4-28-2023

Investigating the Impact of Baselines on Integrated Gradients for Explainable AI

Ajay Shewale
as1763@rit.edu

Investigating the Impact of Baselines on Integrated Gradients for Explainable AI

by

Ajay Shewale

A thesis submitted in partial fulfillment of
the requirements for the degree of
Master of Science in Software Engineering

Department of Software Engineering
B. Thomas Golisano College of Computing and Information Sciences

Rochester Institute of Technology
Rochester, New York
April 28th, 2023

Abstract

Deep Neural Networks have rapidly developed over the last few years, demonstrating state-of-the-art performances on various machine learning tasks such as image classification, natural language processing, and speech recognition. Despite their remarkable performance, deep neural networks are often criticized for their need for more interpretability, which makes it difficult to comprehend their decision-making process and get insights into their workings. Explainable AI has emerged as an important area of study that aims to overcome this issue by providing understandable explanations for deep neural network predictions. In this thesis, we focus on one of the explainability methods called Integrated Gradients (IG) and propose a contour-based analysis method for assessing the faithfulness of the IG algorithm.

Our experiments on the IG algorithm showcase that it is an effective technique for generating attributions for deep neural networks. We found that the IG algorithm effectively generated attributions consistent with human intuition, highlighting relevant regions of the input images. However, there are still significant issues with the performance and interpretability of IG. For example, choosing the correct baselines for computing IG attributions is still important. The baseline in this context refers to the lack of features, which is used as a starting point to get the attributions. To address this issue, we assessed the performance of the IG algorithm by using multiple random baselines and aggregating the resulting attributions using mean and median techniques to obtain the final attribution.

To evaluate the aggregated attributions, we propose a contour-based analysis method. This method provides an important continuous patch of aggregated IG attribution's top 10% values. The continuous patch of important features allows us a more intuitive interpretation of IG's performance. We use the Captum library to implement the IG algorithm and experiment with multiple random baselines to compare the attributions generated by the IG algorithm. Our results demonstrate that the contour-based analysis method can be used to evaluate

the performance of the IG algorithm for different baselines and can be applied to other attribution algorithms as well. Our findings suggest that the IG algorithm can identify the most critical elements of an image, and the contour-based approach can extract more localized and detailed information.

Our research sheds light on the effectiveness of the multiple random baselines on the Integrated Gradients (IG) algorithm. It provides valuable insights into its performance when generating attributions for deep neural networks with different baselines. We also identify several limitations of our study, such as focusing on a single model architecture and data type and using a perturbation-based method to create random baselines. Future work can address these limitations by evaluating the performance of IG on other types of models and data using different ways to create the baselines.

Acknowledgements

This work was done under the mentorship of Dr. Nidhi Rastogi, Principal Investigator, and committee members Dr. Mohammed Wiem Mkaouer, Dr. Daniel Krutz, and Michael Clifford. I want to thank all of them for their time and guidance through the thesis process. I thank my research advisor and committee members for their valuable advice and support.

I want to acknowledge the contribution of the data sources used in this research. I appreciate the open availability of these datasets, which allowed us to conduct our research.

Finally, we would like to thank our colleagues and friends who provided us with their support, encouragement, and helpful discussions during this project. Their insights and suggestions were valuable in shaping our ideas and improving the quality of our work.

I would like to dedicate this work to the memory of my deceased brother Himalaya Shewale, who was taken from us too soon. He was not only my brother but also my closest friend and a source of inspiration in my life. He instilled in me the value of hard work, dedication, and perseverance. I would also like to honor his memory by acknowledging his beautiful and loving daughter, my niece Jijau Shewale, who continues to thrive and grow. Though he is no longer with us, his legacy lives on through her, and I know he would be so proud of the amazing girl she is becoming.

Committee Approval

Nidhi Rastogi — Date

Thesis Advisor

Mohamed Wiem Mkaouer — Date

Committee Member

Daniel Krutz — Date

Committee Member

Michael Clifford — Date

Committee Member

Contents

1 Introduction

Machine learning models have become very popular in recent years due to their accurate insights into the data and automation of complex tasks. For example, machine learning models in healthcare performed better than humans in detecting diseases such as skin cancer [1]. In finance, machine learning models have been used to detect credit card fraud by analyzing large amounts of data [2].

Even with many applications of machine learning models, often, these models are difficult to interpret, making it challenging for humans to understand how machine learning models make predictions. In domains such as healthcare, finance, or autonomous driving, this lack of interpretability can be a critical problem because the consequences of model errors can be severe. For example, interpretability is crucial in the healthcare industry to ensure that model decisions are consistent with medical knowledge and ethics [3]. To identify potential biases and stop unfair behaviors, it is crucial in finance to understand the reasoning behind a model's predictions [4]. Similarly, interpretability is required in autonomous driving to ensure that the model's choices comply with safety standards and laws [5].

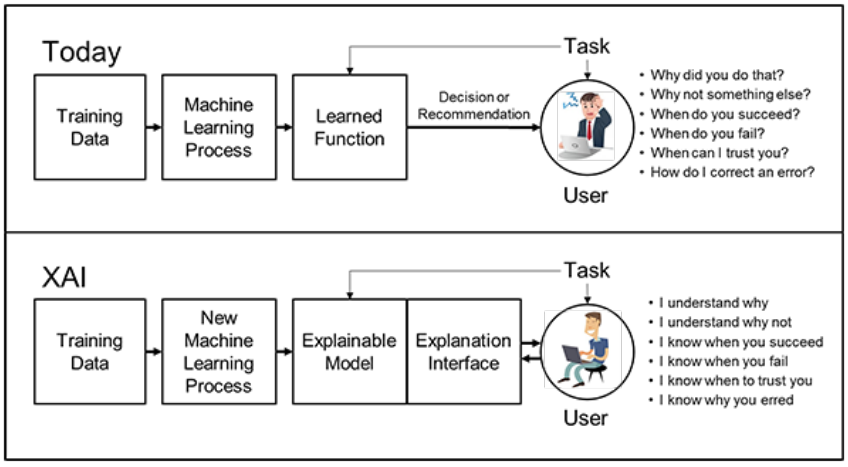


Figure 1: XAI concepts [6]

As a result of these challenges, a new area of research called Explainable Artificial Intelligence (XAI) has emerged [7]. XAI is a rapidly growing field of research that aims to develop methods for creating more transparent and interpretable machine learning models [8]. The growing interest in XAI has resulted in the development of several techniques, such as local explanation methods, global explanation methods, and feature visualizations, for providing insights into how machine learning models make decisions [7]. Figure 1 demonstrates the core concepts of how XAI works.

Post-hoc explainability methods are popular for understanding black box models [9]. Figure 2 illustrates the overview of Post-hoc explainability methods.

