Integrated Grad-CAM: Sensitivity-Aware Visual Explanation of Deep Convolutional Networks via Integrated Gradient-Based Scoring

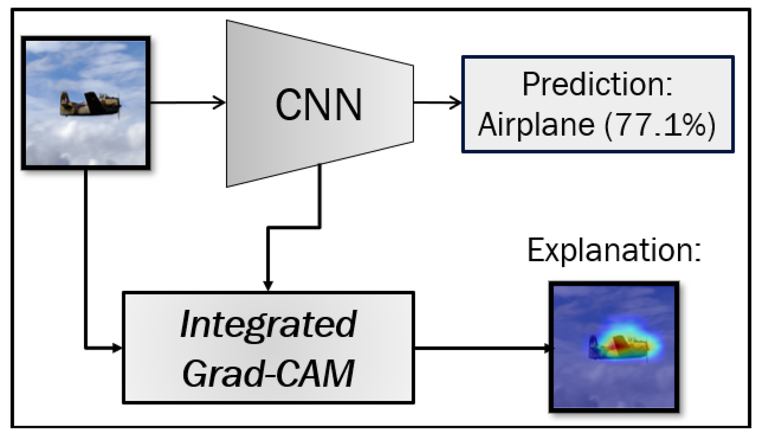
LG AI Research Sam Sattarzadeh, Mahesh Sudhakar, Konstantinos N. Plataniotis, Jongseong Jang, Yeonjeong Jeong, Hyunwoo Kim

University of Toronto, LG AI research

UNIVERSITY OF TORONTO

Introduction

- **Explainable AI (XAI):** Understanding Convolutional Neural Networks (CNNs) is crucial for high-impact and high-risk applications in computer vision.
- **Our aim:** *Visual Explainability:* Visualizing the input features responsible for CNN prediction.



Background

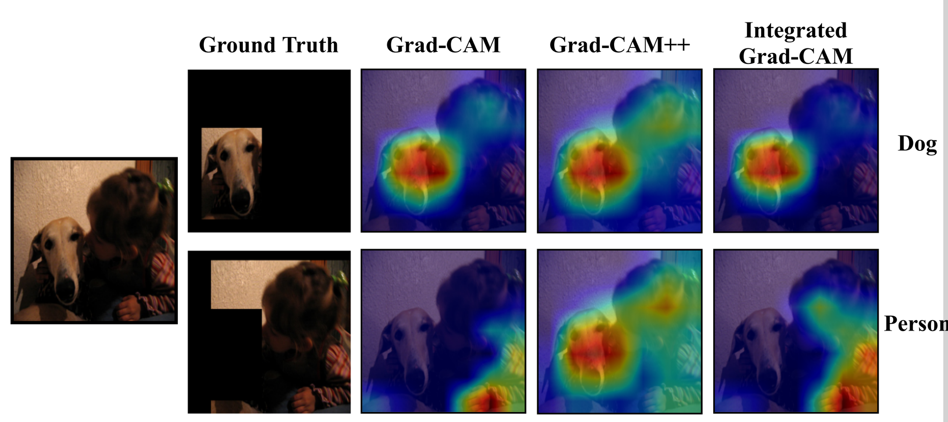
- **Methods for visual explainability:**
  - **Backpropagation-based methods :** Computing the gradient of CNN's output to the input features or hidden neurons.
  - **CAM-based methods :** Visualizing the features extracted in a single layer of the CNNs.
  - **Perturbation-based methods :** Probing the model's behavior using perturbed copies of the input image.

Contributions

- Our proposed approach: **Integrated Grad-CAM**
- Addressing the limitations of backpropagation in explaining non-linear models.
- Solving the gradient limitations by employing gradients.

Integrated Grad-CAM

Novelty:
Scoring the feature maps in the last convolutional layer of CNNs based on *Integrated Average Gradient* values, instead of "Average Gradient" values utilized in Grad-CAM.

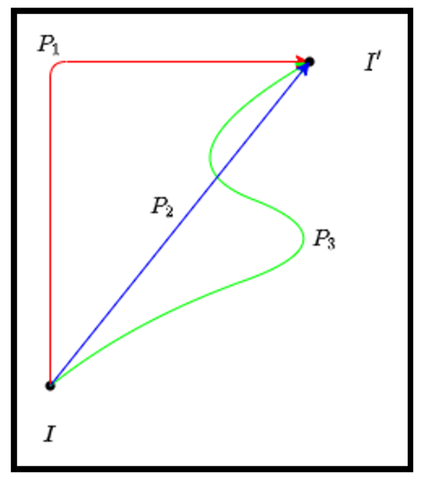


Intuition: Sensitivity axiom: (Sundararajan et al. '17) *For each pair of input and baseline differing only in one feature, an attribution method should highlight this difference by assigning different values corresponding to that feature.*

Idea: Calculating the integral of gradient values in a path that links a certain baseline to the input.

