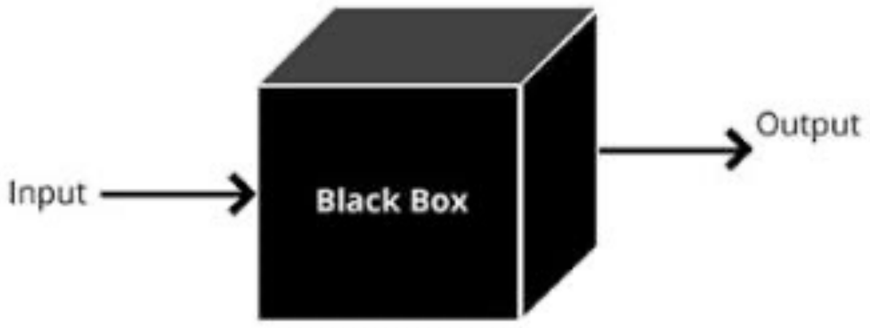
RESTRICTING THE FLOW
INFORMATION BOTTLENECKS FOR ATTRIBUTION

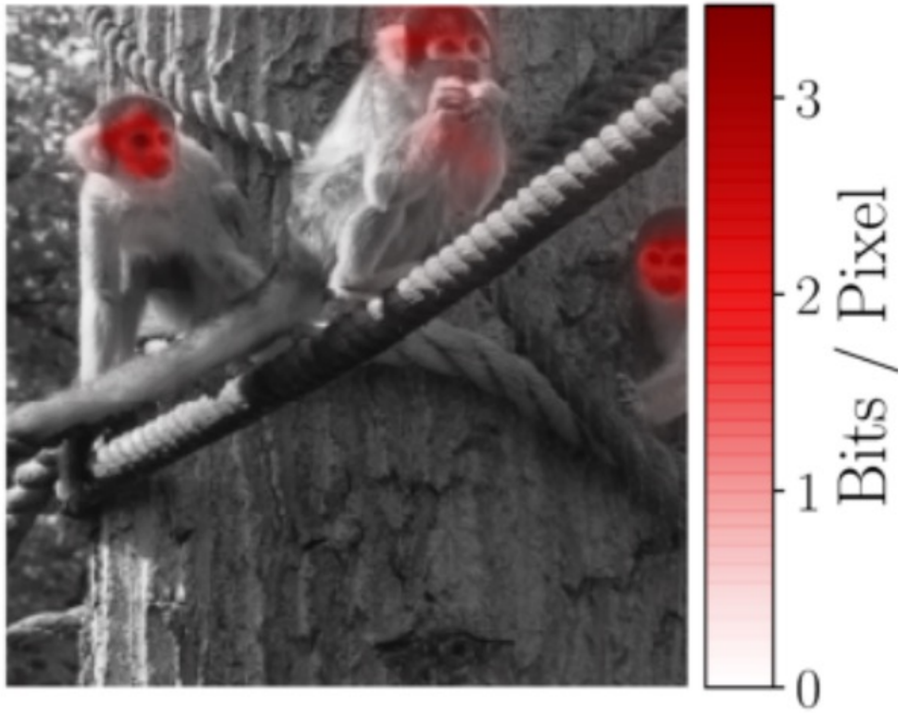
HOOMAN RAMEZANI

Applying the Information Bottleneck for Attribution

- Deep Learning Interpretability: Demystify model reasoning by highlighting important inputs.
- Role of Attribution Methods: Quantifies the influence of input features on model predictions.
- Why Information Bottlenecks for Attribution? Tailored to isolate crucial features by measuring their information contribution
  - Effectively filter out irrelevant or redundant information -> Principle of minimal sufficient statistics
  - Quantify of how much each input feature contributes to the model's decision



The Papers Contributions



- **Adapts** information bottleneck for **attribution** to estimate the information used
  - Information theory guarantees that areas scored irrelevant are not necessary for prediction.
- **Evaluation of** attribution is difficult – no ground truth exists
  - Novel evaluation method – bounding boxes
  - Metric, Ancona et al. (2017), to provide a single scalar value and improve the metric's comparability.

How it Works

1. Introduce *Z* to limit information flow Objective $\max I[Y;Z] - \beta I[X,Z]$
2. Attribution: Inject the bottleneck into a target layer of pre-trained network.
3. Noise Addition: Reduce information by adding noise to intermediate representation *(R)*
4. Estimate mean μ_R and variance σ_R^2 of R
5. Signal and Noise Interpolation: Linear interpolation between signal *(R)* and noise to create *Z,* $\lambda(X)$ controls the mixing

