Explainable AI- Understanding and Interpreting Computer Vision Models

Team 10
Lahari | Divija | Hima Chandh | Anurag

XAI in Computer Vision

- Sets up the techniques and methods designed to make decision making process of AI Models especially DL models
- XAI tries to make the models more Transparent and interpret
- It is often considered as a black box due to its high complex working mechanism making it difficult to understand or trust the predictions.

Importance.. Why XAI??

- Trust and Transparency: XAI enables users to comprehend and trust AI decisions, which is crucial in high-stakes domains such as healthcare, finance, and autonomous systems. In these fields, unexplained errors can have serious consequences
- XAI helps identify hidden biases and model misinterpretations, thus improving robustness and fairness.
- and many more..

XAI in Computer Vision

Saliency-Based Methods (Gradient-Based XAI)

Saliency Maps: These highlight the most influential regions of an image for a model's prediction by computing gradients with respect to input pixels

- Deconvolution Networks & Guided Backpropagation: These techniques backpropagate activations to reconstruct relevant features, with guided backpropagation suppressing irrelevant gradients for clearer visualizations

Class Activation Mapping (CAM) and Variants

- CAM: Generates heatmaps that highlight image regions most responsible for a specific class prediction by weighting feature maps using global average pooling. It is mainly applicable to CNNs without fully connected layers

Perturbation-Based Methods

- Principle: These methods explain model decisions by systematically modifying (perturbing) input data and observing how predictions change. The importance of each input region is inferred from the impact of its perturbation on the model's output .
- Key Techniques:
  - LIME: Locally approximates the complex model with a simpler, interpretable model by perturbing the input and analyzing prediction shifts .
  - SHAP: Uses Shapley values to fairly assign feature importance, ensuring consistent exfplanations

Problem Statement

Core Challenge

Designing AI systems for computer vision tasks (e.g., object detection, semantic segmentation, classification) that are both *accurate* and *interpretable*, enabling users to:

1. Understand how input features (e.g., pixels, regions) influence model predictions.
2. Verify the logical consistency of model decisions.
3. Identify biases, errors, or vulnerabilities in model behavior.

Key Definitions

1. Interpretability: Static analysis of model parameters or equations to infer decision logic (e.g., visualizing filters in a CNN)
2. Explainability: Dynamic, post-hoc attribution of predictions to input features (e.g., heatmaps highlighting influential pixels)
3. Faithfulness: The degree to which explanations reflect the model’s true reasoning process

Challenges

- Trade-off Between Accuracy and Interpretability: More interpretable models can sometimes sacrifice predictive accuracy, and vice versa.
- Conflicting Explanations: Different XAI techniques may yield divergent explanations for the same prediction, highlighting the need for standardization and rigorous evaluation metrics.
- Scalability and Real-World Deployment: Many XAI methods are computationally expensive and may not scale easily to large, real-world datasets or complex models.
- Alignment with Human Reasoning: Ensuring that explanations are not only technically accurate but also understandable and meaningful to domain experts and end users is a persistent challenge.

Assumptions

1. Model Transparency:
   - Gradient-based methods (e.g., Grad-CAM) assume access to model internals ($\partial y/\partial A$).
   - Perturbation-based methods (e.g., LIME, BODEM) assume black-box access only
2. Locality:
   - Explanations are valid within a local input neighborhood (e.g., LIME's linear approximation holds near x).
3. Human-AI Alignment:
   - Explanations must align with human intuition

References

[1]Guoyang Liu, Jindi Zhang , Antoni B. Chan,Janet H. Hsiao “Human Attention-Guided Explainable Artificial Intelligence for Computer Vision Models” arXiv:2305.03601v1 [cs.CV] 5 May 2023

[2]Milad Moradi, Ke Yan,David Colwell,Matthias Samwald,Rhona Asgari “Model-agnostic explainable artificial intelligence for object detection in image data”Tricentis GmbH, Leonard-Bernstein-Straße 10, 1220 Vienna, Austria

[3]Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, Vineeth N Balasubramanian “Grad-CAM++: Improved Visual Explanations for Deep Convolutional Networks”

[4]Alphonse Inbaraj Xavier , Charlyn Villavicencio , Julio Jerison Macrohon , Jyh-Horng Jeng and Jer-Guang Hsieh “Object Detection via Gradient-Based Mask R-CNN Using Machine Learning Algorithms” Machines 2022, 10, 340

[5]Mohammed Bany Muhammad Mohammed Yeasi “ Eigen-CAM: Class Activation Map using Principal Components “

Literature Review

Model-agnostic Explainable AI for Object Detection

- Propose BODEM (Black-box Object Detection Explanation by Masking), a model-agnostic XAI method for explaining object detection models using hierarchical random masking.