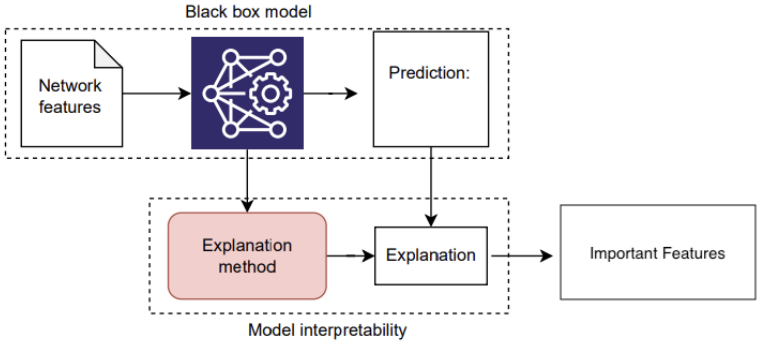


Figure 2: Overview of Post-hoc Explainability methods

This study aims to systematically evaluate Integrated Gradient (IG), a popular post-hoc explainability method, and propose potential improvements.

Integrated Gradient(IG) is an XAI technique that attributes the prediction for deep neural networks to its inputs [10]. IG computes the attribution of each input feature to the output, considering the output gradient concerning the input along a path from a baseline to the input [10].

Nevertheless, there are still significant challenges with the performance and interpretability of IG [11]. One of the significant problems is choosing which baseline to use to compute attributions, which can considerably impact the at-

tributions [10]. The baseline here refers absence of the feature, which is used to get the attribution by accumulating gradients on the images interpolated between the baseline and input image [12]. Another challenge is the faithfulness of IG, which refers to the degree to which the attributions reflect the actual contribution of each feature to the output. Low faithfulness can lead to misleading explanations, making it difficult for humans to understand how the model makes its decisions [13].

Our research aims to improve the understanding of Machine Learning models by making IG attributions more robust. This research analyzes and evaluates the impact of using multiple random baselines for the Integrated Gradient method. Additionally, we propose the Contour Analysis method used to evaluate our IG explanations.

1.1 Research Questions

RQ1: How does the choice of baseline impact the performance of IG?

RQ2: How can we analyze the faithfulness of IG to identify potential issues with the explanations provided?

RQ3: How does the Contour Analysis method affect the IG attributions?

1.2 Motivation

The motivation for this research arises from the growing concerns about the lack of transparency and interpretability in Machine Learning models, particularly in high-stakes applications such as medical diagnosis and autonomous driving [14].

The proposed research will contribute to the ongoing development of XAI by conducting a thorough analysis of the performance and interpretability of IG. This research will build on P. Sturmfels and S. Lundberg's [12] work, which proposed several ways to choose the baseline for IG.

Additionally, this research paper has broader implications for the ethical and governance issues surrounding the development and deployment of AI systems.

The lack of transparency and interpretability in machine learning models has raised concerns about their potential to perpetuate biases and discrimination, as noted by Floridi et al. [15]. By improving the interpretability and transparency of machine learning models; our research has the potential to address these concerns and contribute to the development of more reliable and responsible AI systems.

1.3 Scope

This study aims to thoroughly analyze the Integrated Gradients algorithm for interpreting deep neural networks. The analysis will focus on the performance and interpretability of the algorithm using different baselines. Also, we proposed the Contour Analysis method, which is used to extract continuous patch of important features using the IG attributions.

The research is limited to image classification tasks. We used Captum Library [16] for IG experimentations and Quantus Library [17] to evaluate explanations quantitatively. The study evaluates the performance of IG using various baselines and aggregation methods based on mean and median. The research proposed a new contour analysis method to assess the IG algorithm's faithfulness, which involves selecting the top 10% of the attribution values and drawing a contour around them. The resultant patch from the Contour Analysis method is fed into a ResNet-50 [18] model to assess the faithfulness of the IG algorithm for different baselines.

The proposed research builds upon the foundation laid by previous researchers who have explored the performance and interpretability of the Integrated Gradient (IG) algorithm. Sundararajan et al. [10] introduced the IG algorithm as a method for explainable AI. Adebayo et al. [11] proposed sanity checks for saliency maps, including IG, to accurately reflect the model's decision-making process. The proposed research extends this work by analyzing the performance of IG using different baselines and proposing a new method for contour analysis to assess the faithfulness of the IG algorithm.

2 Literature Review and Related Work

In this section, we go over the relevant literature in our domain and work related to the work done in this thesis.

2.1 Overview of Explainable AI

Explainable Artificial Intelligence (XAI) is a subfield of machine learning that focuses on developing algorithms and techniques that provide human-understandable explanations for the outputs generated by machine learning models. The importance of XAI has been recognized in various industries, such as healthcare, finance, and autonomous driving, where the consequences of model errors can be severe [8].

Explainability and Interpretability: The terms interpretability and explainability are often used interchangeably in research. Doshi-Valez and Kim define interpretability as the ability to explain or to present in understandable terms to a human [19]. Miller defines it as the degree to which a human can understand the cause of a decision [20]. On the other hand, explainability is associated with the internal logic and mechanics inside a machine learning system [21]. Having more explainability in the model, we can deeply understand the procedures while the model trains or makes decisions. Achieving high performance with interpretable models can be difficult, where complex deep learning models outperform simpler models [22].

2.2 Importance of Explainable AI

This section discusses the importance of XAI in various applications and how it can benefit us in different ways.

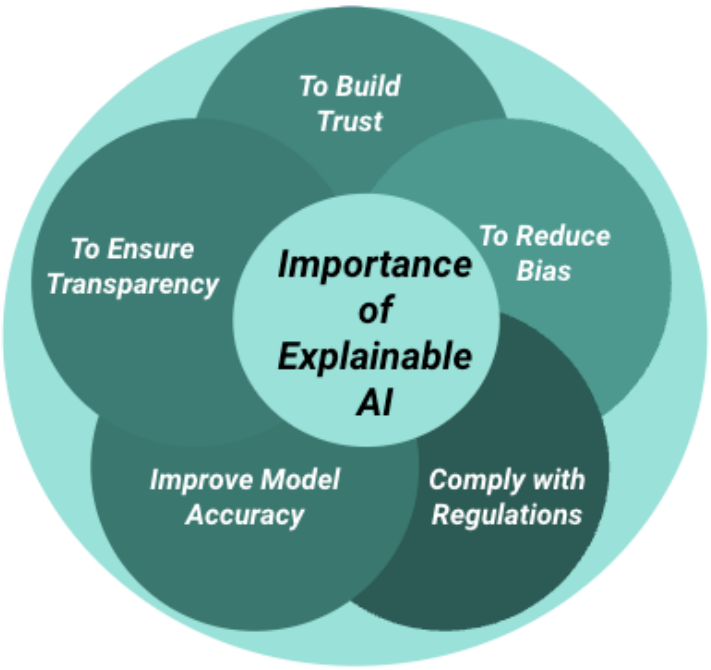


Figure 3: Importance of Explainable AI in Decision-Making

1. To Ensure Transparency: Explainable AI helps us to understand how the model makes decisions and to ensure transparency in the decision-making process ([7], [23]).

