Assignment 3 Writeup

DO NOT TAG

Name:
GT Email:

Visualization

DO NOT TAG

Implementation Question 1

DO NOT TAG

In your coding homework, you were given the following hint:
“There are two approaches to performing backprop using the PyTorch command tensor.backward()… Alternatively, one can take the sum of all the elements of the tensor and do a single backprop with the resulting scalar. This second approach is simpler and preferable as it lends itself vectorization.”

Question: Referring to the coding task completed by you, why is the suggested alternative approach mathematically sound? Please provide a brief but succinct answer on the next slide.

Answer for Implementation Question 1

Answer:

The alternative approach is mathematically sound because differentiation is a linear operation. When you take the sum of all elements in the tensor and compute the gradient using tensor.backward(), it is equivalent to calculating the gradient for each element individually and then summing them. By summing the tensor first and performing a single backpropagation, you efficiently combine these calculations into one step. This simplifies the process without altering the mathematical outcome, as the gradients for each element are still being properly accumulated, leading to the same result as the element-wise approach. By leveraging the sum, we effectively compute the gradient over the whole tensor, which simplifies computation. This speeds up calculations exponentially which is very important as these are calculations we have to do many many times during training.

Implementation Question 2

DO NOT TAG

In your network visualization tasks, you need to compute gradients for which one of the following three quantities:

A. Cross entropy loss
B. Unnormalized score corresponding to the correct class
C. Class probabilities

Please answer on the next slide.

Now briefly justify why the other two options are not optimal.

Answer for Implementation Question 2

Answer (A, B or C):

B

Now briefly justify why the other two options are not optimal for tasks on hand.

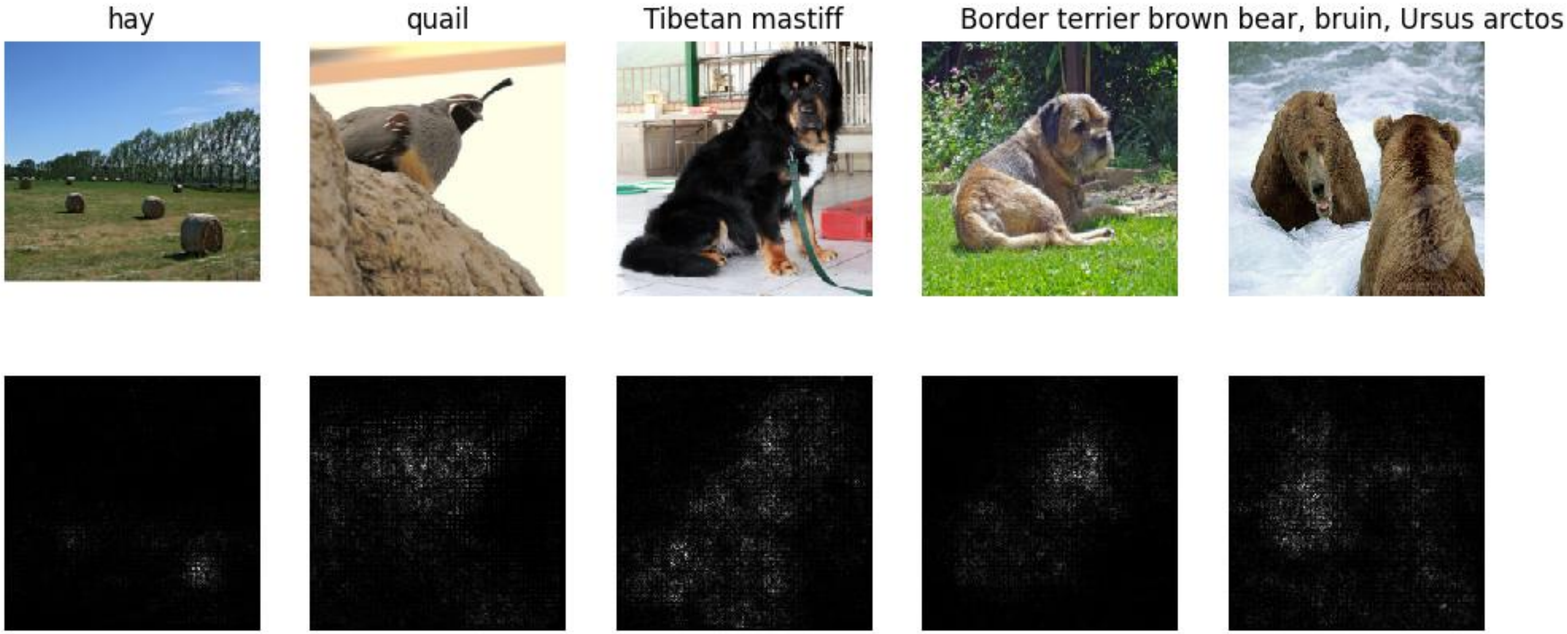
Cross entropy loss: The cross entropy loss is a scalar value that measures the overall difference between the predicted class probabilities and the true class labels. While it is essential for training, it aggregates information across all classes, blending the contributions of correct and incorrect predictions. This makes it less focused on a single class, which limits its utility in visualizing how specific neurons respond to a particular class. In other words, using cross entropy loss for gradient computation dilutes the interpretability, as it reflects a combination of multiple class outputs rather than focusing on the one you are interested in.

Class probabilities: The class probabilities are the softmax-normalized outputs of the network. While they represent the likelihood of each class, they are the result of a softmax transformation, which introduces non-linearity and interaction between all class scores. This non-linearity makes the gradients less informative, as changes in one class probability affect all others. For visualization purposes, the unnormalized scores are more direct and allow for clearer insight into how the network responds to the specific class of interest, without the obscuring effects of the softmax normalization.

By computing gradients of the unnormalized score corresponding to the correct class, you retain more direct and interpretable information about how specific features in the input influence the network’s decision for that particular class.