BODEM operates in three stages:

- **Hierarchical Mask Generation** - divides the image into blocks (starting at KxK size and decreasing in successive levels), creating masks to identify and refine salient regions.
- **Model Inquiry** - probes the detection model with masked images to observe changes in bounding box predictions;
- **Saliency Estimation** - computes importance scores using Intersection over Union (IoU) between original and perturbed detections, iteratively updating a saliency maps.

Literature Review

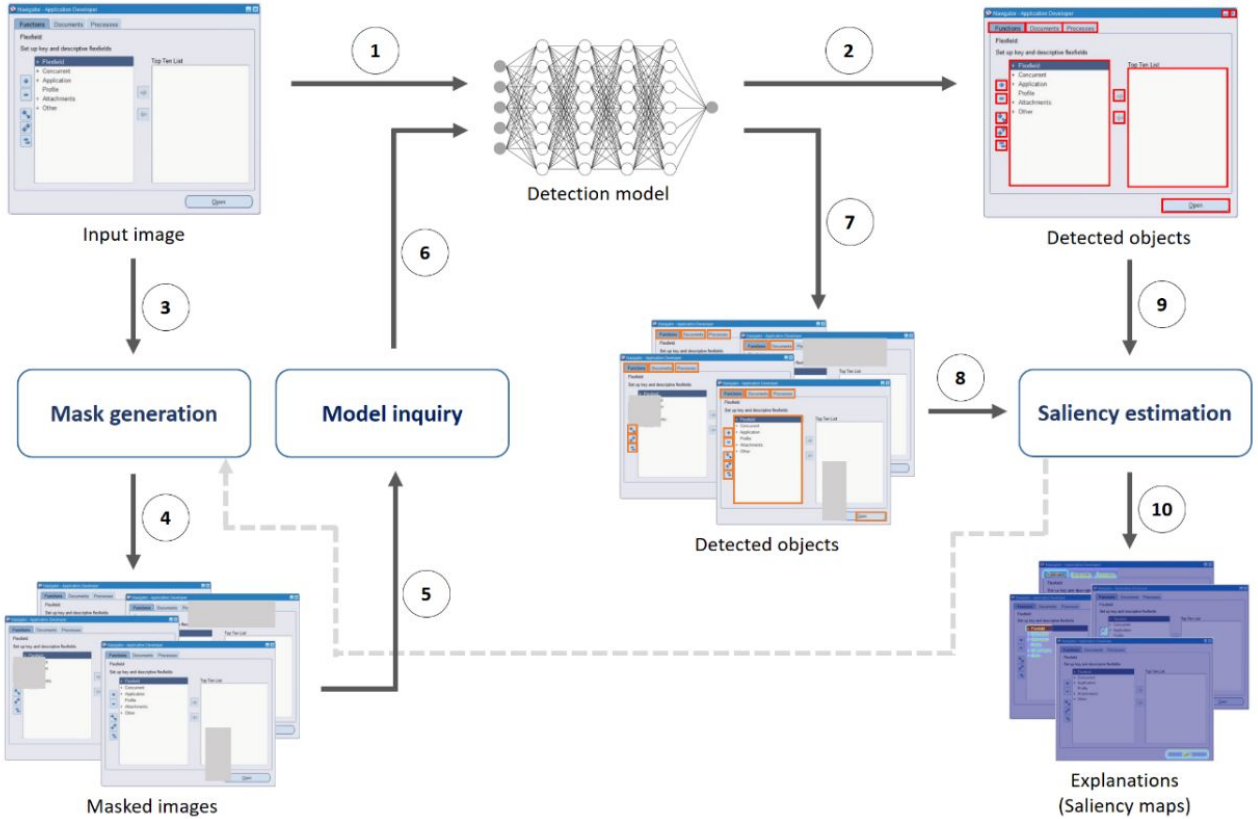
Key Strengths

- **Versatility in Application**: Black-box approach makes it applicable to any object detection model without internal access.
- **Noise Reduction**: Hierarchical masking minimizes irrelevant pixel importance, improving saliency map accuracy and stability.