2. To Build Trust: Understanding how AI systems arrive at a decision can help build trust between the user and the system [8].

3. To Reduce Bias: By understanding the features driving the decision-making process, we can identify and remove any potential biases that may be present [24].

4. To Comply with Regulations: Some industries, such as healthcare and finance, require transparency and accountability in the decision-making process [25].

5. To Improve Model Accuracy: By understanding which features are most important to the model's decision, we can identify areas for improvement and fine-tune the model to improve accuracy ([7], [26]).

2.3 Classification of Post-hoc explanations methods

The classification is based on four categories of problems, and the explanation methods are classified according to the problem they can solve [27]. Post-hoc explanations methods can be classified into the following categories [22]:

(a) **The granularity of explanations (local vs. global) [23]**:

According to W James Murdoch, there are two types of explanation methods: local and global. Global explanations offer a comprehensive understanding of the model's behavior over multiple instances by identifying the most crucial features in making decisions, enabling the assessment of biases and strengths, and guiding improvements [22]. On the other hand, local explanations explain the model's decision-making process for a particular instance by identifying input features that significantly contribute to the model output. [22].

(b) **Supported model(Model agnostic vs. model specific) [23]**:

Another crucial distinction is between model-specific and model-agnostic explanation methods[[28], [27], [29], [30], [31]]. Model-agnostic explanation methods can be applied to any machine-learning model regardless of its architecture. Examples of model-agnostic methods include LIME [32], SHAP [33], and Integrated Gradients [10].

On the other hand, model-specific explanation methods are designed to work with a specific type of model architecture, and their interpretability may depend on the model's specific structure. Examples of model-specific methods include GradCAM [34], Guided Backpropagation [35], and Occlusion Sensitivity [36].

(c) **Type of explanation(feature attribution, rules, and counterfactuals) [23]**:

The type of explanation can be divided into three parts. First, the feature attribution method assigns a relevance score to each feature to determine its importance in the model's prediction[6]. Feature attribution-based explanation methods are popular post-hoc explanation techniques due to their ability to identify crucial features responsible for a model prediction [6]. Some examples of feature attribution-based explanation methods are LIME [32], SHAP [33], Integrated Gradients [10], and DeepLIFT [37]. Rule-based explanations are commonly used for tabular data and provide a decision rule of the form x $\rightarrow$ y, where x represents conditions on input features. Y represents the model prediction [22]. ANCHOR is an example of a rule-based explanation method that uses a game-theoretic approach to identify the smallest subset of input features that can determine the model prediction [38] Counterfactual-based explanations: This type of explanation method tries to find the closest instance of opposite prediction where the difference in feature distribution of the two samples provides explanations for the model prediction [22]. Counterfactual explanations are valuable in cases where the model prediction does not align with the user's expectation. Providing an alternative scenario that would have led to a different prediction is necessary. One of the popular counterfactual methods is the "What-If Tool" [39], which allows users to explore different scenarios and understand how the model behaves under various conditions.

2.4 Integrated Gradient

Integrated Gradient is a commonly used method for explaining deep neural networks and other differentiable models [25]. Integrated Gradient is based on two axioms: Sensitivity and Implementation invariance [40].

2.4.1 Definitions

Axiom Sensitivity: An attribution method satisfies Sensitivity. If there is a difference in one feature between every input and baseline with different predictions, then the differing feature should be given a non-zero attribution [41]. Simply put, non-zero attributions are given to every input and baseline that differ in one feature but have different predictions [25].

Axiom Implementation invariance: refers to the principle that if two models are functionally equivalent or behave identically, their attributions should also be identical [25].

The sensitivity axiom in IG uses a baseline [25]. A baseline can be defined as the absence of a feature in an input [41]. Another baseline definition can be "input from the input space that produces a neutral prediction" [41].

2.4.2 Intuition behind Integrated Gradient

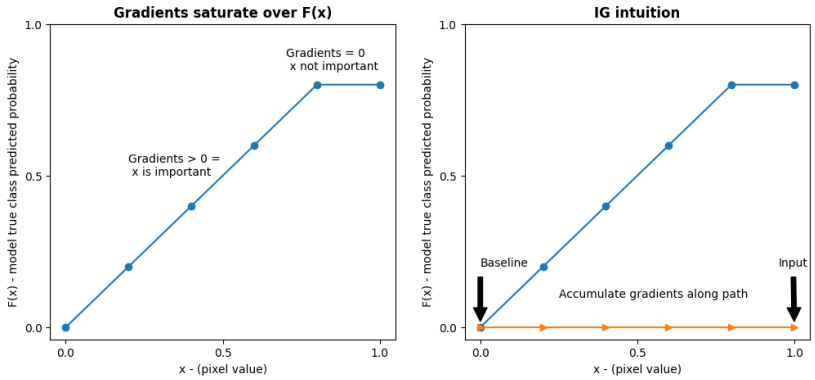


Figure 4: Pixel x Importance Analysis using Simple Gradients and Integrated Gradients [42]

Left: Gradients saturate over F(x); we can see that the model's gradient for pixel x is positive between 0.0 to 0.8; however, pixel x plays a crucial role in driving the model toward a predicted probability of 80%. Does it make sense that a pixel x's importance is minor or discontinuous? [42]

Right: The IG method aims to accumulate the local gradients of pixel x and measure its significance as a score for how much it contributes or subtracts to the model's overall output class probability [42].

2.4.3 How is Integrated Gradient calculated?

In the IG definition [10], function $F : R^n \rightarrow [0, 1]$ represents the deep network, an input $x \in R^n$, and a baseline $x' \in R^n$. We consider the straight-line path from baseline to the input x and accumulate the gradients along that path. The Integrated Gradient along the i^{th} dimension is defined as:

$$IntegratedGradients_i(x) ::= (x_i - x'_i) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha \quad (1)$$

Computing a definite integral numerically and computationally expensive may not always be possible. Therefore, implementation involves computing the following approximation:

$$IntegratedGrads_i^{approx}(x) ::= (x_i - x'_i) \times \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m} \times (x - x'))}{\partial x_i} \times \frac{1}{m} \quad (2)$$

Where m is the number of steps in the Riemann sum approximation of the integral.

Figure 5 represents the five steps interpolated images along a linear path between a black baseline image and the example "Traffic Signal" image. Figure 6 shows the results of the IG.