$$Z = \lambda(X)R + (1 - \lambda(X))\epsilon\ ,$$

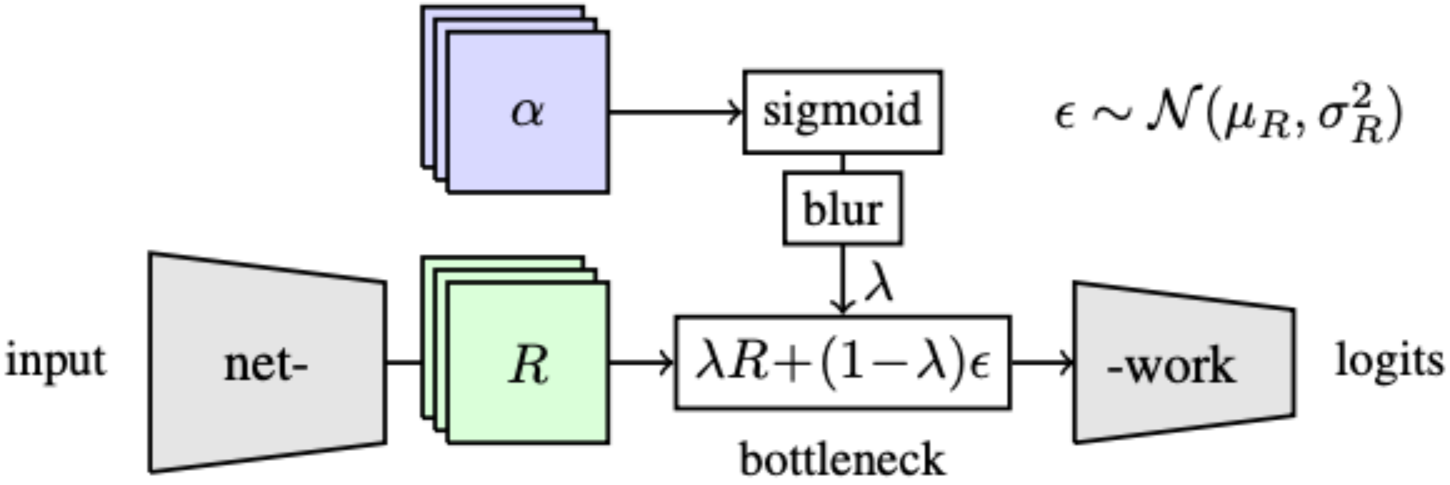
6. Mutual Information allows attribution process to identify features of greatest importance
   - Not directly computable, estimated with variational approximations

$$\mathrm{I}[R, Z] = \mathbb{E}_R[D_{\mathrm{KL}}[P(Z|R)||Q(Z)]] - D_{\mathrm{KL}}[P(Z)||Q(Z)]$$

Per Sample Bottleneck

Per-Sample Bottleneck: Tailored for individual data points.

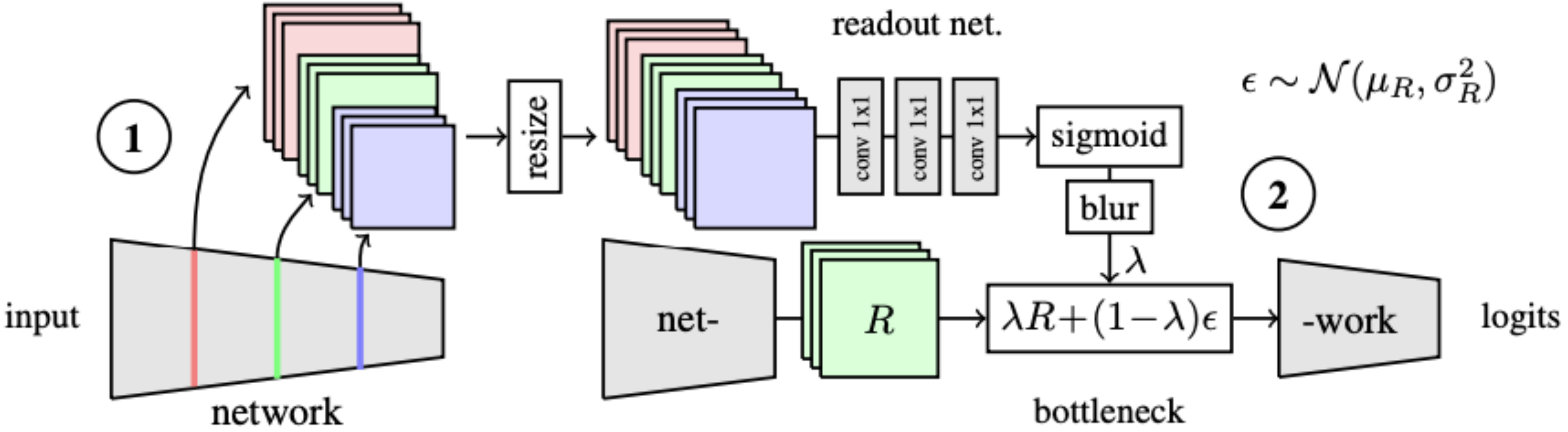
- Optimizes noise for each sample.
- Focuses on what's essential for specific instances.



Readout Bottleneck

Readout Bottleneck: Employs the entire dataset.

- Optimizes a global noise pattern.
- Seeks universally significant features across data.