Areas of Improvement

- **Computational Overhead**: Hierarchical masking and multiple model inquiries are resource-intensive, potentially limiting scalability for real-time or large-scale applications.
- **Missing User Validation**: Lacks qualitative validation of saliency map interpretability by users; planned for future work.

Architecture of BODEM explanation method



EigenCAM Implementation

A. N. Rahman, D. Andriana and C. Machbub, "**Comparison between Grad-CAM and EigenCAM on YOLOv5 detection model**," 2022 International Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 2022

EigenCAM:

EigenCAM is an unsupervised and gradient-free visualization technique used to interpret the decision-making process of models. Unlike Grad-CAM, which requires gradient computation, EigenCAM uses only the activation maps from convolution layers and applies Singular Value Decomposition(SVD) to extract important features.

The main intuition behind EigenCAM is:

- Every convolutional layer in a CNN outputs multiple feature maps (activation maps).
- Some of these maps contain more “important” spatial information about the image than others.
- EigenCAM compresses these maps into a single 2D attention heatmap by extracting the dominant direction of variance in the activation space (via first principal component).

Eigen CAM Implementations

- ➔ Choose a convolutional layer in the neural network (usually the last one).
- ➔ Extract the activation maps from that layer for a given input image.
- ➔ Flatten the spatial dimensions of each map (e.g., 7×7 map → 49-dim vector per map).
- ➔ Stack all the flattened vectors into a matrix.
- ➔ Apply SVD to this matrix:
  - ◆ The first principal component (eigenvector with the largest eigenvalue) captures the direction of maximum variance across maps.
- ➔ Project the stacked matrix onto this component to get a heatmap.
- ➔ Upsample and overlay this heatmap on the original image.

Literature Review

Human Attention-Guided Explainable AI (HAG-XAI) for Computer Vision[1]

- Human attention maps, derived from eye-tracking data, have been shown to provide more *faithful* and *plausible* explanations than standard XAI methods, particularly in object detection. Faithfulness measures how well highlighted regions truly influence model decisions, while plausibility assesses how understandable explanations are to humans
- **Key Innovations:**
  - *FullGrad-CAM & FullGrad-CAM++:* Extensions of gradient-based methods to generate object-specific, less noisy explanations for object detection models
  - *HAG-XAI:* A novel framework that uses human attention data to train interpretable, model-specific activation functions and smoothing kernels. This approach maximizes the similarity between AI-generated saliency maps and human attention maps, enhancing both faithfulness and plausibility for object detection .
- **Main Findings:**
  - For object detection, HAG-XAI outperforms existing XAI methods, simultaneously improving faithfulness and plausibility, and generalizes well across datasets (BDD-100K, MS-COCO) .
  - For image classification, HAG-XAI increases plausibility but may reduce faithfulness, reflecting differences between human and model focus.
  - The learned parameters in HAG-XAI are interpretable and reveal model-specific adaptation needs (e.g., larger smoothing for YOLOv5s, grid-pattern smoothing for Faster R-CNN).
  - Human attention can serve as a benchmark and guide for developing more intuitive, trustworthy XAI in safety-critical applications

Evaluation Metrics

Faithfulness Evaluation Metrics

1. Insertion Score

- Purpose: Measures how well saliency maps correlate with model confidence increase when adding salient regions.
- Calculation:
  - AUC (Area Under Curve) of model confidence as salient pixels are incrementally inserted.
  - Higher AUC = Better (Ideal: ~1.0).

2. Deletion Score

- Purpose: Quantifies confidence drop when removing salient regions.
- Calculation:
  - AUC of confidence decay as salient pixels are removed.
  - Lower AUC = Better (Ideal: ~0.0).
- Trade-off: High insertion + low deletion → explanations are both relevant and complete.