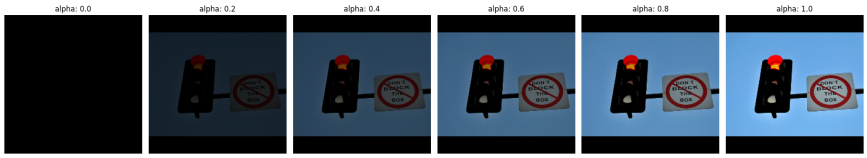


Figure 5: Visualisation of interpolated images along a linear path

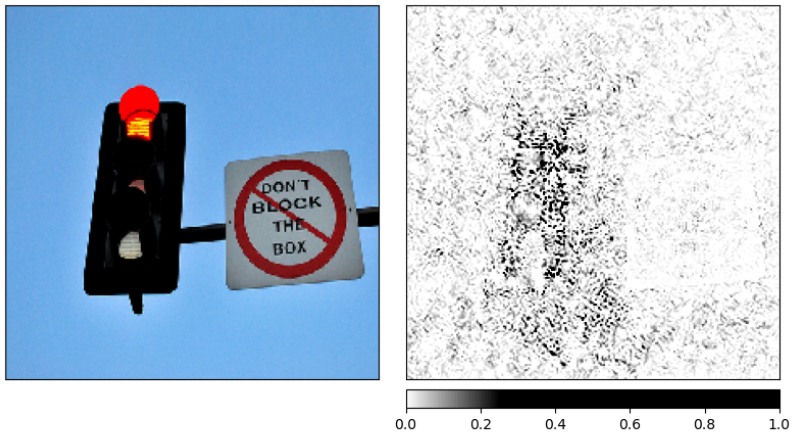


Figure 6: Visualisation of IG attributions on Traffic Signal image

2.5 Prior studies on IG baselines and their challenges

Integrated gradients use a baseline input to model the absence of a feature. Choosing an appropriate baseline is crucial. It is common practice to set the baseline input as a vector of all zeros, but the best choice for representing the absence of a feature is still a matter of debate [12]. For example, using a black image as a baseline may not accurately highlight the important features if the image contains black pixels. The IG does not account for the color used as a baseline input, leading to potential blindness to certain features. Several studies have been proposed for choosing the baseline for the IG method.

(a) **The maximum distance baseline:** Take a baseline with the farthest image L1 distance from the input image [12]. The problem with the maximum distance baseline lies in its inability to represent missingness effectively, as it retains information regarding the input image.

(b) **The blurred baseline:** Fong and Vedaldi's [43] proposed using the

blurred baseline image to represent the absence of information. This approach is appealing as it provides a very intuitive representation of missing information in images. A potential limitation of the blurred baseline is its tendency to emphasize high-frequency information, potentially reducing the importance of pixels similar to their neighbors. This is due to the baseline's use of a weighted average of a pixel and its surrounding neighbors, leading to the unequal weighting of similar and dissimilar pixels [37].

(c) **The Uniform Baseline:** creating a baseline by randomly sampling an image from the range of valid pixels using a uniform distribution [43].

(d) **The Gaussian Baseline:** Smilkov et al. [14] proposed using the Gaussian distribution on the input image to create the baseline.

A uniform random image used as a baseline can suffer from the same blindness issue as a constant image. This is because some baseline pixels might be too close in value to their corresponding input pixels and not be highlighted as important, leading to artifacts in the resulting saliency map [37]. One solution to this issue is to average the results over multiple different baselines, as suggested in previous works ([39], [44], [45]). We can obtain a more robust saliency map by drawing multiple samples from the same distribution and averaging the importance scores [37].

The proposed research builds on this related work by conducting a thorough analysis of the performance and interpretability of IG, focusing on evaluating the impact of different baselines on the performance of IG.

3 Proposed Approach and Evaluation Criteria

3.1 Proposed Approach

The proposed approach involves a series of steps to evaluate the Integrated Gradients (IG) algorithm's performance and interpretability. Firstly, we collected and pre-processed the images. Then, we computed the IG attributions for multiple baselines to evaluate the IG attributions. Next, we aggregated the IG attributions based on mean and median and assessed their impact on the model's performance.

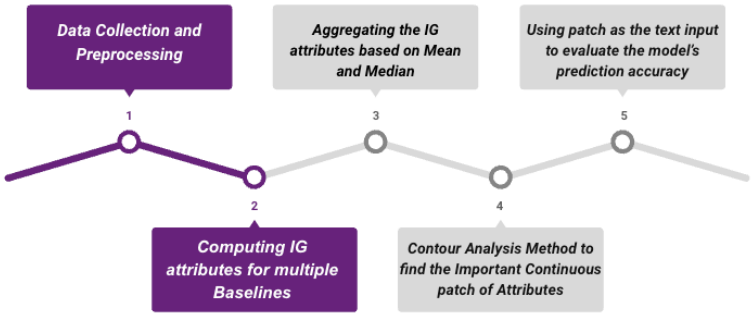


Figure 7: Overview of proposed approach

We utilized the Contour Analysis method to identify the significant continuous patch of attributions, which enabled us to visualize and extract the most relevant features contributing to the model's prediction. Finally, we used this patch as input to evaluate the model's prediction accuracy, providing a visual explanation of the model's decision-making process. Following are the detailed steps of our proposed approach.

3.1.1 Data Collection and Preparation

Data collection and preparation are integral to Machine Learning. This section describes our data collection and preparation process, which includes getting

and normalizing the input images for our analysis.

Dataset: We collected image data from the ImageNet database [46], a widely used dataset for image classification tasks. The ImageNet dataset contains over 14 million images classified into 1000 categories. We selected a subset of the ImageNet dataset for our experiments, consisting of images in the **Traffic Light** category [46]. We chose this category because it contains diverse traffic images, allowing us to test our algorithm's robustness across different car types.

Deep Learning Model: We used the ResNet-50 model [47], a deep convolutional neural network pre-trained on the ImageNet dataset. This model has achieved state-of-the-art performance on various computer vision tasks, including image classification [47]. To prepare our data, we first collected images from the ImageNet dataset [46]. We then preprocessed each image by resizing it to 224x224 pixels and normalizing the pixel values using the same mean and standard deviation values used to train the ResNet-50 model. The preprocessing of the images ensures that the input images are in the same format as the ResNet-50 model.

Next, we selected a set of target classes from the ImageNet dataset and manually verified that each image in our dataset belonged to one of these target classes. We also ensured that the target class labels were consistent with those in the ImageNet dataset. Our data collection and preparation process was designed to ensure that our experiments were conducted on a well-curated dataset that is representative of the types of images that the ResNet-50 model was trained on while also ensuring that our target classes are consistent with the ImageNet dataset.

3.1.2 Experimentation using Captum Library

To analyze the Integrated Gradients algorithm and compare it with other baselines, we used the Captum library [16] for experimentation. Captum provides a suite of attribution algorithms that can be used to explain the predictions of a machine-learning model. In our study, we used the Integrated Gradients

algorithm provided by Captum to compute the attribution scores of the input features for a given target output class.