Paper Results

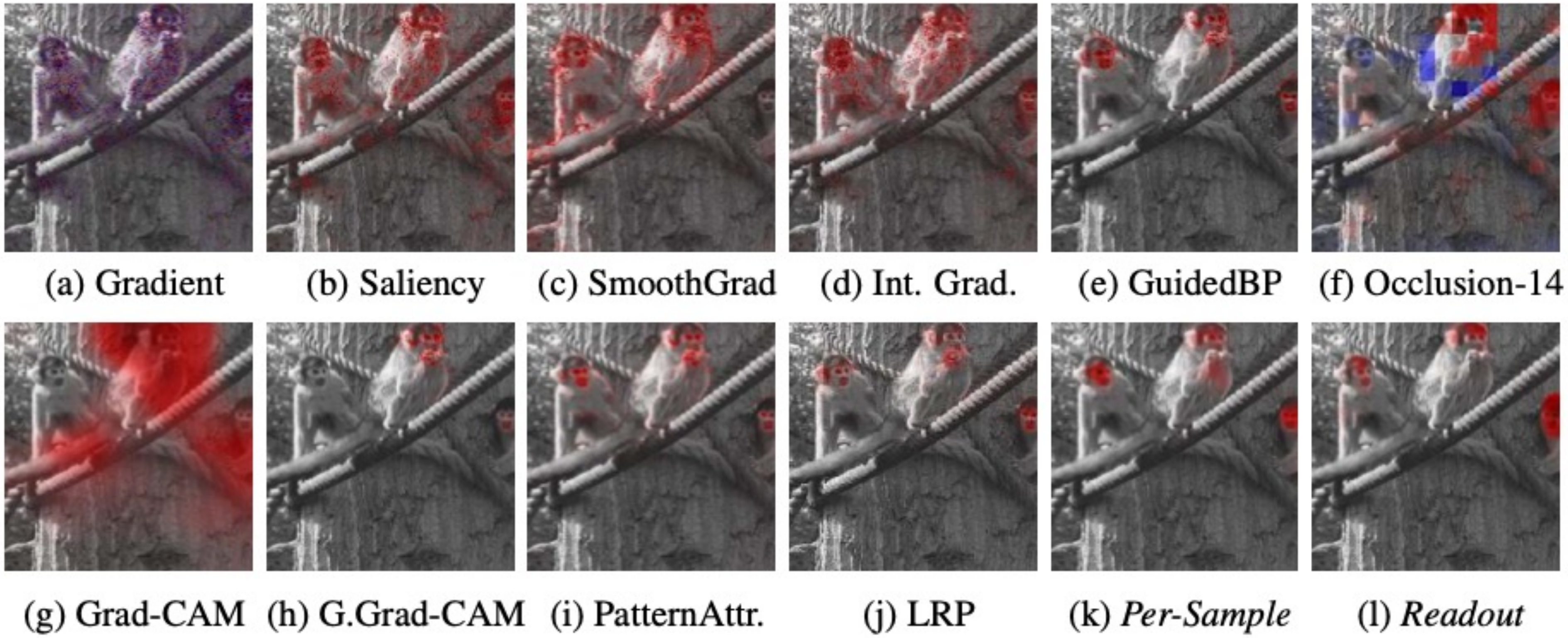
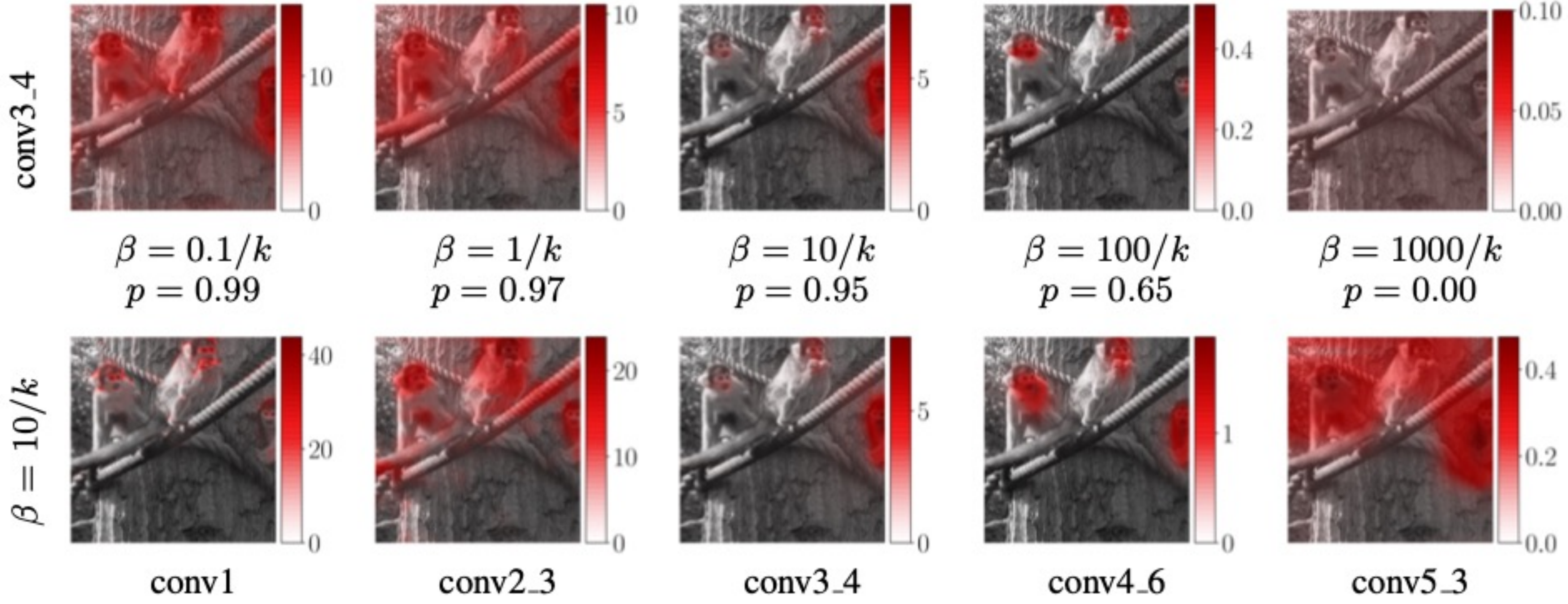


Figure 5: Heatmaps of all implemented methods for the VGG-16 (see Appendix A for more).