Evaluation Metrics

Plausibility Evaluation Metrics

Flatten the Maps before computing

1. Pearson Correlation Coefficient (PCC)

- Purpose: Measures linear correlation between saliency scores and human annotations.
- Formula:

$$PCC(u_1, u_2) = \frac{\bar{u_1}^T \bar{u_2}}{\sqrt{\bar{u_1}^T \bar{u_1} \cdot \bar{u_2}^T \bar{u_2}}}$$

2. Root Mean Square Error (RMSE)

- Purpose: Quantifies pixel-level differences between flattened saliency maps and ground truth.
- Calculation:
  - Lower RMSE = Better (Ideal: ~0.0).

$$RMSE = \frac{1}{HW} \|u_1 - u_2\|_2$$

Implementation

This implementation focuses on object detection using the SSD (Single Shot MultiBox Detector) model on the Pascal VOC 2012 dataset, followed by explainability through the BODEM (Black-box Object Detection Explanation by Masking) method.

Dataset - Pascal VOC 2012, taken from from torchvision.

Object detection Model: SSD300 with VGG16 backbone, initialized with pre-trained weights

Training

Modified the classification head to support 21 classes

- **Optimizer**: SGD with learning rate 0.001 and momentum 0.9.

BODEM Explanation Implementation

Generates saliency maps to explain SSD detections by highlighting important image regions for each detected object.

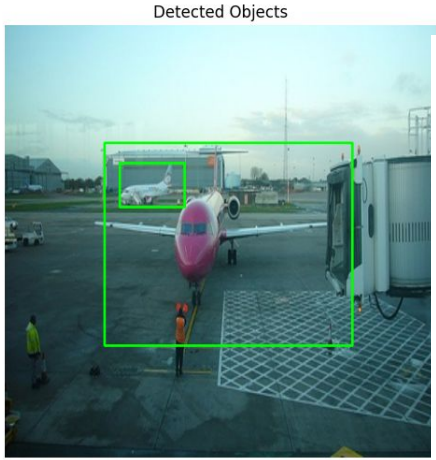
Parameters: Input image, target bounding box , detection function, number of layers L, initial block size K, and hyperparameters alpha=0.5 and beta=0.3 .

Process:

1. Initializes a saliency map (SM) of size (H x W).
2. Iterates over (L) layers, reducing block size ((BS = K / 2)).
3. For each layer:
    a. Divides image into blocks and selects candidate seeds (CS) based on prior saliency.
    b. Generates masks by randomly selecting a seed and 50% of its neighbors within distance (I).
    c. Applies masks to the image, setting masked blocks to zero.
    d. Computes impact scores (IS) using IoU between original and perturbed detections.
    e. Averages IS to get overall impact scores (OIS) per block.
    f. Update SM values based on OIS calculated and SM values at previous layer.

$$SM_n^l[b_p] = \begin{cases} (\alpha)SM_n^{l-1}[b_p] + (1-\alpha)OIS(b_p), & \exists m_q \in M^l | m_q(b_p) = 1 \; and \; OIS(b_p) \neq 0 \\ (\beta)SM_n^{l-1}[b_p], & otherwise \end{cases}$$

Saliency maps Generated by BODEM Method



Insertion score : 0.9000
Deletion score : 0.1040

Insertion score : 0.9000
Deletion score : 0.0500

Grad CAM Implementation(Classification)

- As discussed before

  ∑μ *RELU* ∑∑*(*(dy/dAk.Ak)) where μ is for normalising , Relu to concentrate on positive values

- Datasets used are : COCO(for multi labels)
- Model Selection : ResNet-50
- Used Last ConvLayer
- Mentioned in [3] ,

Grad CAM Implementation(Semantic Segmentation)