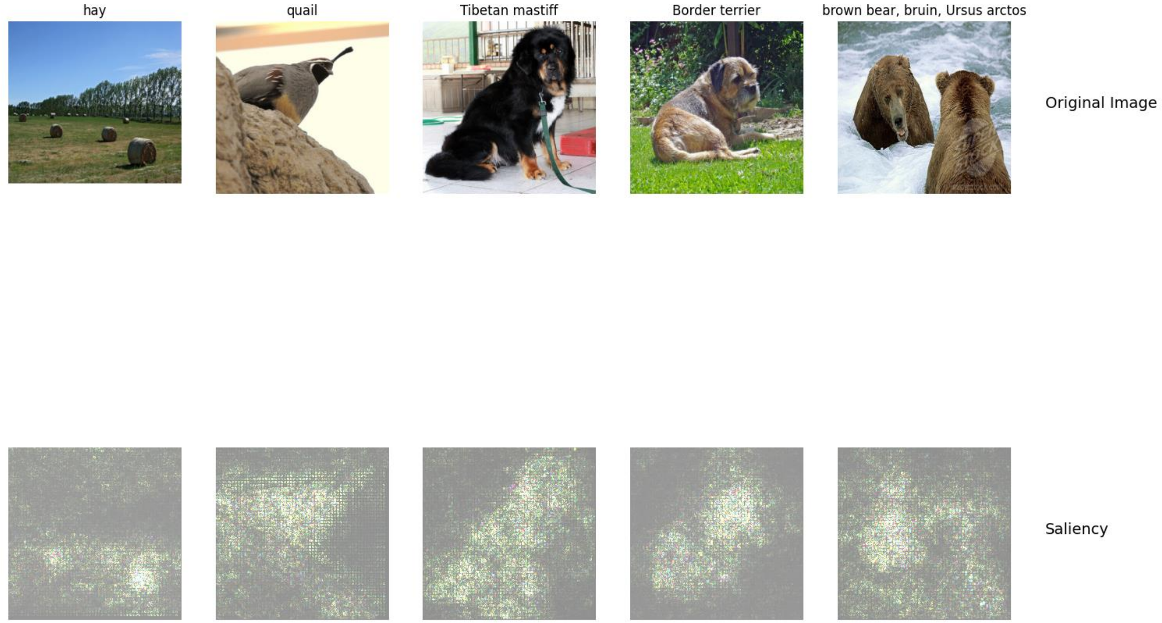
Saliency Map

- Include your saliency map here



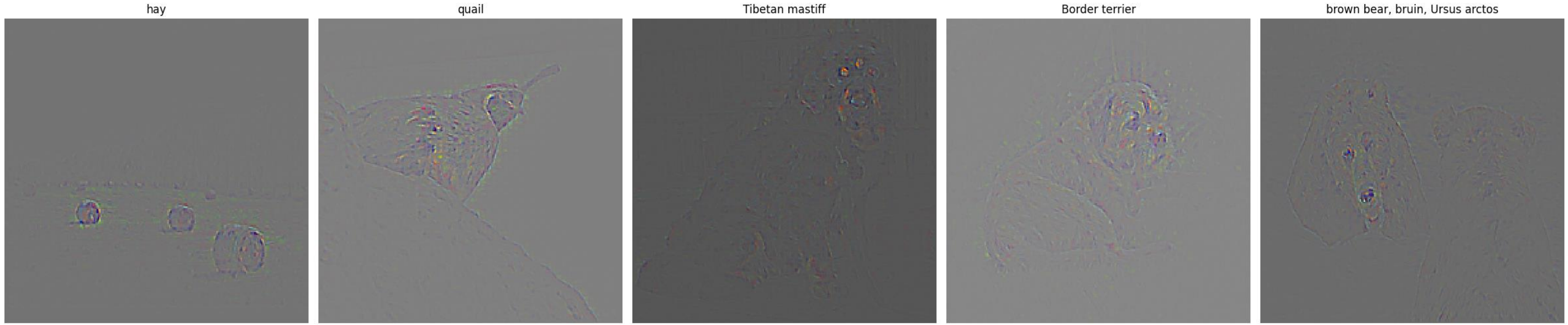
Saliency Map

- Include your saliency map from Captum here



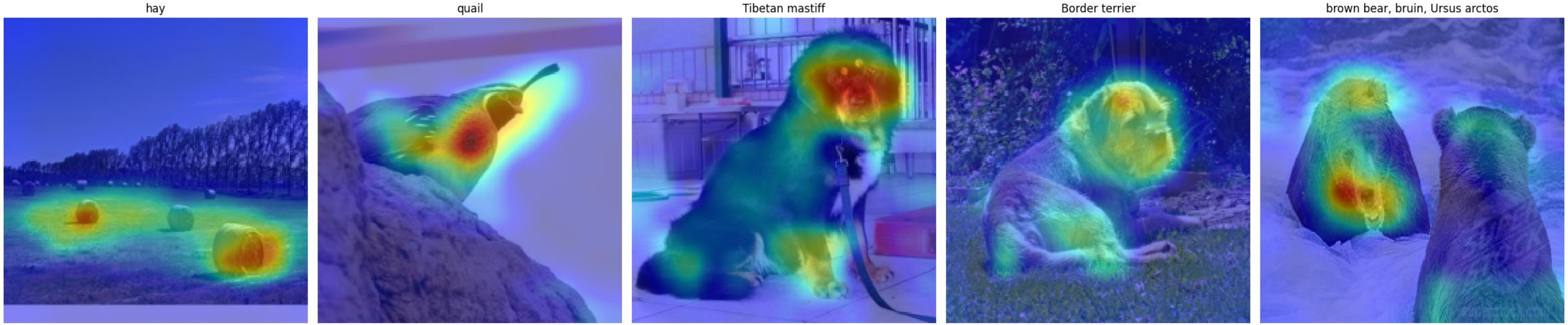
GradCam

- Include your visualization of Guided Backprop here



GradCam

- Include your visualization of GradCam here



hay

quail

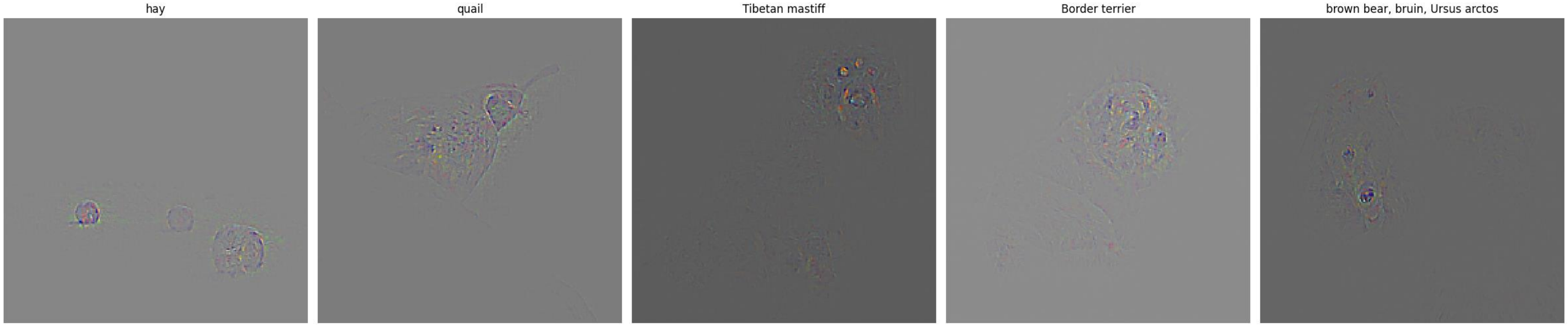
Tibetan mastiff

Border terrier

brown bear, bruin, Ursus arctos

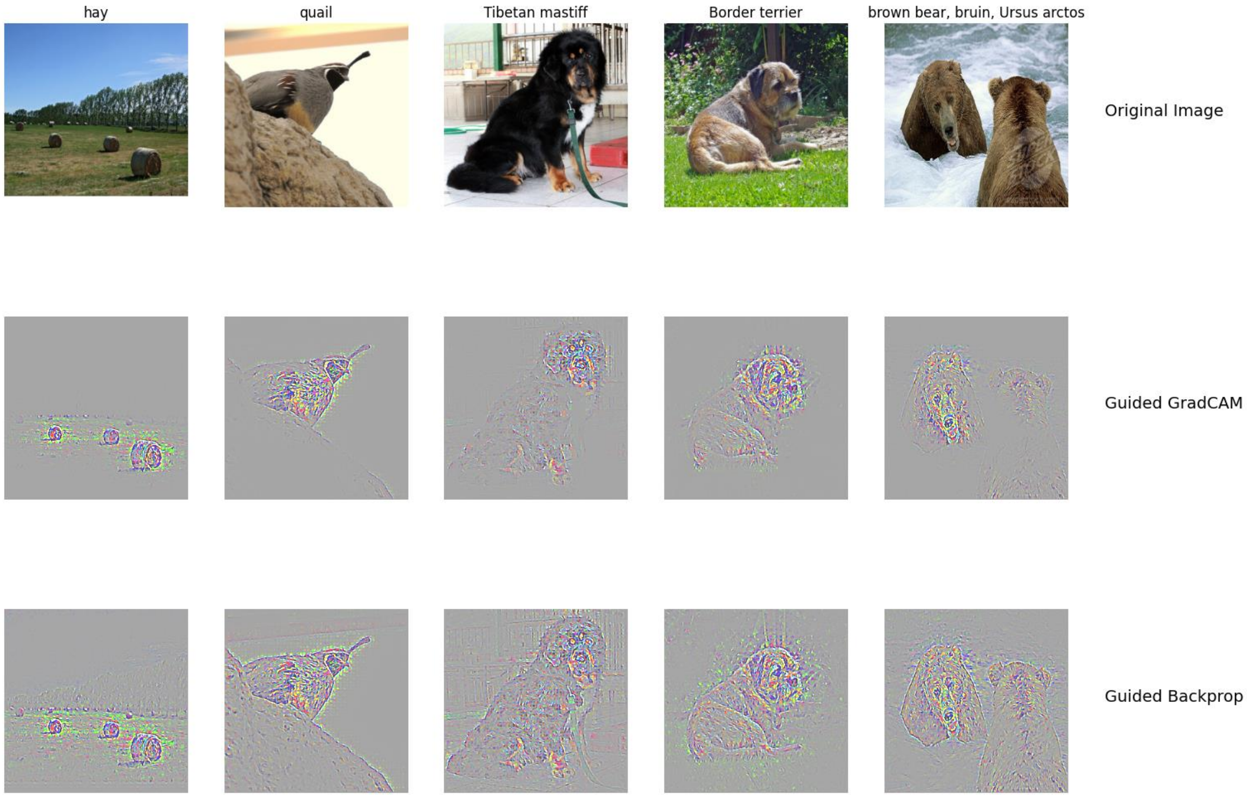
GradCam

- Include your visualization of Guided GradCam here



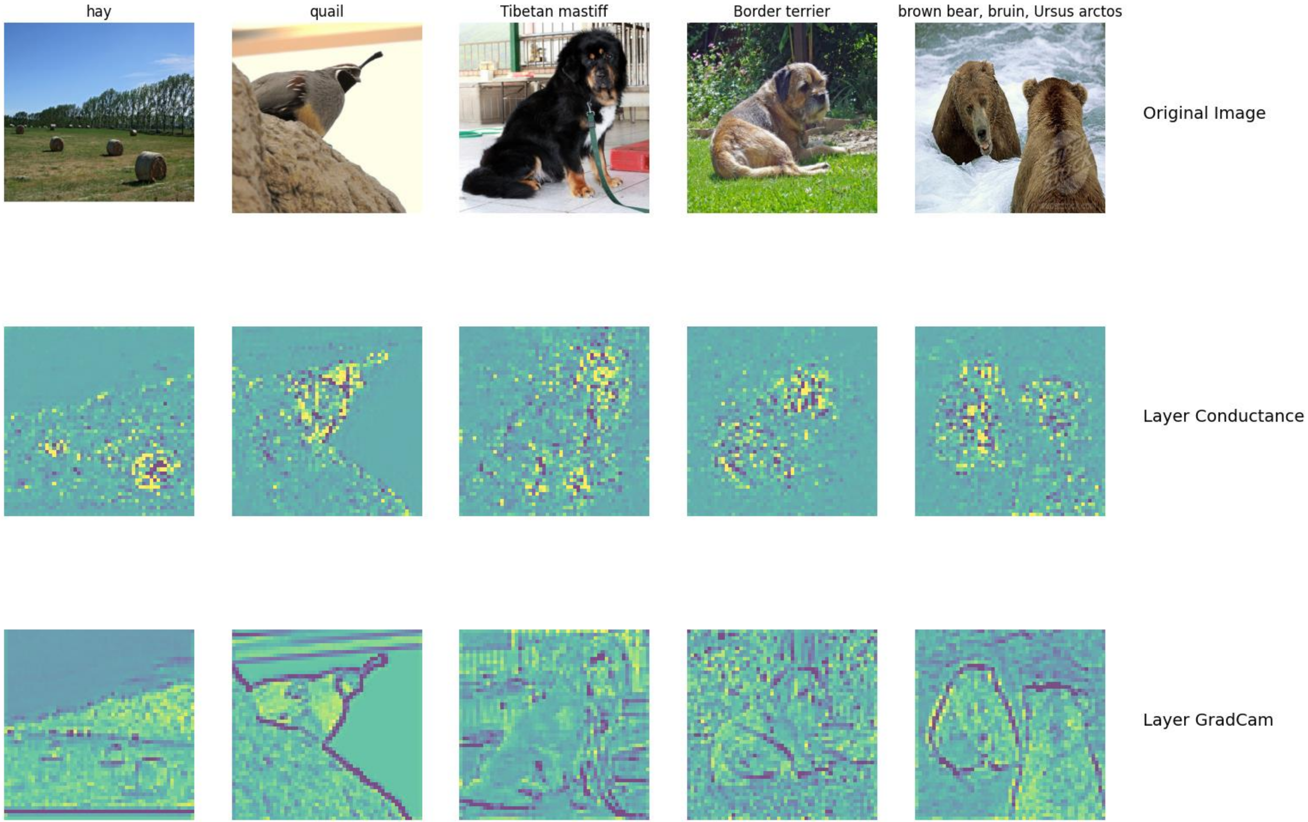
GradCam

- Include your visualization of Guided Backprop and Guided Gradcam from Captum here



GradCam

- Visualization of layers and neurons using Captum here:



What do saliency map and Gradcam tell you? How are they different? Is one better than the other?

Answer:

Saliency Maps and Grad-CAM (Gradient-weighted Class Activation Mapping) are both techniques used to interpret and visualize the inner workings of deep learning models, particularly in convolutional neural networks (CNNs). They help understand which parts of an input (such as an image) contribute most to a model’s decision.

