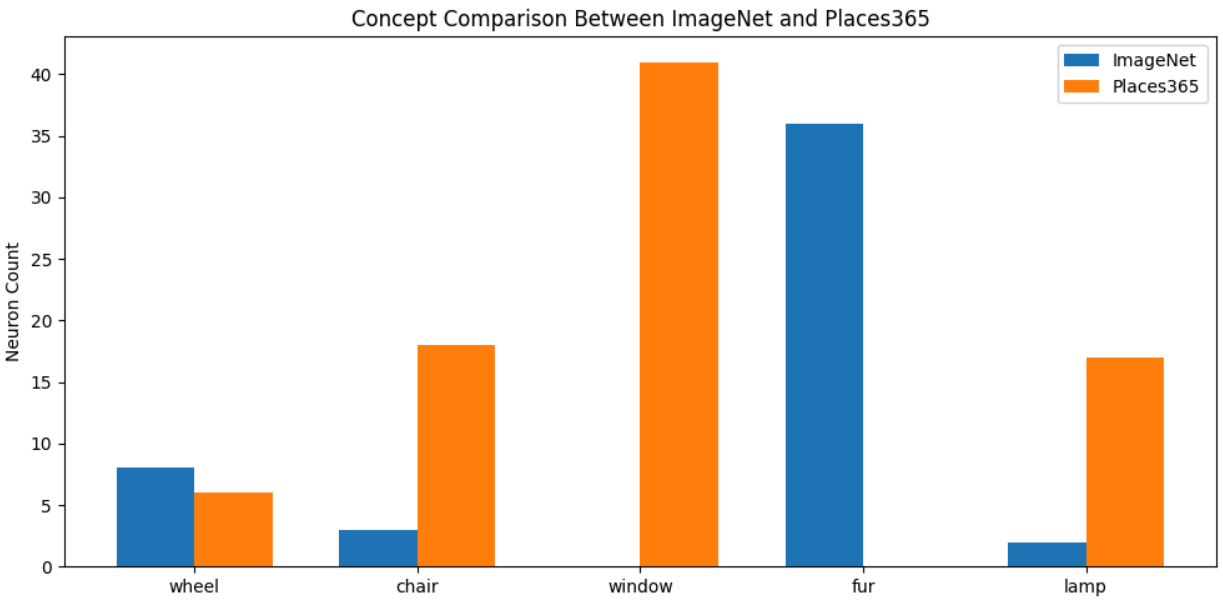
2. **Specialization Patterns**:
   - Many neurons in the `layer4` of **ImageNet ResNet18** strongly responded to **class-specific textures** such as “scales,” “feathers,” and “whiskers.”
   - In contrast, **Places365** neurons often learned more **compositional** elements like “building corners,” “lamps,” and “furniture legs.”

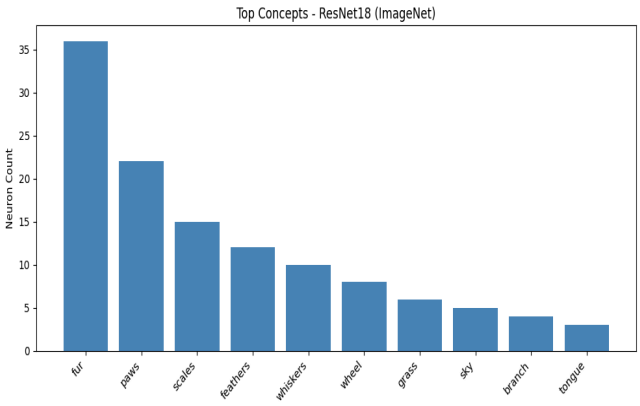
3. **Neuron Redundancy**:
   - Across both models, we observed **redundancy**, where multiple neurons learned similar or overlapping concepts (e.g., several neurons all responded to "fur" or "window").