First, we preprocessed the input images using the *preprocess_input* function provided by the Keras module [48]. This function scales the image's pixel values to be in the range of [-1, 1] and applies mean subtraction.

Then we used the ResNet-50 model [47] of Keras Library [48] as our machine learning model. ResNet-50 is a deep convolutional neural network architecture effective for image classification tasks. We used the pre-trained ResNet-50 model in the Keras module, which has been trained on the ImageNet dataset.

We used the *load_img* and *img_to_array* functions provided by Keras utility functions to load the input image and convert it to a NumPy array. We then used the $predict\_fn$ function to obtain the predicted class label and probability for the input image using the ResNet-50 model.

To compute the attribution scores of the input features, we used the IntegratedGradients class provided by the Captum library. We passed the preprocessed input image and the target output class as input to the attribute method of the IntegratedGradients class, which computed the attribution scores of the input features using the Integrated Gradients algorithm.

We used the *get_random_baseline* function to generate random baselines. We used the attribute method of the IntegratedGradients class to compute the attribution scores for each baseline and aggregated the results using either the mean or median method. We also used the $get\_important\_features$ function to extract the most important features of the input image based on the attribution scores. Finally, we saved the attribution maps and important features as images in a specified directory.

The experimentation using the Captum library allowed us to analyze the performance of the Integrated Gradients algorithm and compare it with other baselines. It also gave us visualizations of the attribution maps and important features, which helped us interpret the results.

3.1.3 Contour Analysis Approach

IG attributions are pixel-wise, making evaluating the faithfulness of the model's predictions based on the attributions challenging. For example, suppose we get the ResNet-50 models prediction on the IG attributions. In that case, we will not get the correct results because of the pixelated attributions, as shown in Figure 8.

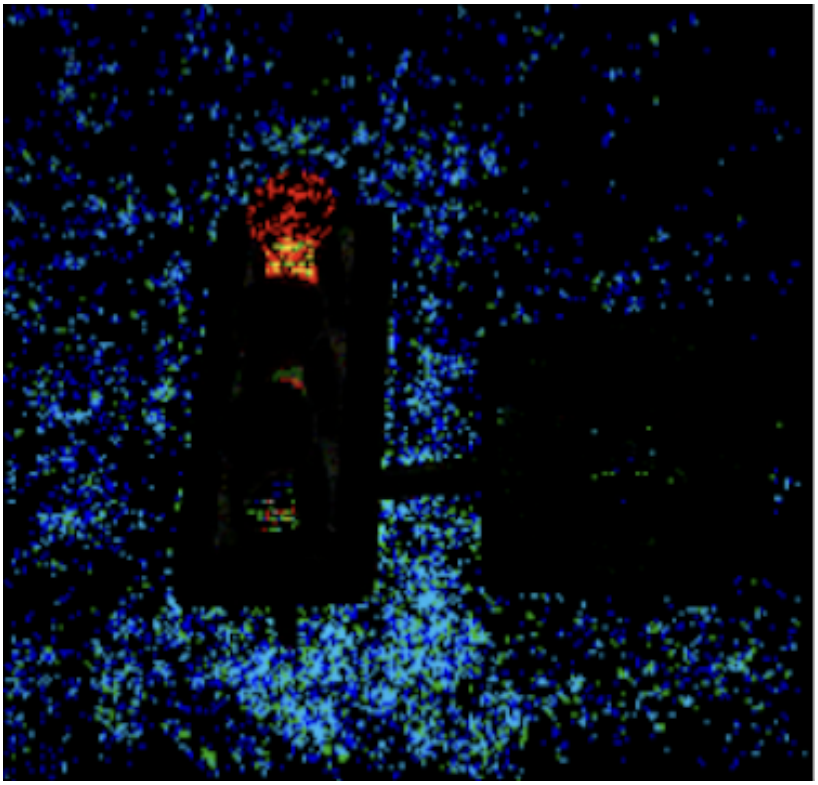


Figure 8: Pixelated IG attributions

We proposed the Contour Analysis method to address the challenge of obtaining a continuous patch of Integrated Gradients attribution values. The algorithm is as follows:

Algorithm 1 Contour Analysis Algorithm

1: Input the integrated gradients attribution, input image, and threshold value.
2: Apply a morphological opening operation to the binary mask to remove small noise and fill gaps between regions.
3: Apply a morphological dilation to the binary mask to merge nearby regions.
4: Find all contours and combine all the contours to get the outer points
5: Find the convex hull of all the contours
6: Draw the convex hull on the mask of the input image and extract the patch from the input image

3.1.4 Using patch as the text input

Using a patch as a test input involves selecting a region of an image and evaluating the model's prediction accuracy based on that region. This approach can be useful for evaluating the faithfulness of a model([49], [50]).

Algorithm 2 Evaluate the prediction accuracy using a patch

1: Select a region of interest in an input image.
2: Extract the selected region as a patch using the Contour analysis.
3: Feed the patch as input to the deep learning model and obtain the predicted class label and probability using the $predict\_fn$ function.
4: By evaluating prediction accuracy using patches, one can gain insights into the model's faithfulness.
5: Using patches as test inputs can be useful for evaluating the performance and interpretability of deep learning-based computer vision models.

3.2 Evaluation Methodologies

Our evaluation is comprised of two methods. First, we proposed our Contour Analysis method to evaluate the faithfulness of the explanations for different baselines. Second, we evaluated the explanations on Functionally grounded quantitative analysis.

3.2.1 Using the Contour Analysis method

The contour analysis method's evaluation criteria were developed to identify important continuous patches of the input image using the proposed contour analysis method and evaluate the prediction accuracy for different baselines.

Our assumption: This approach is based on the assumption that the important features of the input image that are relevant to the model's prediction are likely to be captured by the IG attributions and can be identified using the contour analysis method.

Why: The assumption that important features of the input image can be identified using the contour analysis method is also supported by the work of Hooker et al. [51] and Adebayo et al. [11]. They demonstrated that the IG attributions could help identify important regions of an image relevant to the model's prediction.

The three steps involved in the evaluation criteria using the contour analysis method are:

(a) Identify the important continuous patch of the input image using our proposed Contour Analysis method.

(b) Get the prediction for important patches using the ResNet-50 model.

(c) Compare the prediction accuracy for different numbers of baselines.

Step 1 involves applying the contour analysis method to the IG attributions to identify the important continuous patch of the input image. Step 2 involves obtaining the model's prediction for the identified important patches, and Step 3 involves comparing the prediction accuracy for different numbers of baselines.

One limitation of this approach is that it assumes that the important features of the input image relevant to the model's prediction are likely to be captured by the IG attributions, which may only sometimes be the case. Additionally, the contour analysis method may only accurately identify the most relevant regions of the input image, as it relies on a set of image processing operations that may only sometimes be optimal for a given attribution.