Optimizing Tradeoff



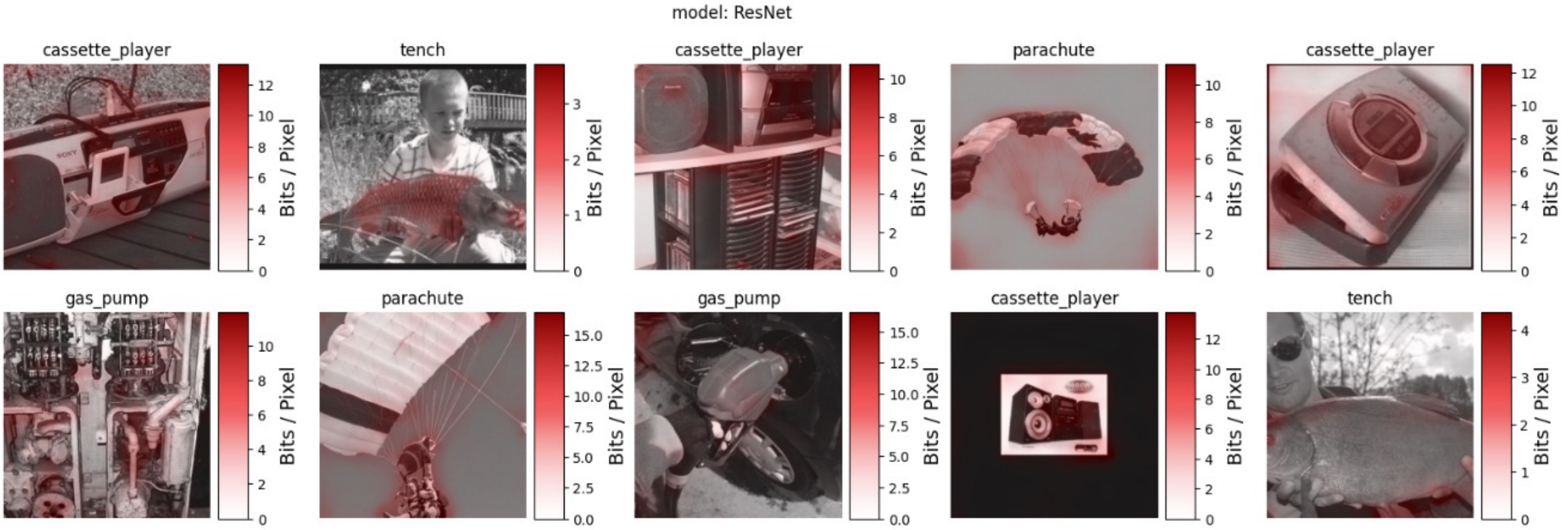
Quantitative

- **Degradation Test**: Assesses model performance when relevant features are masked out.
  - **Results**: Per-Sample Bottleneck outperforms, showing a significant margin in preserving model performance.
- **Bounding Box Method**: Ratio of top-n highest scored pixels (per the attribution) are within the bounding box of the object
  - **Results**: Per-Sample Bottleneck excels, with a higher ratio of relevant pixels identified within bounding boxes

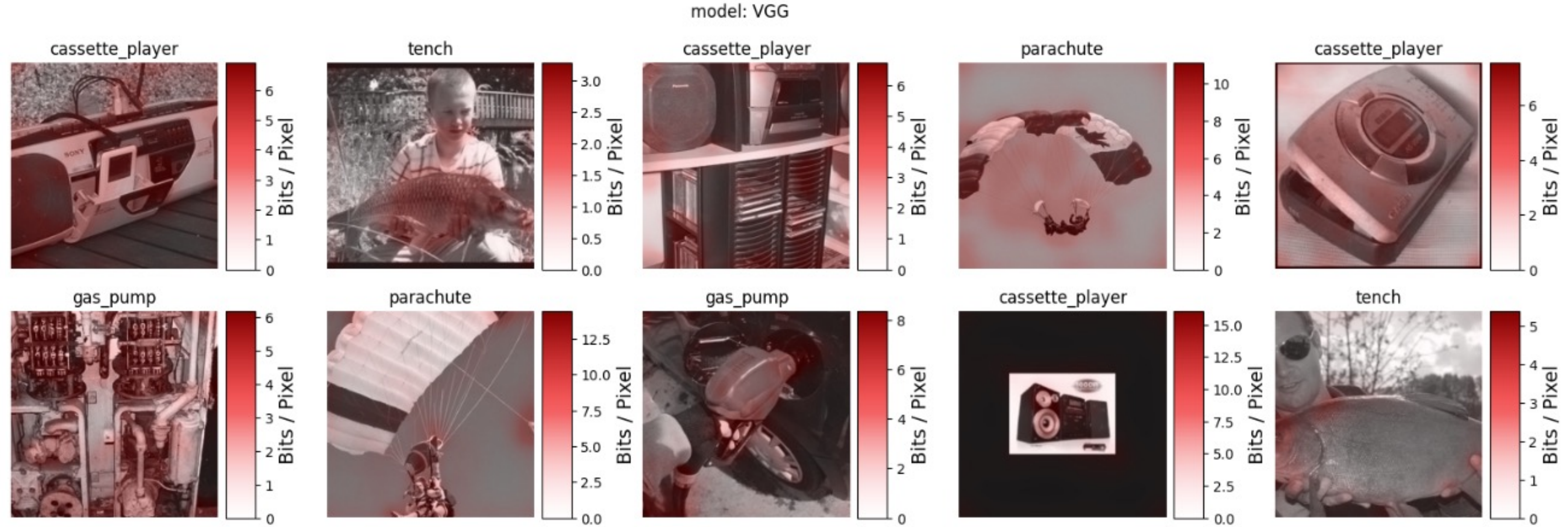
Model & Evaluation	ResNet-50 deg. 8x8	ResNet-50 deg. 14x14	VGG-16 deg. 8x8	VGG-16 deg. 14x14	ResNet bbox	VGG bbox
Random	0.000	0.000	0.000	0.000	0.167	0.167
Occlusion-8x8	0.162	0.130	0.267	0.258	0.296	0.312
Occlusion-14x14	0.228	0.231	0.402	0.404	0.341	0.358
Gradient	0.002	0.005	0.001	0.005	0.259	0.276
Saliency	0.287	0.305	0.326	0.362	0.363	0.393
GuidedBP	0.491	0.515	0.460	0.493	0.388	0.373
PatternAttribution	–	–	0.440	0.457	–	0.404
LRP $\alpha=1, \beta=0$	–	–	0.471	0.486	–	0.397
LRP $\alpha=0, \beta=1, \epsilon=5$	–	–	0.462	0.467	–	0.441
Int. Grad.	0.401	0.424	0.420	0.453	0.372	0.396
SmoothGrad	0.485	0.502	0.438	0.455	0.439	0.399
Grad-CAM	0.536	0.541	0.510	0.517	0.465	0.399
GuidedGrad-CAM	0.565	**0.577**	0.555	0.576	0.468	0.419
IBA Per-Sample $\beta=1/k$	**0.573**	0.573	0.581	0.583	0.606	0.566
IBA Per-Sample $\beta=10/k$	0.572	0.571	**0.582**	**0.585**	**0.620**	**0.593**
IBA Per-Sample $\beta=100/k$	0.534	0.535	0.542	0.545	0.574	0.568
IBA Readout $\beta=10/k$	0.536	0.536	0.490	0.536	0.484	0.437