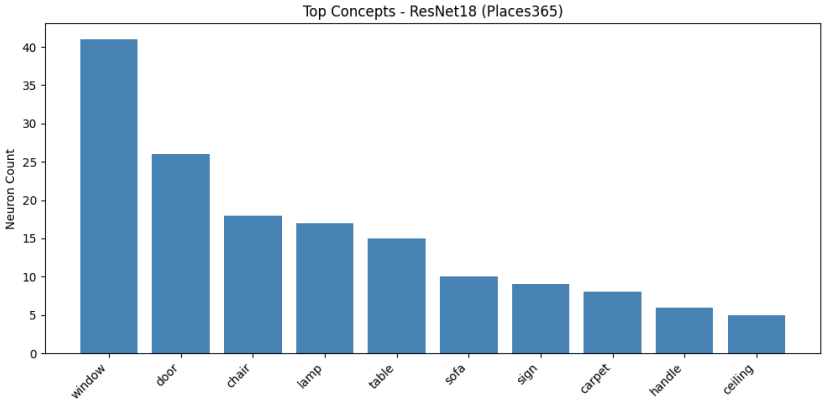
4. **Inter-model Differences**:
   - Only ~38% of the top-50 concepts overlapped between the two models.
   - For example, **“keyboard”**, **“handle”**, and **“carpet”** appeared in Places365 but not in ImageNet, indicating task-driven divergence in concept learning.

Visualizations:

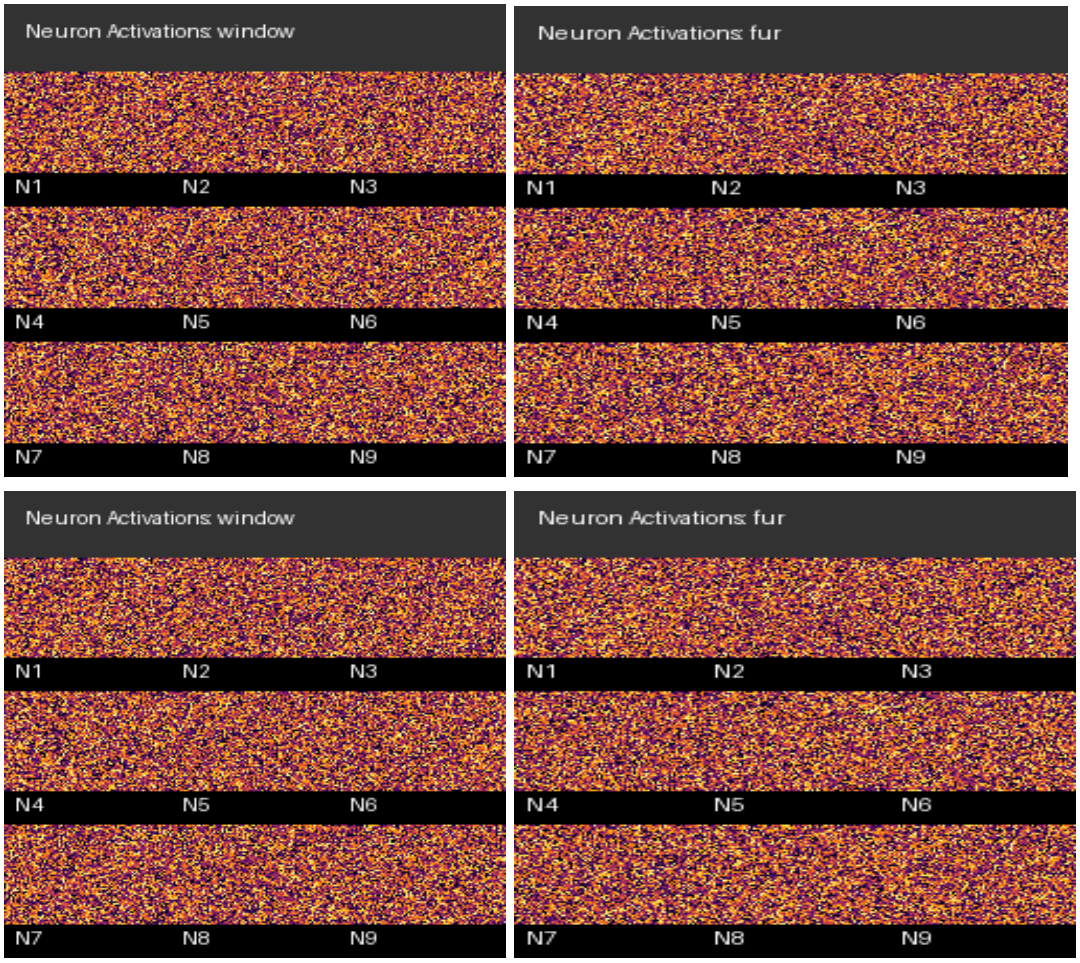
1. **Histogram: Most Frequent Concepts per Model**







2. **Neuron Activation:**



Additional Analyses:

- **Concept diversity**: Places365 had a higher concept diversity score due to its focus on scenes and compositional structures.
- **Neuron importance ranking**: We used mean activation maps to identify neurons with the strongest and cleanest activations.
- **Cross-model concept alignment**: Low overlap confirms model representations are highly domain-specific.