What they tell you

Saliency Map: A saliency map highlights which pixels in an input image influence the model's prediction the most. It computes the gradient of the output (often a class score) with respect to the input image pixels. The magnitude of these gradients indicates how sensitive the model’s output is to changes in each pixel. Saliency maps essentially tell you which parts of the input have the greatest impact on the final prediction, making them useful for understanding how the network is responding to features across the image.

Grad-CAM: Grad-CAM works at a higher, more abstract level by highlighting which regions of the input image are most important in determining the model’s prediction. It uses the gradients flowing into the last convolutional layer to identify the most relevant regions in the image that influenced the model's decision. Grad-CAM provides a heatmap of importance over the image, which is more interpretable, as it focuses on larger spatial regions rather than individual pixels.

What do saliency map and Gradcam tell you? How are they different? Is one better than the other?

Answer (continued):

Differences

Level of Abstraction: Saliency Maps operate at the pixel level, showing the importance of individual pixels in the input. Grad-CAM works at the feature map level (from a convolutional layer), showing which regions or patterns in the image contributed to the decision. It is more spatially interpretable and focuses on larger, more meaningful regions.

Interpretability: Saliency Maps can be harder to interpret visually because they often look noisy, especially in complex images where many pixels contribute small amounts to the prediction. Grad-CAM provides smoother, more human-understandable visualizations by highlighting entire regions or objects rather than specific pixels, making it easier to comprehend the network's attention.

Granularity: Saliency Maps give finer, pixel-level detail but are more difficult to aggregate meaning from. Grad-CAM is less granular but highlights coherent regions, making it more useful for visualizing object-based importance.