3.2.2 Using Quantitative Analysis

The XAI metrics most often belong to the categories shown in figure 9, which can further be classified into subcategories [17].

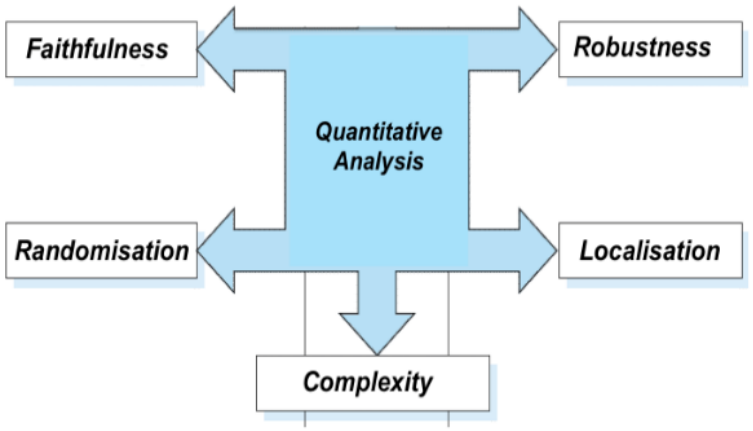


Figure 9: Main categories of Quantitative Analysis for Attributions Evaluation.

Following are the metrics we used to evaluate our explanations:

(a) **Max-Sensitivity**: uses Monte Carlo sampling to approximate the change in explanation when a slight perturbation is introduced to the input [42]. A lower max-sensitivity indicates a better or more stable explanation method.

(b) **Faithfulness**: quantifies the extent to which explanations align with the predictive behavior of the model. Specifically, it assesses whether more

important features, as determined by an attribution method, play a more significant role in the model's outcomes. This is achieved by computing the correlation between probability drops and attribution scores on various points [32].

(c) **Monotonicity**: measures the attribution faithfulness by analyzing if adding important features incrementally improves the model's performance [52]. A monotonic increase in performance as more features are added indicates that the explanation method has successfully captured the relevant features. [22]

(d) **Relative output stability**: measures the stability of an explanation with respect to changes in the output logits of the model [53] [22]. A lower value is considered to be a stable explanation.

(e) **Complexity**: measures the degree to which a few features are used to explain a model's prediction [42].

(f) **Continuity by Local Lipschitz Estimate**: tests the consistency between adjacent examples' explanations. This metric assesses how well the explanation method maintains consistency in its attribution scores for similar inputs [2] [53].

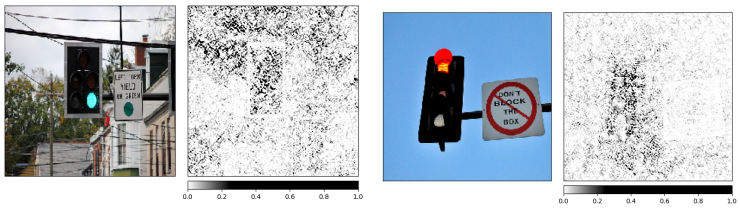
(g) **Model parameter randomization**: Computes robustness of an explanation method by measuring the difference in feature attributions when the model parameters are randomly modified. This difference is calculated as the correlation between the original feature attribution and the new attribution[11].

4 Experimental Evaluation and Results

This section includes our experimentation with the Integrated Gradient and the Contour Analysis method. We investigated the performance of the IG algorithm using different baselines and compared them to the results obtained with the Contour Analysis method.

4.1 Analysis of the Integrated Gradient Algorithm

For the selected sample image dataset, we analyzed the performance of the IG algorithm by applying it to the ResNet-50 model. The Figure shows the input image and its corresponding IG attribution. From the Figure, we can conclude that the IG algorithm highlighted the relevant pixels of the input images.



(a) Example 1

(b) Example 2

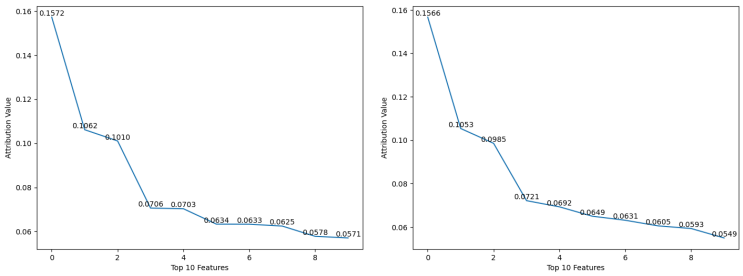
Figure 10: IG attributions for two input images using a single baseline

Next, for a given input image, we compared computed the IG attributions maps for multiple random baselines. We have run the experiments for 5, 10, 20, 50, and 100 baselines for each input image. The Figure shows the example of one input image for different baselines. We can observe that IG produces slightly different attribution maps with different baselines.

Figure 11 shows the values of the top 10 IG attribution values for a single baseline and ten random baselines.

Left: This plot shows the top 10 IG attributions that ran on a single baseline.

Right: This plot shows the top 10 values of IG attributions which ran on ten



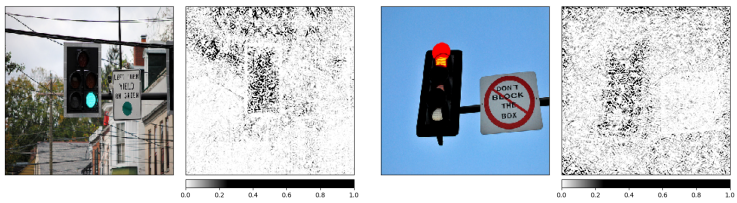
(a) Left: For single baseline (b) Right: For baseline 10

Figure 11: Comparison of Top 10 IG attributions values for 1 Baseline and 10 Baseline

random baselines.

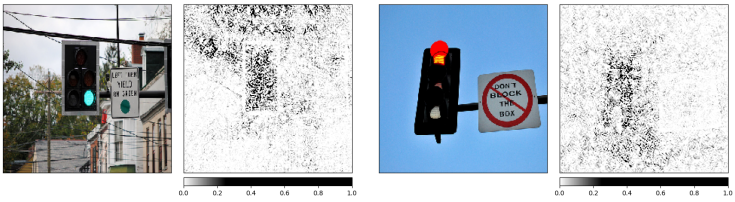
4.2 Comparison of IG using Different Baselines

We compared the attribution maps generated with Mean and Median aggregation methods to analyze the IG performance with different baselines. Figure 12 compares mean and median aggregation methods on IG attributions for a fifty-random baseline. Attribution maps for Mean and Median visually look similar; however, the attribution scores differ. We have used the faithfulness metric using the Contour Detection algorithm to evaluate the attributions for Mean and Median methods.