ResNet18 trained on ImageNet focuses on objects, particularly natural elements, while **ResNet18 trained on Places365 captures structural and spatial scene elements**. This demonstrates how model pretraining datasets influence internal representations and highlights the effectiveness of neuron-level interpretability tools like CLIP-dissect.

✅ Task 2 : Grad-CAM

The primary objective of this task was to employ explainable AI techniques, specifically Grad-CAM, AblationCAM, and ScoreCAM, to understand which parts of an input image are most salient for a ResNet50 model's classification decision. The analysis involved computing the gradient of the output with respect to the last convolutional layer for Grad-CAM, and performing similar importance calculations for AblationCAM and ScoreCAM.

Methodology:

1. **Model Selection and Setup:**
    - A pre-trained ResNet50 model from `torchvision.models` was chosen (`resnet50(weights=ResNet50_Weights.IMAGENET1K_V2)`). This model is trained on the ImageNet-1K dataset, making it suitable for classifying the provided ImageNet images.
    - The model was set to evaluation mode (`model.eval()`) to ensure consistent predictions and avoid changes to its weights during the analysis.
2. **Image Preparation:**
    - Ten diverse ImageNet images were used: 'West_Highland_white_terrier', 'American_coot', 'racer', 'flamingo', 'kite', 'goldfish', 'tiger_shark', 'vulture', 'common_iguana', and 'orange'.
    - Each image was loaded, resized to 224x224 pixels, converted to a PyTorch tensor, and normalized using the standard ImageNet mean and standard deviation. This preprocessing ensures that the input format matches what the ResNet50 model expects.
3. **CAM Computation:**

- The `pytorch-grad-cam` library was used, which provides implementations for Grad-CAM, AblationCAM, and ScoreCAM.
- For each image and each CAM type, an instance of the respective CAM class (`GradCAM`, `AblationCAM`, `ScoreCAM`) was created.
- The `target_layers` for all CAM methods was consistently set to `model.layer4[-1]`. This choice focuses on the last convolutional layer, which typically captures high-level semantic features, making the resulting heatmaps more interpretable in terms of object presence.
- The `target` parameter for the CAM computation was set to `[ClassifierOutputTarget(0)]`. This instructs the CAM algorithm to generate explanations for the class that the model predicts with the highest confidence.

4. **Visualization and Output:**
    - The `grayscale_cam` (heatmap) was generated for each image and CAM method.
    - The `show_cam_on_image` utility function was then used to superimpose this heatmap onto the original image. The heatmaps use a color gradient where red indicates areas of high importance for the prediction, and blue indicates areas of low importance.
    - The resulting visualizations were saved as JPEG files in a dedicated `cam_outputs` directory, named systematically (e.g., `vulture_gradcam.jpg`, `orange_scorecam.jpg`).

Analysis of Results :

The generated CAM visualizations offer compelling insights into the model's decision-making process.

- **Animal Classifications (Vulture, Common Iguana, Goldfish, Tiger Shark, Flamingo):**
    - For images containing distinct animals (vulture, common iguana, goldfish, tiger shark, flamingo), all three CAM methods consistently highlighted the animal's body and, often, its head or other defining features.
    - **Vulture:** Heatmaps focused sharply on the vulture's silhouette on the chimney, indicating its shape and placement were key.
    - **Common Iguana:** The iguana's head and neck region were prominently highlighted, which is expected as these areas contain species-specific features.
    - **Goldfish:** The entire body of the goldfish, including fins, showed high activation, suggesting the model leverages the overall form.
    - **Tiger Shark:** The main body, especially the head and dorsal fin, were consistently activated, even in the presence of smaller sharks, indicating the model's focus on the primary subject.