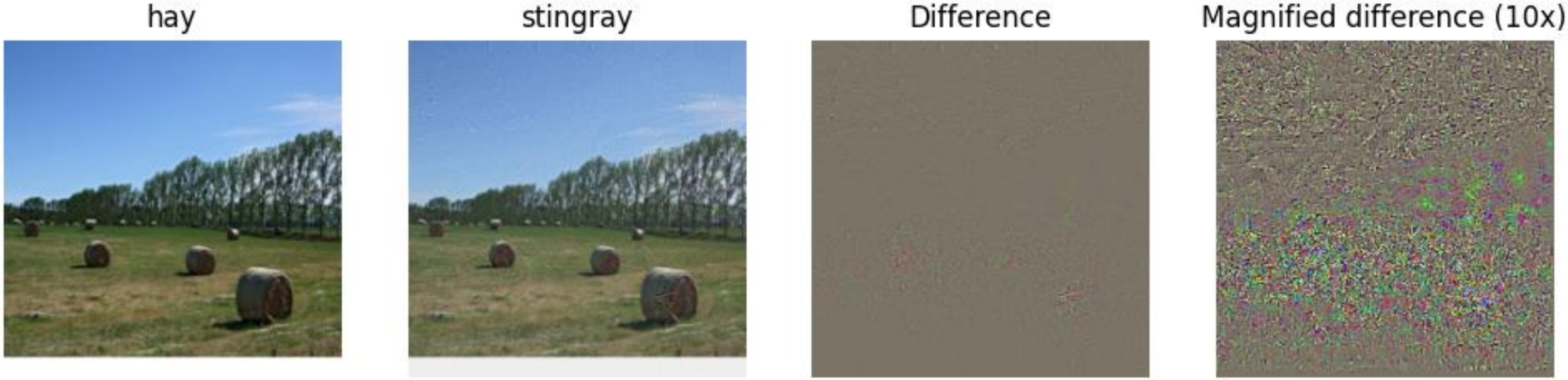
Is One Better Than the Other?

Grad-CAM is generally considered better for interpretability, especially in image-based models, because it produces more visually coherent explanations. It highlights important regions rather than scattered pixels, making it easier to understand why the model made a specific decision. Saliency Maps, while more granular, can be harder to interpret and can sometimes be noisy. However, they provide fine-detail information and can be useful in specific contexts where pixel-level precision is needed.

In summary, Grad-CAM is often more intuitive for most users, but saliency maps can offer deeper insights in certain scenarios, especially when pixel-level information is critical.

Fooling Image

Include the fooling image here:



Fooling Image

What insights do you get from fooling images:

Answer:

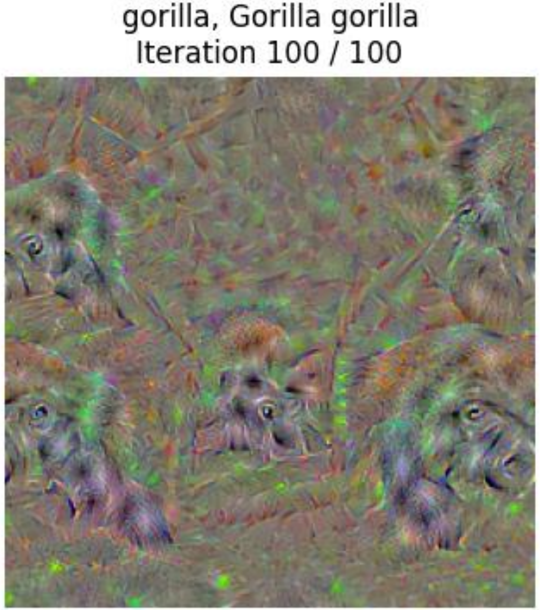
Fooling images, or adversarial examples, provide several important insights about neural networks:

- Vulnerability: Neural networks can be easily tricked by tiny changes in images, showing they might not always make reliable decisions.
- Decision-Making: If small changes lead to big mistakes, it suggests that the network might focus on irrelevant details instead of important features.
- Robustness: Studying how networks respond to these tricky images helps researchers develop ways to make models more resilient and improve their performance.
- Interpretability: Analyzing adversarial examples can reveal how models make decisions, aiding efforts to make AI systems more understandable.
- Ethical Concerns: The ease with which networks can be fooled raises serious ethical issues, especially in critical applications like self-driving cars or medical diagnoses.
- Training Improvements: Insights from fooling images can lead to better training methods, helping models learn to defend against these attacks.

Overall, examining fooling images helps us understand the strengths and weaknesses of neural networks and improve their safety and effectiveness.

Class Visualization

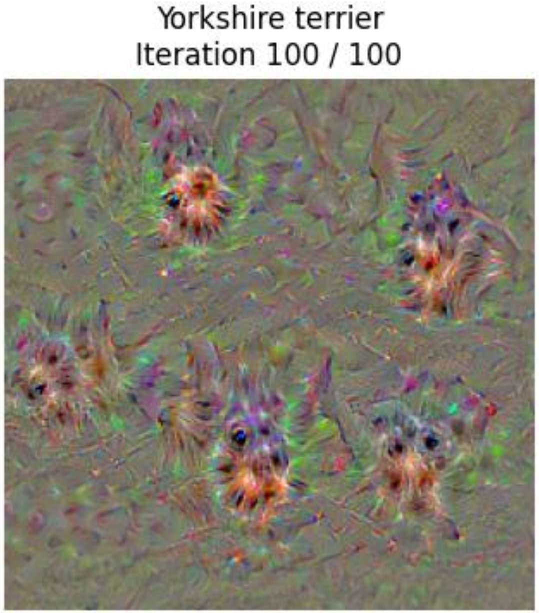
Include class visualization of Gorilla (target_y = 366) here:



gorilla, Gorilla gorilla
Iteration 100 / 100

Class Visualization

Include class visualization of Yorkshire Terrier (target_y = 187) here:



Question: Class Visualization – Use saliency?

In order to find an image that maximizes the correct score, Jane performs gradient ascent on the input image, but instead of the gradient she uses the saliency map in each step to update the image. List and briefly explain two reasons why this is an incorrect approach. (Hint: refer to Section 1.1 of the assignment pdf)

Answer:

Jane's approach of using the saliency map instead of the gradient for updating the image in gradient ascent is flawed for the following reasons:

1.Saliency Map is not the Gradient: Saliency maps are visualizations that highlight which parts of the input influence the model's prediction the most, but they are not the actual gradients with respect to the image. The gradient gives the direction of the steepest ascent (or descent) in the model’s loss landscape, which is crucial for updating the image in gradient ascent. By using the saliency map, Jane is not directly following the gradient of the correct score, leading to suboptimal or incorrect updates to the image.