Table 1: *Degradation (deg.)*: Integral between LeRF and MoRF in the degradation benchmark for different models and window sizes over the ImageNet test set. *Bounding Box (bbox)*: the ratio of the highest scored pixels within the bounding box. For ResNet-50, we show no results for PatternAttribution and LRP as no PyTorch implementation supports skip-connections.

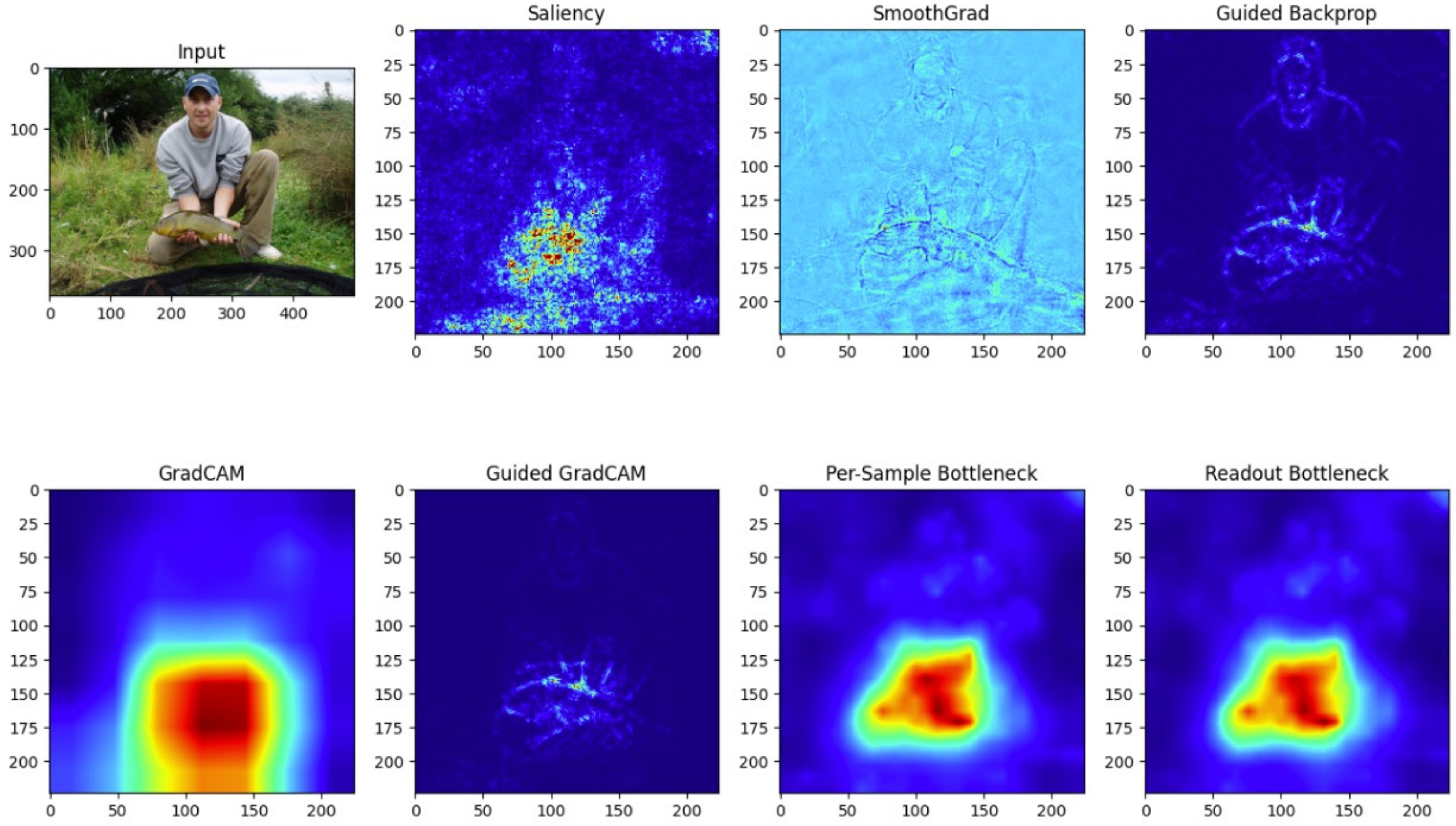