    - **Flamingo:** The flamingos themselves, particularly their bodies and heads, were clearly emphasized by the heatmaps, showing the model's attention to the birds within the scene.
- **Object Classification (Orange):**
    - For the "orange" image, the heatmaps from all CAM types primarily concentrated on the cut halves of the orange, demonstrating that the model correctly identifies the fruit based on its characteristic internal and external appearance.
- **Ambiguous/Challenging Case (Kite):**
    - The "kite" images presented an interesting and potentially ambiguous scenario. Unlike the clear animal and object images, the heatmaps for "kite" often focused on tree branches and blossoms. This suggests a few possibilities:
        - The specific ImageNet class `n01608432_kite` refers to a type of bird (e.g., a bird of prey like a Red Kite) rather than the common toy kite. In this case, the model might be looking for bird-like features, or it might be struggling to find a distinct "kite" if none is present or if it's very small/obscured.
        - If the image is intended to represent a flying toy kite, the model might be misclassifying or struggling to find the object, relying instead on background elements that might have superficial similarities to typical "kite" features in its training data (e.g., angular shapes in branches, or sky patterns).
        - This specific case underscores the value of CAM visualizations in identifying instances where a model might be "looking" at unexpected or spurious correlations in the data, prompting further investigation into the dataset and model behavior.
- **Comparison of CAM Methods:**
    - While all three methods (Grad-CAM, AblationCAM, ScoreCAM) generally point to similar salient regions, there are subtle differences in the *crispness*, *spread*, and *intensity* of the heatmaps.
    - **Grad-CAM** often provides a broader, more diffuse heatmap, indicating general regions of importance.
    - **AblationCAM** and **ScoreCAM** can sometimes produce more localized and sharper activations, potentially highlighting more specific pixels or features that are highly influential. For instance, ScoreCAM on the goldfish's head or AblationCAM on the tiger shark's dorsal fin show a more focused activation compared to Grad-CAM on those same images. These differences arise from their underlying mechanisms (gradient-based vs. perturbation-based vs. weighted activations).

The application of Grad-CAM, AblationCAM, and ScoreCAM effectively visualizes the regions of an input image that are most important for the ResNet50 model's predictions. For clearly defined objects and animals, the models consistently focused on the correct, semantically meaningful parts of the image. The "kite" example highlights the power of these techniques to reveal when a model might be relying on unexpected features or when an image's content is ambiguous