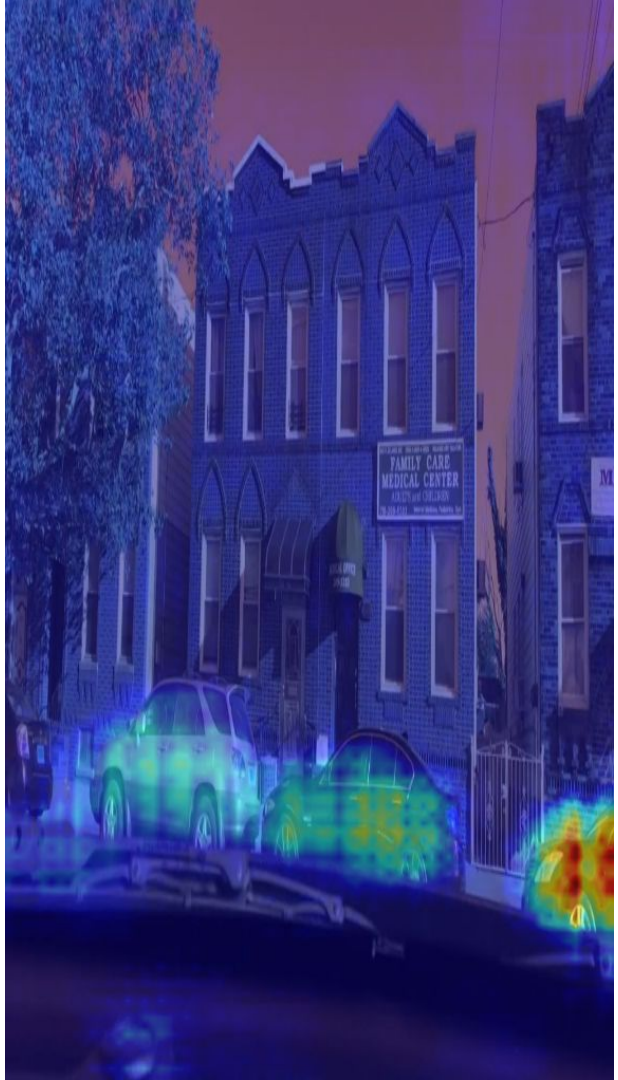
- As discussed before

  ∑μ *RELU* ∑∑*(*(dy/dAk.Ak)) where μ is for normalising , Relu to concentrate on positive values

- Datasets used are : used one image for inference
- Model Selection : Deeplabv3_resnet50
- Used Final ConvLayer,

Visualization of gradcam

Above image- for classification,
firstimage(bottle) and second image(table)
Right image - semantic segmentation
-Should work on semantic segmentation



Grad CAM++ Implementation

- Now as discussed in GradCAM same process but instead of weighting the activation map directly with gradient take RELU of gradient

  ∑μ *RELU* ∑∑*RELU(aij*(dy/dAk.Ak)) where μ is for normalising , Relu to concentrate on positive values

- Datasets used are : ImageNetmini
- Model Selection : ResNet-50, Xception
- Used Last Conv Layer
- Mentioned in [3] ,
- We are not able to get proper explanations for Xception Model

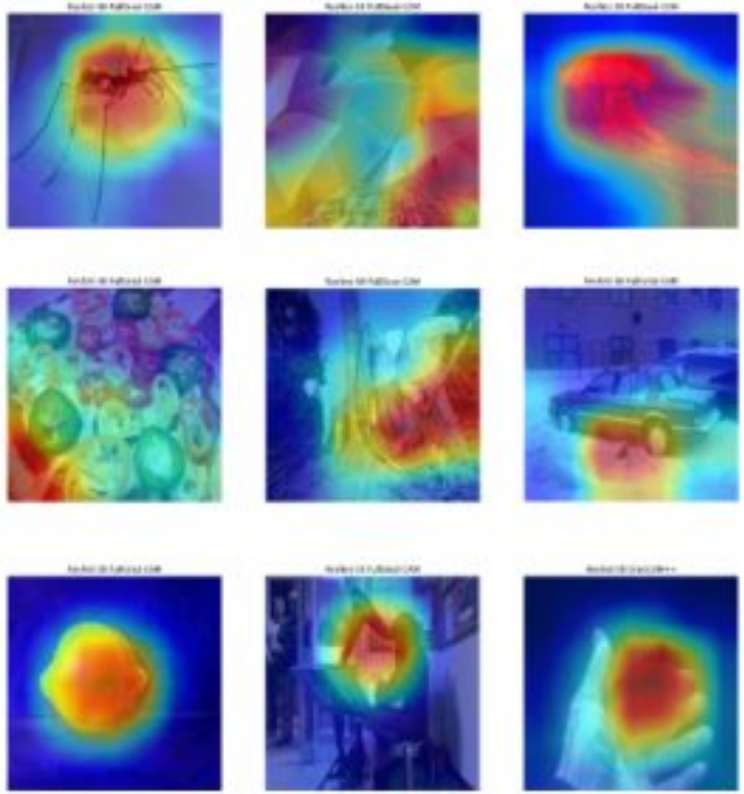
Full Grad CAM Implementation[1]

- Now as discussed in GradCAM same process but instead of weighting the activation map by multiplying we do element wise multiplication.