2.Saliency Maps Lack Precision: Saliency maps are typically coarse approximations or visualizations meant for interpretability, not for precise optimization tasks. They often highlight large regions of the input, but do not provide the fine-grained, exact values that the gradient offers. As a result, using saliency maps for updating the image will likely lead to less effective or noisy updates, making it harder to maximize the correct score efficiently.

Question: Class Visualization – Regularization

DO NOT TAG

When generating an image that the network will recognize as the target class, the quality of the generated image is improved by regularization. In your work, you applied L2-regularization and blurring for this purpose. What is the effect of these on the optimization process (that is, what is it that these techniques are discouraging)?

Please answer on the next slide.

Answer for Class Visualization – Regularization

Answer

L2-regularization and blurring, when applied during image generation to optimize for a target class, help improve the quality of the generated image by discouraging certain unwanted behaviors in the optimization process

L2-regularization:

Effect: It penalizes large pixel values in the generated image, pushing the optimization to produce smoother and more natural-looking images rather than ones with extreme, high-intensity values.
What it discourages: L2-regularization discourages the model from creating images with exaggerated pixel activations, which can lead to noisy or unrealistic images. This helps prevent the generation of images that strongly activate the target class but don't resemble real-world examples.

Blurring:

Effect: Blurring smooths out sharp edges and high-frequency noise in the generated image.
What it discourages: Blurring discourages the model from relying on overly fine details or noise to activate the target class. Instead, it promotes the generation of more coherent, larger-scale patterns in the image, which are typically more representative of real objects.

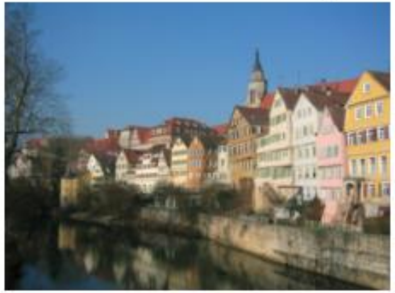
Together, these techniques guide the optimization process toward producing images that are more natural and interpretable by the human eye, while still activating the network's recognition of the target class.

Style Transfer

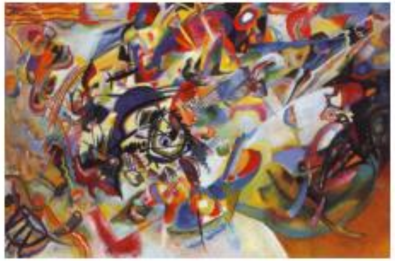
DO NOT TAG

Composition VII + Tubingen

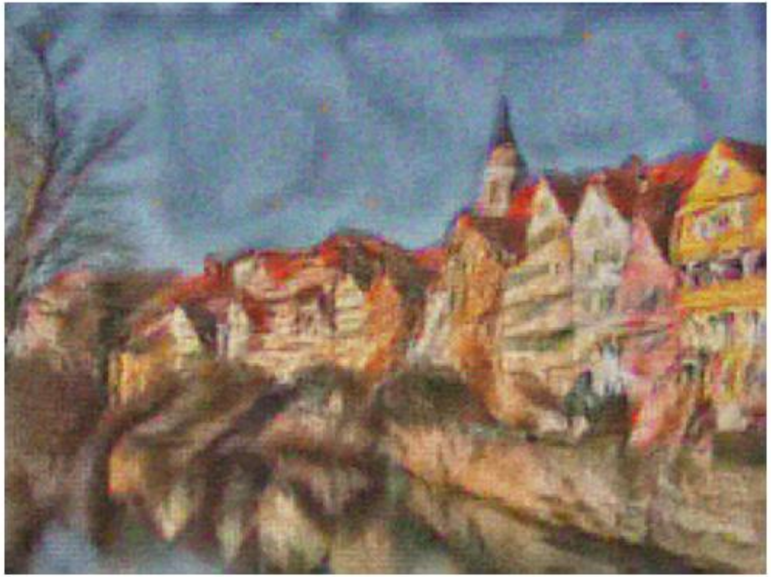
- Include both original images and the transferred image



Content Source Img.

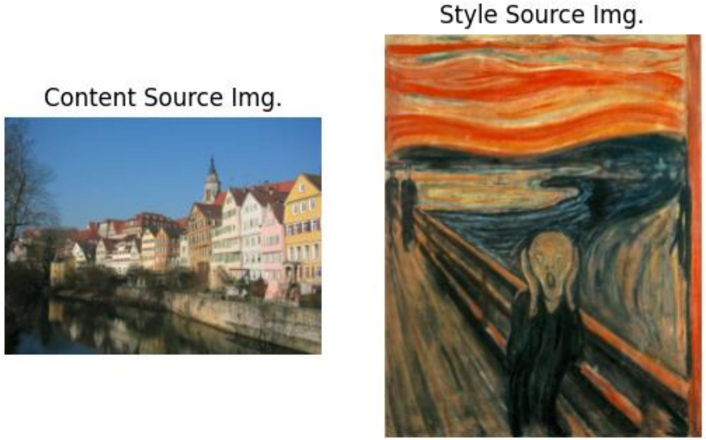


Style Source Img.