Reproduction of Results – ResNet-50



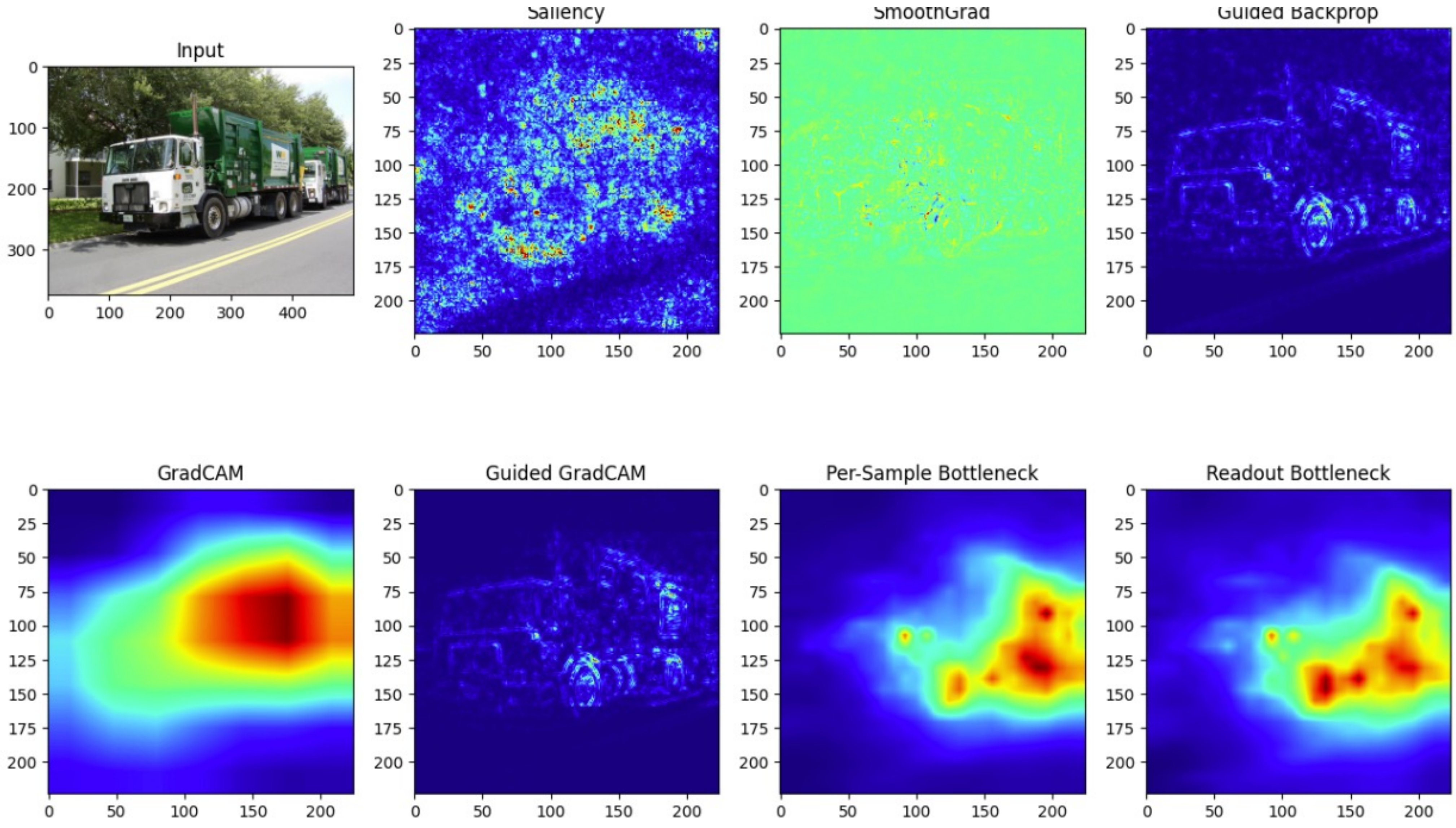
VGG-16


model: VGG
cassette_player
tench
cassette_player
parachute
cassette_player
gas_pump
parachute
gas_pump
cassette_player
tench
Bits / Pixel

Comparing Methods – Trench

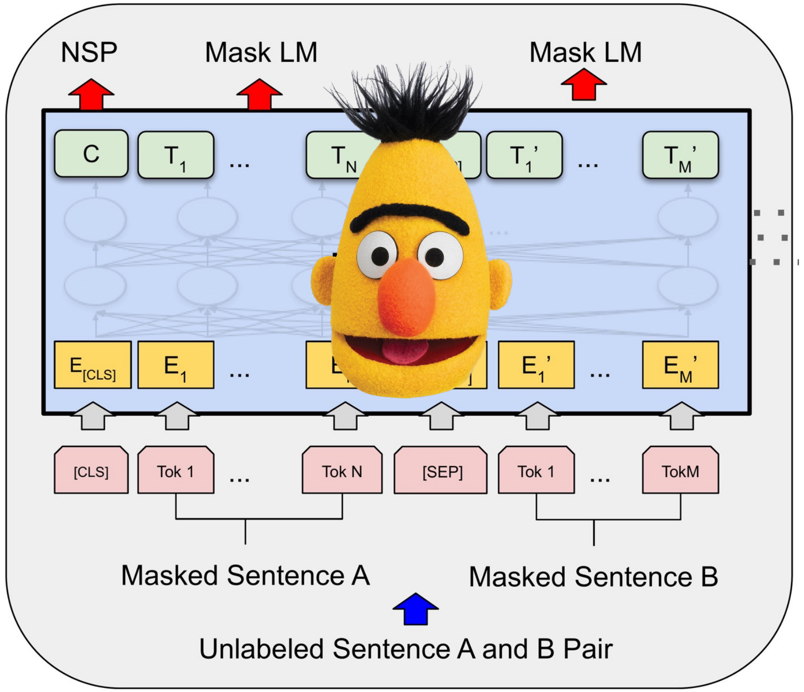


Comparing Methods – Garbage Truck



Can This Be Applied To Transformers?