(a) Input image 1 (b) Input image 2

Figure 12: **Mean Aggregation** of attributions for 50 baselines



(a) Input image 1 (b) Input image 2

Figure 13: **Median Aggregation** of attributions for 50 baselines

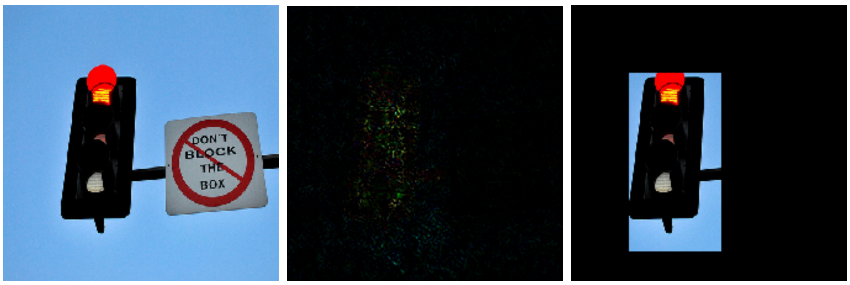
4.3 Analysis of the Contour Method

We used the Contour algorithm to find the important features using IG attribution. The Figure shows the contour generated by the method using IG attributions. Once we have the contour, we map the contour to the input image and extract the important features from the image. We evaluated the model predictions of the generated continuous patch from the top 10% of the IG attributions. We evaluated different images with different numbers of random baselines. By doing so, we are evaluating the faithfulness of the attribution generated by IG.

Our findings indicate that both the IG algorithm and the Contour Analysis approach can successfully identify the most important elements of an image. At the same time, they each have particular strengths and weaknesses. The contour approach can extract more localized and detailed information, while the IG algorithm can highlight more important sections of an image and is more noise-resistant.

4.4 Evaluating important features of attributions

The tables represent the experimental results of evaluating the important features of attributions of IG using the contours method. The evaluation is performed by different baselines used in attribution calculation and types of aggre-



(a) Input image (b) IG attribution (c) Contour Analysis

Figure 14: Continuous patch of Input Image using IG attribution

gation methods used(mean and median).

Table 1: Evaluating Important Features of Attributions for Mean Aggregation

No. of baselines	Aggregation Method	Model Prediction	Contour Prediction	Contour Prediction Label
1	NA	0.999079347	0.95627075	traffic_light
5	mean	0.999079347	0.97275734	traffic_light
10	mean	0.999079347	0.9898733	traffic_light
20	mean	0.999079347	0.97275734	traffic_light
50	mean	0.999079347	0.97275734	traffic_light

Table 1 reports the results of mean-based aggregation. It can be observed that the prediction accuracy for a single baseline is lower than that of using multiple baselines.

Table 2: Evaluating Important Features of Attributions for Median Aggregation

No. of baselines	Aggregation Method	Model Prediction	Contour Prediction	Contour Prediction Label
1	NA	0.999079347	0.95627075	traffic_light
5	median	0.999079347	0.9535067	traffic_light
10	median	0.999079347	0.97275734	traffic_light
20	median	0.999079347	0.97275734	traffic_light
50	median	0.999079347	0.97275734	traffic_light

Table 2 reports the results of median-based aggregation. Similarly, a single baseline's prediction accuracy is lower than multiple baselines.

Comparing the two aggregation methods, mean-based aggregation outperforms median-based aggregation regarding prediction accuracy.

4.5 Evaluation using Quantitative metrics

We computed the performance of different baselines for quantitative evaluations and summarized the results in a table. In Table 3, the symbol ↑ indicates larger values are better for that metric; similarly, ↓ indicates smaller values are better. Table 3 shows that the multiple baselines outperformed the single baseline almost for all the metrics.

Table 3: Quantitative analysis of IG attributions.

No of baseline	Max Sensitivity ↑	Relative Output Stability ↓	Local Lipschitz Estimate ↓	Avg Sensitivity ↓	Sparseness ↑	Complexity ↓	Faithfulness Estimate ↑	Model Parameter Randomisation ↓
1	0.029677952	10.2584109	0.360847613	0.02960295	0.58879829	10.2111315	-0.0112714	0.30925385
5	0.030191214	9.44674807	0.379447938	0.03005376	0.58849138	10.2113134	-0.0094008	0.30886812
10	0.028830782	8.944581107	0.372391622	0.02838973	0.58850471	10.2093682	-0.008943	0.316598
20	0.02802491	11.96360426	0.384525647	0.02848902	0.58924746	10.2092548	-0.0201291	0.31950475
50	0.027492214	9.433153571	0.357073711	0.02756846	0.58934905	10.2090589	-0.0090146	0.31962904

5 Discussion, Analysis, and Conclusion

RQ1 showed that multiple random baselines yielded more robust explanations than the single random baseline. The choice of the baseline can significantly impact the performance of IG. For **RQ2**, we proposed the Contour Analysis method to extract the continuous patch of the input image using IG attributions. Since IG attributions are computed at the pixel level, the Contour Analysis method helped us to extract a continuous and meaningful patch from the attribution map. **RQ3**, the Contour Analysis method does not directly impact the IG attributions; however, it helped us evaluate them.

5.1 Limitations and Future Work

While our research has provided important insights into the performance and limitations of the Integrated Gradients algorithm for explaining the predictions of deep neural networks, some limitations still need to be addressed in future work.

One limitation of our study is that we only tested the IG algorithm on the ResNet-50 architecture, which may limit generalizability to other models. While we evaluated IG using multiple random baselines, evaluating it using other baselines, such as uniform or Gaussian baselines, would be useful.

Additionally, our study only evaluated the performance of the IG algorithm using the Captum library. While the Captum library is a powerful and flexible tool for evaluating the interpretability of deep neural networks, other libraries and tools are available for this purpose. It would be useful to evaluate the performance of IG using these other libraries and tools.

In future work, we plan to address these limitations by evaluating the performance of IG on other types of models and data, using different baselines, and using other libraries and tools. Creating methods to determine the number of baselines needed for reliable results accurately. Additionally, we plan to explore the use of IG for explaining the predictions of deep neural networks in applications such as medical diagnosis or fraud detection, where interpretability is

critical for gaining trust and acceptance from end-users.

Finally, our Contour Analysis algorithm can be used in any Computer Vision domain where the identification of continuous regions in data is needed. For example, In medical imaging, it can be used to identify and isolate regions of interest, such as tumors or lesions.

5.2 Conclusion

In this study, we analyzed the Integrated Gradients algorithm and proposed a contour-based analysis method for evaluating the faithfulness of the IG algorithm concerning different baselines. We used the Captum library to implement the IG algorithm and experimented with multiple random baselines to compare the attributions generated by the IG algorithm.We utilized the Quantus library to evaluate the quantitative metrics of our explanations.