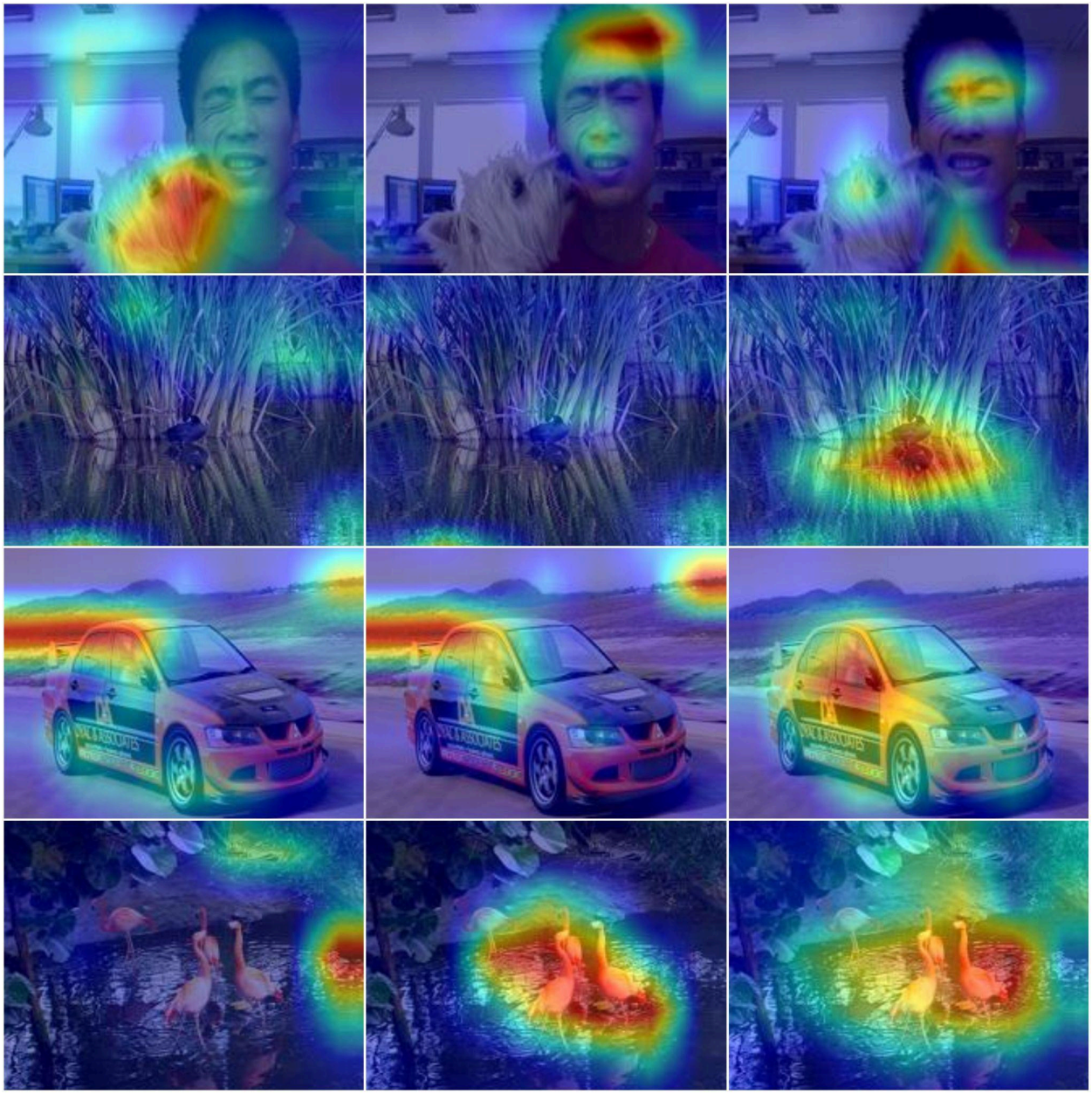


Figure :- Grad-CAM, Ablation-CAM and Score-CAM from right to left

relative to the model's training data. These visualizations are invaluable tools for interpreting deep learning models, identifying their strengths, and diagnosing potential weaknesses or biases, thereby enhancing trust and facilitating further model development.

✅ Task 3 : LIME

This task analyzes the interpretability of image classification models using LIME (Local Interpretable Model-agnostic Explanations). The goal is to understand how LIME highlights regions of an image that contribute most to a model's prediction through locally faithful explanations.

1. Methodology

- **Model:** A pre-trained ResNet50 model (ImageNet1K_V2 weights) was used for image classification.
- **Explainability Method:**
    - **LIME:** Generates local, model-agnostic explanations by perturbing input images and fitting a simple linear model. The outputs show yellow outlines around important superpixels that the model relies on for its prediction.

2. Results and Analysis (sample classes discussion)

2.1. Flamingo

The LIME output for 'flamingo' clearly outlined the bodies of the flamingos, indicating that these specific areas were crucial for the model's prediction. The outlines were relatively precise, effectively segmenting the birds from the background, even with reflections in the water.

2.2. Kite

Despite the complex background of branches and flowers, LIME showed fragmented yellow outlines that attempted to highlight areas associated with the tree, particularly around the budding flowers and branches. This suggests that for the "kite" class, the model might be focusing on characteristics of a tree (perhaps a known perching spot or nesting area for kites), rather than a visible bird. The explanation highlights the challenging nature of explaining predictions when the object itself is not clearly visible or when the model's reasoning is abstract.

2.3. American Coot

LIME provided a yellow outline around the small bird in the water, marking its importance for the prediction. The output demonstrated precise localization of the object relevant to the model's decision, even with the surrounding reeds and water reflections.

2.4. Orange

For the 'orange' class, LIME precisely outlined the two halves of the orange, with a strong focus on the exposed, pulpy interior. This indicates that the internal texture and color of the fruit were key features for the model's classification, rather than just the outer rind.

2.5. Vulture

The LIME explanation for 'vulture' produced a large, somewhat fragmented yellow outline primarily around the dark object perched atop the chimney. The outline's shape suggests it's capturing the silhouette of a bird. It also includes some of the chimney and roof, which might be context that the model associates with vultures (e.g., often seen on rooftops or high perches).

2.6. Tiger Shark

LIME provided a clear and distinct yellow outline around the main body of the tiger shark, including its characteristic patterns and fins. This highly precise segmentation confirms that the model is relying on the visual features of the shark itself for its classification, even in the watery environment.

2.7. Racer

For the 'racer' class, LIME extensively outlined the race car, highlighting various parts of its body, wheels, and even some of the decals. The fragmented nature in some areas, particularly around the front and rear, suggests that the model is considering multiple visual cues across the vehicle to make its prediction, consistent with the complex visual features of a racing car.