- We will continue by exploring the extension of the IB method to Transformer models
    - Transformers, unlike CNNs, are dominant in NLP tasks but lack clear interpretability.
    - Understanding feature attribution is especially important in Transformers
- We will adapt IBA to **BERT-based** transformers
    - Assessing the impact on prediction probability by removing tokens identified as important through IB

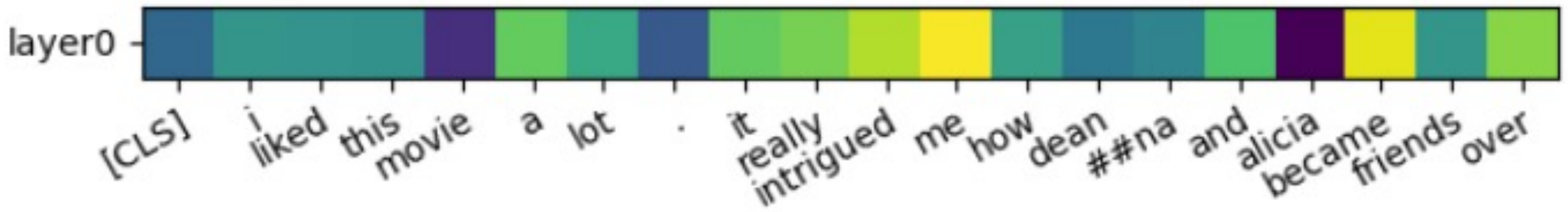


Pre-training

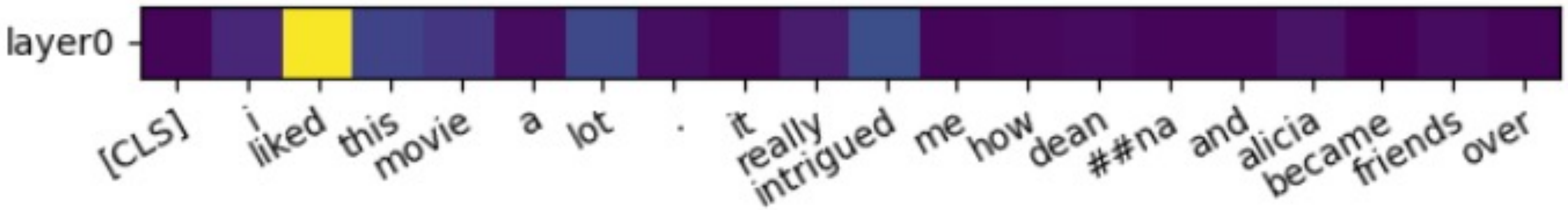
Method

- **Instance level:** focusing on token-level features and cross-layer behavior
  - Different transformer layers encode different types of information
  - Help us find the most information rich layers
- Dealing with token representations of text rather than images
- Due to higher complexity and greater layer interconnectedness layer is very important and less predictable, more nuanced