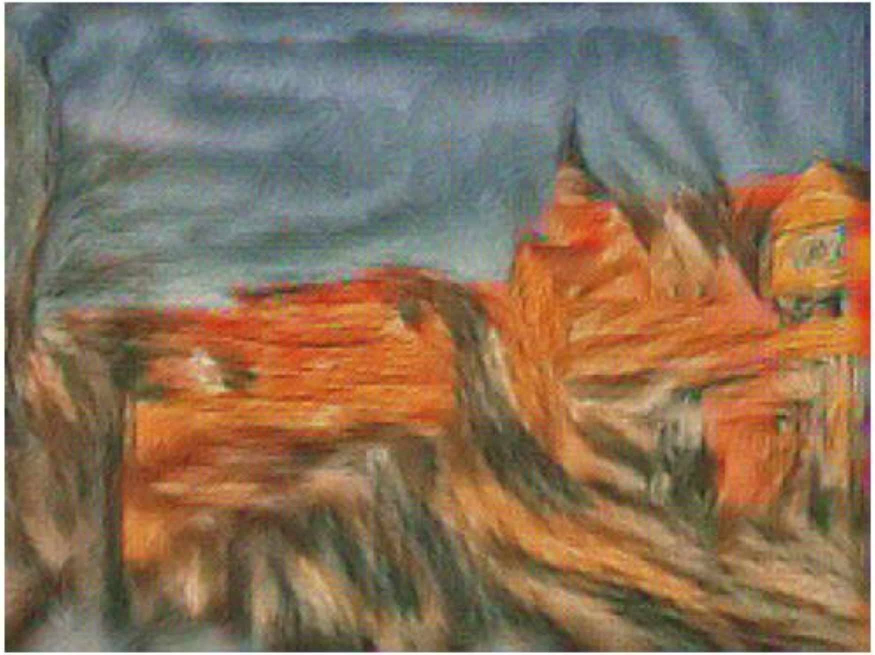
Scream + Tubingen

- Include both original images and the transferred image



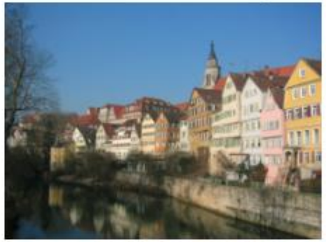
Content Source Img.

Style Source Img.

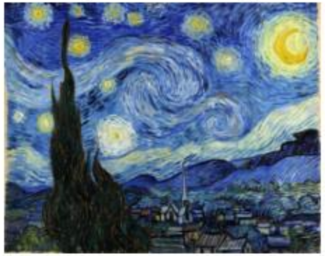


Starry Night + Tubingen

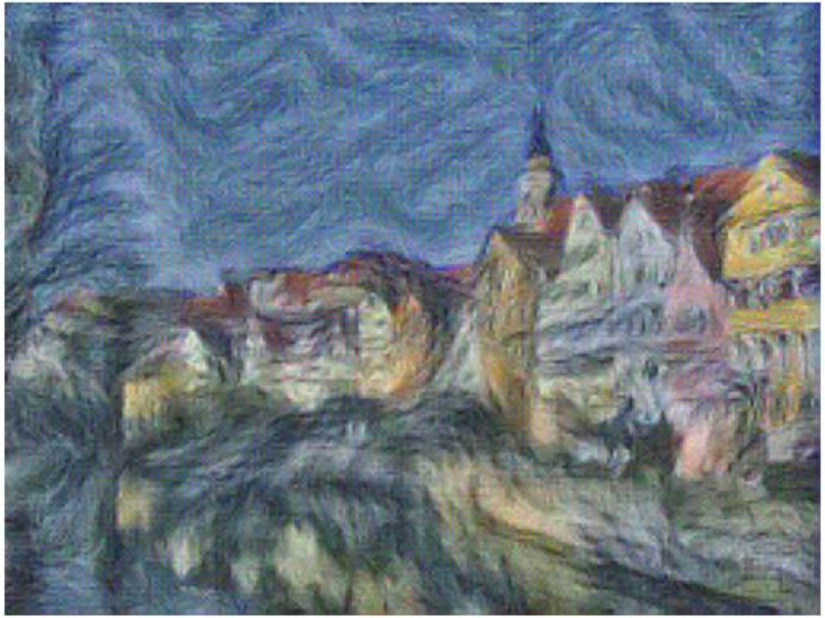
- Include both original images and the transferred image



Content Source Img.



Style Source Img.



Style Transfer – Unleash Your Creativity

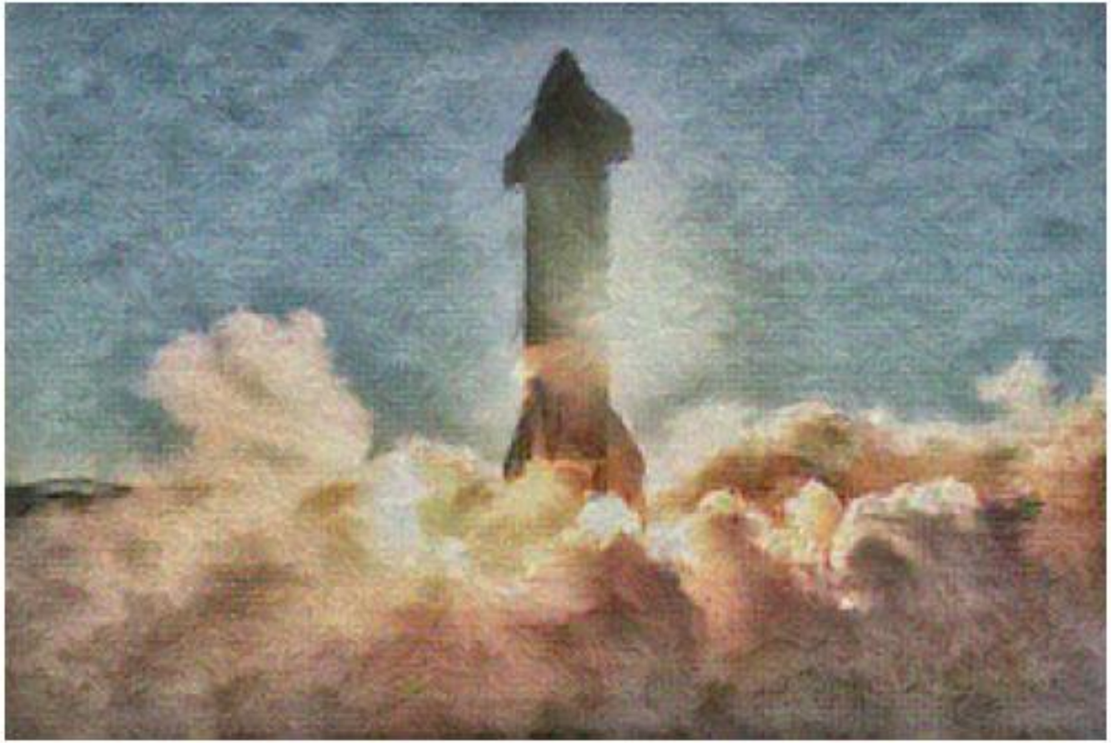
Include your two original images (content and style images)
(Spacex Starship, Monet – Haystacks)