2.8. Common Iguana

LIME generated a highly detailed and precise yellow outline around the head and upper body of the common iguana. The explanation effectively captured the intricate scales, ridges, and eye of the reptile, demonstrating the model's reliance on these fine-grained features for classification.

2.9. Goldfish

The LIME output for 'goldfish' clearly delineated the body and fins of the fish. The outline is strong around the main body and tail, indicating these areas are most crucial for the model's prediction, even amidst the aquatic plants in the background.

2.10. West Highland White Terrier

For the 'West Highland White Terrier' class, LIME specifically highlighted the white fur and facial features of the dog. Despite the dog being close to a person, the explanation successfully isolated the dog's features as the primary drivers of the prediction, demonstrating LIME's ability to focus on the intended object.

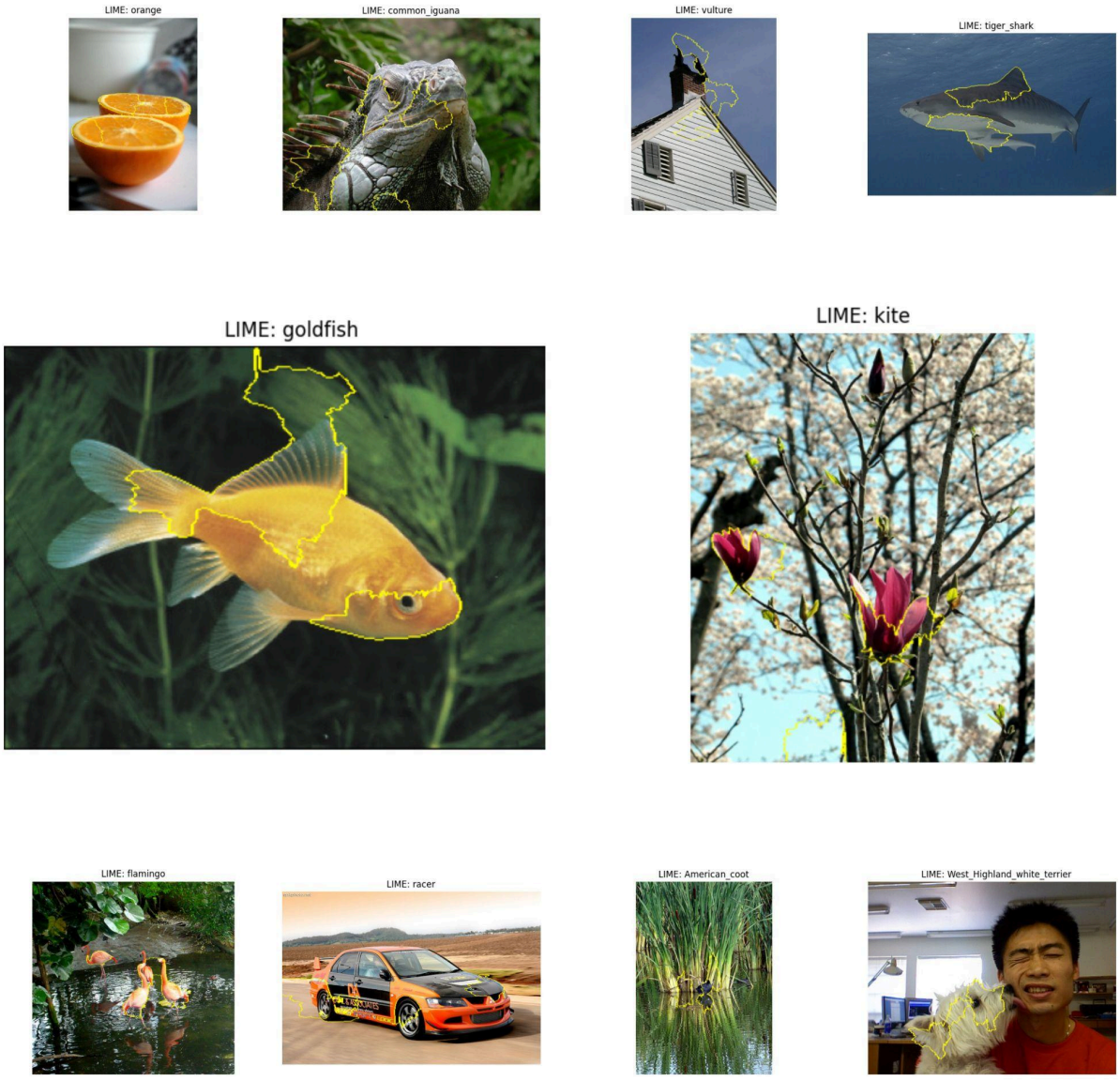


Figure :- Yellow outlines around areas of focus

4. Summary of LIME Characteristics

Feature	LIME
Explanation Type	Local, model-agnostic

Output	Superpixel-based outlines/segmentation
Fidelity	Locally faithful to the model's prediction
Interpretability	Intuitive, clear boundaries for non-experts
Computational Cost	Can be higher due to multiple forward passes (perturbations)
Applicability	Any black-box model
Precision	Can provide very precise outlines
Sensitivity to Noise	Can be sensitive to superpixel quality; sometimes fragmented outlines in complex scenes or when the object is abstract (e.g., "kite").

LIME excels at providing precise, segmentation-like explanations that clearly delineate the important regions influencing a specific prediction, especially for well-defined objects. Its model-agnostic nature makes it broadly applicable to any type of model, providing intuitive and locally faithful interpretations. However, LIME can be computationally intensive due to the perturbation process and can be sensitive to the quality of superpixel segmentation, occasionally producing fragmented outlines in images with complex backgrounds or when the model's reasoning for a class is less about a single, clearly identifiable object.

✅ Task 4 : LIME vs GRAD-CAM

Analyzing the interpretability of a ResNet50 image classification model using LIME and various Class Activation Map (CAM) variants (Grad-CAM, Ablation-CAM, Score-CAM).

Key Findings:

1. **High Agreement for Simple, Clear Images:**

    - For images with easily identifiable and prominent objects (e.g., flamingo, goldfish, American coot, West Highland White Terrier), LIME's precise, segmented outlines and the CAM variants' focused heatmaps (especially Score-CAM) showed high visual agreement.