IMDB Movie Reviews

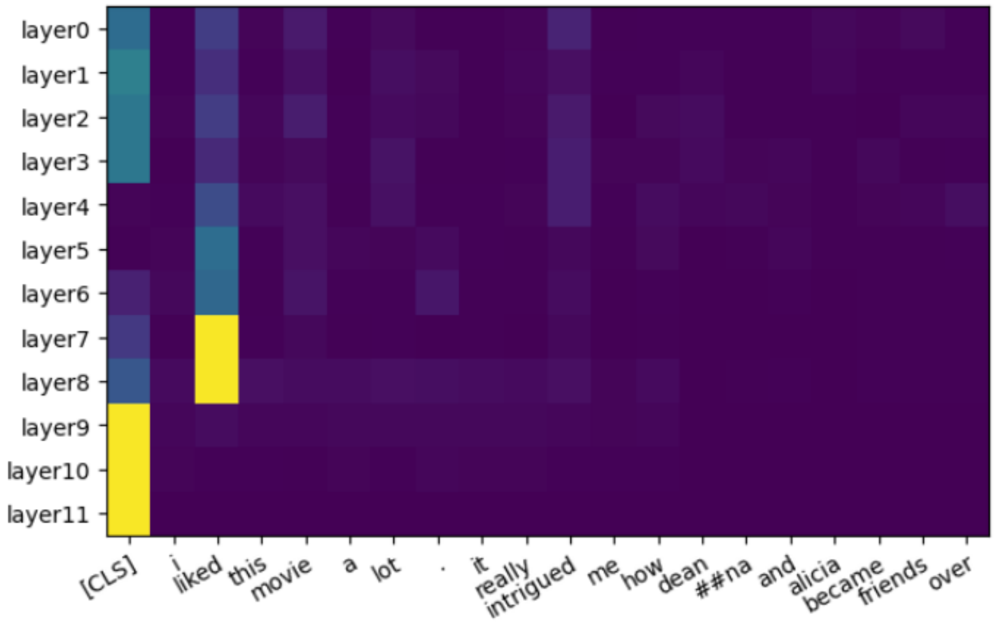


General BERT model on IMDB positive review

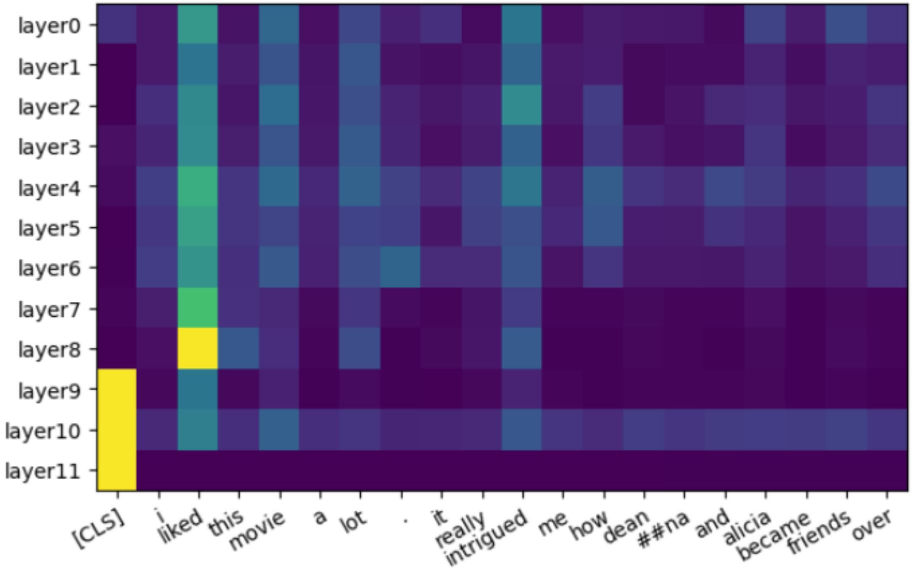


Finetuned BERT model on IMDB positive review

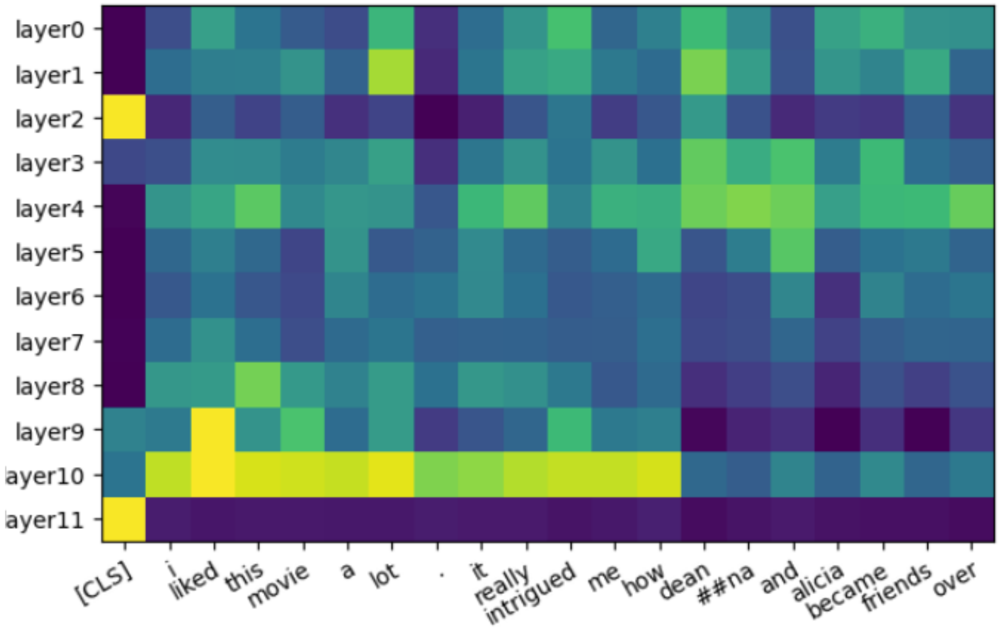
Adjusting β (noise)



$\beta = 1 \times 10{-}5$

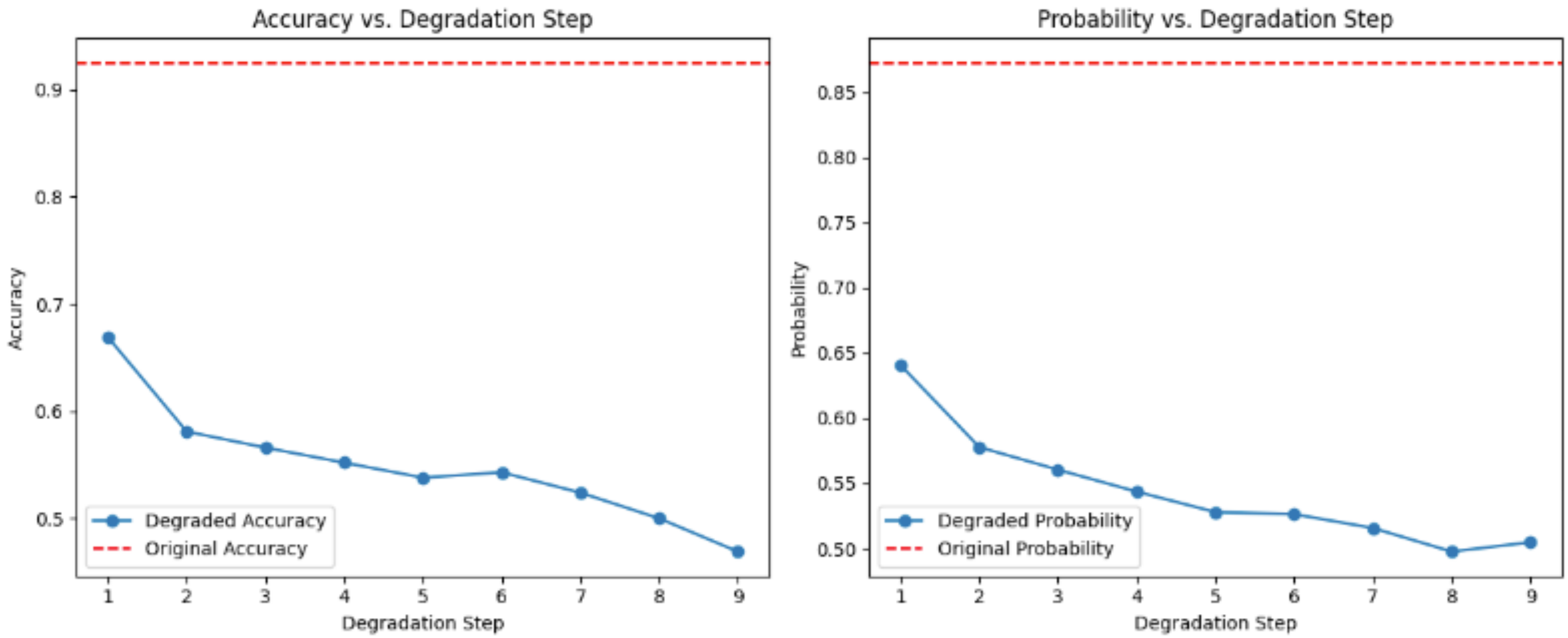


$\beta = 1 \times 10{-}4$



$\beta = 1 \times 10{-}6$

Degradation Test – Removing Top k Tokens



Thank You