Content Source Img.

Style Source Img.

Style Transfer – Unleash Your Creativity

Include your final stylized image



Assignment 3 Paper Review

DO NOT TAG

Provide a short preview of the paper of your choice.
I chose: “Sanity Checks for Saliency Maps" by Julius Adebayo et al.

The paper *Sanity Checks for Saliency Maps* by Adebayo et al. (NeurIPS 2018) critically evaluates the effectiveness of saliency maps as interpretability tools, especially in the context of convolutional neural networks (CNNs). In deep learning, particularly in image classification tasks using CNNs, saliency maps are commonly used to highlight which parts of an image contribute most to a network's prediction. The assumption is that these visual explanations correlate strongly with the decision-making processes of the model. However, Adebayo et al. challenge this assumption by introducing simple yet powerful tests to assess whether these maps provide meaningful insights.

Saliency maps are important in the realm of CNNs because CNNs process high-dimensional input data like images, where the complex patterns learned by the network are difficult to interpret. By visualizing which pixels or regions of an image contribute most to a CNN’s output, saliency maps attempt to offer transparency. Methods such as gradient-based saliency, such as vanilla gradients, Guided Backpropagation, or Integrated Gradients, create visualizations by backpropagating the gradient of the output with respect to the input image. However, despite their popularity, their reliability has not always been thoroughly scrutinized.

The authors of the paper perform several sanity checks to evaluate whether saliency maps truly reflect the learned representations of CNNs. Specifically, they investigate if the maps change when the underlying model is changed, including scenarios where the model is trained or where its weights are completely randomized.

Provide a short preview of the paper of your choice (continued)
I chose: “Sanity Checks for Saliency Maps" by Julius Adebayo et al.

Main Contributions

Randomization Tests for Model Weights: The authors propose a straightforward sanity check: reset the CNN's weights to random values and observe whether the saliency maps generated afterward differ significantly from those produced by the original, trained model. Shockingly, for many popular saliency methods, the maps do not change substantially even when the model is completely randomized, raising concerns about their reliability.

Randomization of Labels: The paper also introduces a test where the labels for the training data are randomized. If a saliency method is robust, the resulting maps should change significantly when the network is trained on random labels, as the relationship between the image and its class is effectively broken. Again, some saliency methods fail this test.

Application Across Architectures: The paper explores the behavior of saliency methods across different CNN architectures, including ResNet and VGG. The findings suggest that some saliency maps are insensitive to model architecture, implying that they may not be representing the network’s internal decisions accurately.

These contributions reveal that some saliency maps may not be trustworthy explanations of a model's decision-making process, since they can remain consistent even when the model itself is fundamentally altered.

Image Fooling and Saliency Maps

Image fooling is closely related to saliency maps. In image fooling, small perturbations or adversarial noise are applied to input images, causing CNNs to misclassify them. The perturbations are often imperceptible to humans but drastically affect the network’s output. Saliency maps should, in theory, highlight these perturbations. However, if the maps remain unaffected even when the model is fooled, their utility as interpretability tools is questionable. The paper’s randomization tests indirectly touch on this, suggesting that saliency maps may not be sensitive to critical model behaviors like adversarial robustness.

Style Transfer and Saliency Maps

Saliency maps are also of interest in style transfer, where the goal is to change the artistic style of an image while preserving its content. CNNs are leveraged in style transfer tasks to extract content and style features, and saliency maps could, in theory, help understand how the network separates these aspects. However, given the paper’s findings, if saliency maps do not meaningfully change across different architectures or even after weight randomization, their use in style transfer becomes less clear. The maps may not accurately reflect the network's separation of style and content, reducing their interpretability value in these applications.

Provide a short preview of the paper of your choice (continued)
I chose: “Sanity Checks for Saliency Maps" by Julius Adebayo et al.

Summary and Observations

The main contributions of the paper are its thorough critique of the reliability of saliency maps, especially under basic randomization tests. The work demonstrates that certain saliency methods do not reflect the true decision-making processes of CNNs, as the maps remain largely unchanged even when models are randomized. This undermines their effectiveness as interpretability tools and calls into question their use in real-world applications, where interpretability is crucial for ensuring model trustworthiness.

Personal Takeaways

The paper offers a compelling argument that saliency maps, while visually appealing, may not provide trustworthy explanations unless rigorously tested. I entered the paper after finishing the assignment and was excited to see how helpful they appeared to be as they are very visually intuitive, but after reading the paper, this excitement may be overblown. One of my key takeaways is the need for caution when using such methods for model interpretability. It is easy to assume that because a saliency map visually highlights image regions, it must reflect the underlying logic of the model. However, the results from this paper emphasize the importance of verifying these assumptions through sanity checks.

Moreover, this paper adds to a growing body of work highlighting the challenges of interpretability in deep learning. As models grow more complex, the need for robust interpretability tools becomes more urgent. The insights from this paper push researchers to explore more reliable methods and to always critically evaluate the tools we rely on for understanding CNNs.