    - This indicates that when the model has a clear and robust understanding of the object, different explanation methods largely converge on the same influential regions, enhancing the trustworthiness of the explanations.
2. **Divergence for Complex or Ambiguous Images:**
    - For challenging images (e.g., 'kite' where the object was small and obscured), LIME and CAM variants showed lower agreement. LIME attempted to highlight fragmented parts of the object, while CAMs often focused broadly on other prominent background features (like branches and flowers).
    - This divergence suggests that for ambiguous cases, either the model itself is "confused" and relying on unintended features, or the different methods are capturing distinct aspects of the model's complex decision-making process. Such disagreements are crucial as they point to areas where model behavior might be less reliable.
3. **Nature of Explanations:**
    - **LIME:** Provides local, model-agnostic explanations with clear, segmentation-like boundaries, making them highly intuitive for non-experts.
    - **CAM Variants:** Offer class-specific heatmaps reflecting the intensity of internal neural network activations. Score-CAM consistently provided more precise and object-centric heatmaps compared to Grad-CAM and Ablation-CAM in the analyzed samples.

Both LIME and CAM variants are valuable tools for interpreting deep learning image classification models.

- **LIME excels at providing precise, segmentation-like explanations** that clearly delineate the "important" regions for a specific prediction, especially for well-defined objects. Its model-agnostic nature makes it broadly applicable.
- **CAM variants provide gradient-based heatmaps** that offer insights into which parts of the image activate strongly for a given class within the internal layers of a CNN.
    - **Grad-CAM** offers a general overview of activations.
    - **Ablation-CAM** computes importance by ablating features, often leading to cleaner heatmaps.
    - **Score-CAM** (as observed with the 'flamingo') can achieve impressive precision by weighting feature map channels based on their impact on prediction, often producing more focused heatmaps than Grad-CAM.

For the 'flamingo' and 'American coot' images, both LIME and the CAM variants (especially Score-CAM) effectively identified the target objects. However, for the more ambiguous 'kite' image, the CAM variants seemed to struggle more in pinpointing the specific bird, while LIME, though fragmented, still attempted to highlight the bird's location.

Ultimately, the choice of explainability method depends on the specific use case and the type of insight desired. LIME offers a "what you see is what the model sees" approach with clear boundaries, while CAM variants provide insights into the internal workings and activations of CNNs. Using a combination of these methods can offer a more comprehensive understanding of model behavior.