  $\sum \mu \ RELU \sum\sum(dy/dA \odot A)$ where μ is for normalising , Relu to concentrate on positive values

- Datasets used are : ImageNetmini
- Model Selection : ResNet-50, Xception
- Mentioned in [1] , Main purpose is for object detection we tried on classification
- We are not able to get proper explanations for Xception Model

Visualizations of heatmaps/cams



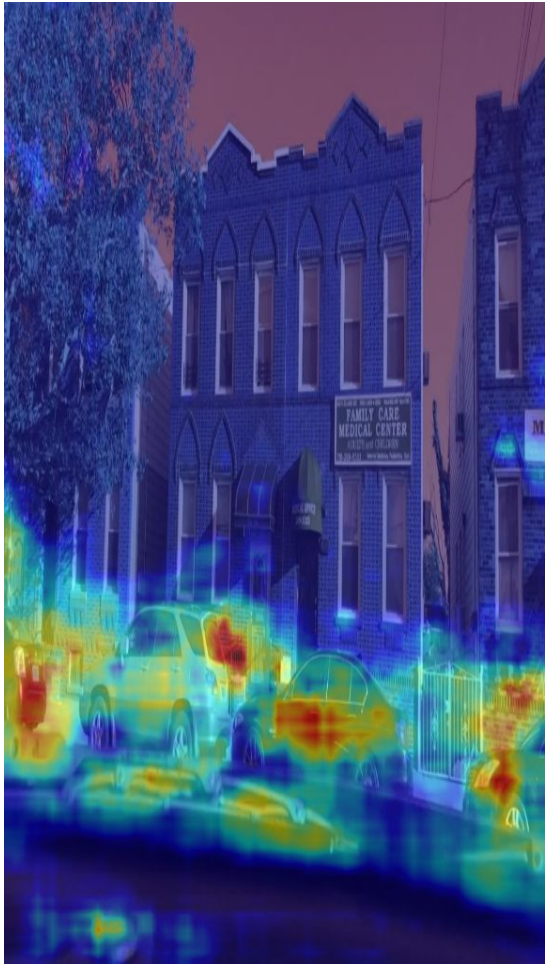
First 8 images are using FullGradCAM and last image is using GradCAM++.
This is obtained for Resnet-50

Metric	Value
Insertion Score	0.7663
Deletion Score	0.1681

Metric	Value
Insertion Score	0.655
Deletion Score	0.203

Eigen CAM Implementation[5]

- Now as discussed in GradCAM same process but instead of weighting the activation map the combined class activation map is multiplied ,by V1 vector
- Datasets used are : Used image for inference
- Last 6th layer is taken as target layer
- Model Selection : YOLOv5
- Mentioned in [5] ,



Results

Metrics Table:

Metric	Eigen CAM	GRADCAM(segmentation)
PCC	0.48198764449908266	0.2732025699221407
RMSE	0.18644159150883763	0.1280490380229643
SROCC	0.595199576793401	0.414123942846947915
Insertion Score	0.659085047096014	-
Deletion Score	0.26119450356593005	-

Learnings

- During this project, I was introduced to the field of Explainable AI (XAI), which was entirely new to me. Unlike traditional AI models that often act as "black boxes," XAI focuses on making AI decisions more transparent and understandable. This shift in perspective helped me appreciate the importance of interpretability and trust in AI systems
- I gained hands-on experience with several state-of-the-art explainability techniques, including:
- Grad-CAM
- Grad-CAM++
- FullGradCAM
- BODEM
- EigenCAM

Evaluation Techniques

A significant learning was understanding and applying evaluation metrics specific to explainability, such as:

- Faithfulness: How accurately the explanation reflects the model’s true reasoning.
- Interpretability: How easily a human can understand the explanation.

Learnings

Other Highlights

- Exposure to current research trends in AI transparency and ethics.
- Greater appreciation for the balance between model performance and explainability.

Individual Contributions

- Lahari- AI22BTECH11008 - GradCAM++, FullGradcam
- Divija - AI22BTECH11026 - BODEM
- Hima Chandh -AI22BTECH11009- GradCAM for Object Classification and Semantic Segmentation
- Anurag -AI22BTECH11011- EigenCAM for Object Detection