Path Integral

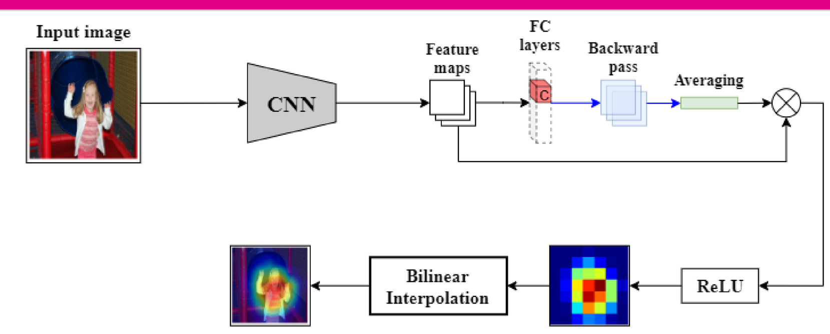
Defining a path linking a baseline I' and an input I:



Path equation: $\gamma(\alpha) = I' + f(\alpha) \times (I - I')$
$f(\alpha) : \mathbb{R} \to \mathbb{R}$:
A differentiable and monotonically increasing function.
$0 \leq \alpha \leq 1$: $f(0) = 0$ and $f(1) = 1$.
For each pair of functions $(h(.), g(.))$:

$$\text{PathIG}_{h,g}(I) \equiv \int_{\alpha=0}^{1} \frac{dh(\gamma(\alpha))}{dg(\gamma(\alpha))}[g(\gamma(\alpha)) - g(I')]d\alpha \quad (1)$$

Methodology



Feature maps derived from a convolutional layer (l): $\{A^{l1}(I), A^{l2}(I), ..., A^{lN}(I)\}$
Grad-CAM formulation:

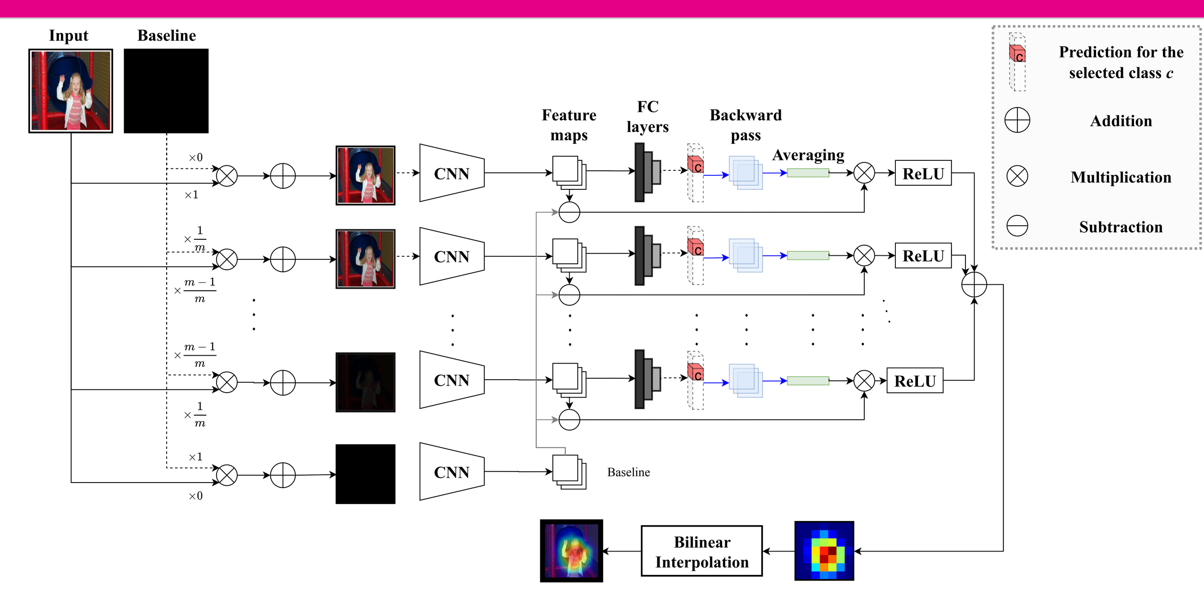
$$M^c_{Grad-CAM} = \text{ReLU}(\sum_{k=1}^{N}(\frac{1}{Z}\sum_{i,j}\frac{\partial y_c(I)}{\partial A^{lk}_{ij}(I)})A^{lk}(I)) \quad (2)$$

Our method: Replacing gradient terms with integrated gradient terms:

$$M^c_{IG-CAM} = \int_{\alpha=0}^{1} \text{ReLU}(\sum_{k=1}^{N}\sum_{i,j}\frac{\partial y_c(\gamma(\alpha))}{\partial A^{lk}_{ij}(\gamma(\alpha))}[A^{lk}(\gamma(\alpha)) - A^{lk}(I')])d\alpha \quad (3)$$

Limitation: The equation above is hard to implement.

Implementation



For simplicity, we assume the path between I' and I to be linear. Then, we approximate the equation (3) using *Reimann's approximation*.

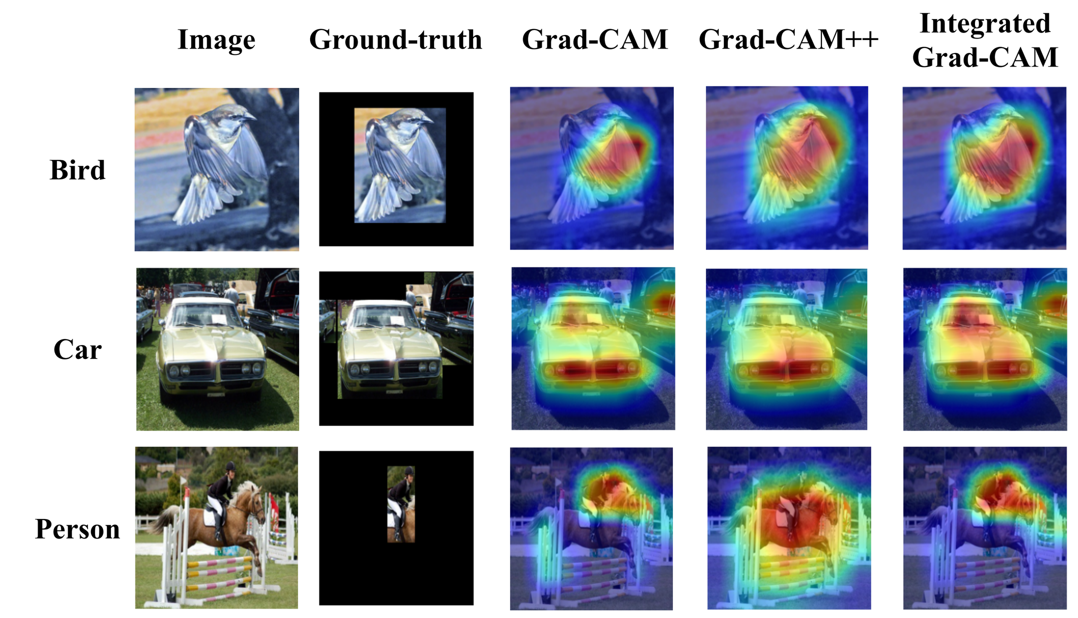
$$M^c \approx \sum_{t=1}^{m} \text{ReLU}(\frac{1}{m}\sum_{k=1}^{N}\sum_{i,j}\frac{\partial y_c(\gamma(\frac{t}{m}))}{\partial A^{lk}_{ij}(\gamma(\frac{t}{m}))})[A^{lk}(\gamma(\frac{t}{m})) - A^{lk}(I')]) \quad (4)$$

The number of sampled points along the linear path: 'm' (set to 20 by default.)

Experiments

Dataset: PASCAL VOC 2007

- **Purpose:** Multi-label image classification, Object Detection.
- Containing 4963 test images in 20 classes, Bounding boxes provided.
- A VGG-16 model and a ResNet-50 model trained on this dataset are utilized.



Quantitative Evaluation

Evaluation metrics:

- *Ground truth-based* like Energy-based Pointing Game (**EBPG**), Mean Intersection-over-Union (**mIoU**) and Bounding Box (**Bbox**) are used to verify the meaningfulness of explanation methods, and their ability in feature visualization.
- *Model truth-based* like **Drop** and **Increase rate** are employed to justify the faithfulness and validity of the generated explanations from the model's perspective.

Model	Metric	Grad-CAM	Grad-CAM++	Integrated Grad-CAM
VGG16	EBPG	55.44	46.29	**55.94**
	Bbox	51.7	55.59	**55.6**
	Drop	49.47	60.63	**47.96**
	Increase	31.08	23.89	**31.47**
ResNet-50	EBPG	60.08	47.78	**60.41**
	Bbox	60.25	58.66	**61.94**
	Drop	35.80	41.77	**34.49**
	Increase	36.58	32.15	**36.84**

Conclusion

Integrated Grad-CAM Takeaways:

- Circumvented the underestimations in Grad-CAM and Grad-CAM++.
- Addressed the issues caused by backpropagation in the methods above.
- Though slower than the conventional methods, offers an acceptable run-time to be used in real-world applications.
- The takeaways above are verified through extensive experiments on the PASCAL VOC 2007 dataset.

References

Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization." (2018).
Sundararajan, Mukund, Ankur Taly, and Qiqi Yan. "Axiomatic attribution for deep networks." (2017).