Our results showed that mean-based aggregation outperforms median-based aggregation regarding prediction accuracy. Our findings indicated that using multiple baselines in attribution calculation and evaluation can improve the model's faithfulness and outperform the single baseline in various quantitative metrics.

Our proposed contour-based analysis method provided a visual representation of the important features highlighted by the IG algorithm, allowing for a more intuitive interpretation of the algorithm's performance. This method can be used to evaluate the faithfulness of the IG algorithm for different baselines and can be applied to other attribution algorithms as well.

Overall, our study contributes to understanding the IG algorithm and provides insights into its performance for different baselines. The proposed contour-based analysis method can also be used to evaluate the performance of other attribution algorithms. This method can be extended to incorporate user feedback and generate personalized explanations based on the user's preferences.

A Appendix A: Figures and Tables

List of Figures

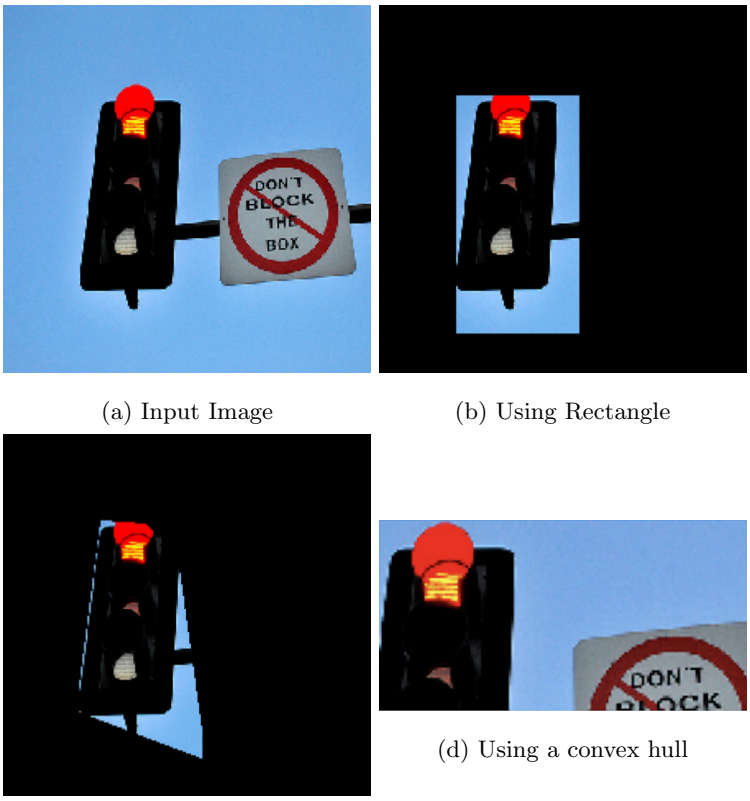
List of Tables

B Appendix B: Additional Experimental Results

B.1 Additional experimental patches using Contour Analysis method

We proposed a novel method called the Contour analysis method for extracting the important features of integrated gradient attributions. Our approach provides the benefit of extracting continuous patches utilizing the outer points of the values. This method can be applied in a broad range of domains that require similar feature extraction techniques. Following are some additional experimental patches using our proposed method.

Various strategies have been experimented with to extract the patches based on boundaries. Figure 15 a is an input image we used for Contour Analysis Method. Figure 15 b uses the bounding rectangle of the new binary mask obtained from the convex hull to extract the patch. The top-left corner coordinates, width, and height define the bounding rectangle. This method provides a rectangular patch encompassing the important features. In Figure 15 c, the patch is extracted, and a line is drawn using polylines, which provides a more precise boundary for the important features. Figure 15 d shows the variation of using a convex hull, a geometric algorithm that can compute the smallest convex polygon that encloses a set of points [54].



(a) Input Image

(b) Using Rectangle

(c) Using Polylines

(d) Using a convex hull

Figure 15: Additional patches of Contour Analysis Method for the given input image

References

[1] A. Esteva, B. Kuprel, and R. Novoa, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, p. 115–118.

[2] S. Bhattacharyya, S. Jha, and J. W. Kurian Tharakunnel, "Data mining for credit card fraud: A comparative study," *Decision Support Systems*, vol. 50, no. ue 3.

[3] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '15*, (New York, NY, USA), Association for Computing Machinery.

[4] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free?-weakly-supervised learning with convolutional neural networks," in *Proceedings of the IEEE Conference on computer vision and pattern recognition*, p. 685–694.

[5] P. Koopman and M. Wagner, "Challenges in autonomous vehicle testing and validation," *SAE International Journal of Transportation Safety*, vol. 4, no. 1, p. 15–24.

[6] D. Gunning, E. Vorm, J. Wang, and M. Turek, "Darpa's explainable ai (xai) program: A retrospective," *Applied AI Letters*, vol. 2, p. 61.

[7] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning." arXiv preprint arXiv:1702.08608.

[8] Z. C. Lipton, "The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery," *Queue*, vol. 16, no. 3, p. 31–57.

[9] D. Vale, A. El-Sharif, and M. Ali, "Explainable artificial intelligence (xai) post-hoc explainability methods: risks and limitations in non-discrimination law," *AI Ethics*, vol. 2, p. 815–826.

[10] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International conference on machine learning*, p. 3319–3328, PMLR.

[11] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," *Advances in neural information processing systems*, vol. 31.

[12] Sturmfels, *Visualizing the Impact of Feature Attribution Baselines"*. Distill.

[13] G. Montavon, W. Samek, and K.-R. Müller, "Methods for interpreting and understanding deep neural networks," *Digital signal processing*, vol. 73, p. 1–15.

[14] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30.

[15] L. Floridi, J. Cowls, M. Beltrametti, R. Chatila, P. Chazerand, V. Dignum, and C. Luetge, "Ai4people—an ethical framework for a good ai society: opportunities, risks, principles, and recommendations," *Minds and machines*, vol. 28, p. 689–707.

[16] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, and O. Reblitz-Richardson, "Captum: A unified and generic model interpretability library for pytorch." arXiv:2009.07896 [cs.LG.

[17] A. Hedström, L. Weber, D. Krakowczyk, D. Bareeva, F. Motzkus, W. Samek, S. Lapuschkin, and M. M. M.-C. Höhne, "Quantus: An explainable ai toolkit for responsible evaluation of neural network explanations and beyond," *Journal of Machine Learning Research*, vol. 24, p. 34.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 770–778.

[19] R. Aumann and L. Shapley. Princeton University Press.

[20] "Smoothgrad: removing noise by adding noise d." Wattenberg.arXiv preprint arXiv:1706.03825.

[21] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable ai: A review of machine learning interpretability methods," *Entropy*, vol. 23, no. 1, p. 18.

[22] SoK, "Modeling explainability in security analytics for interpretability, trustworthiness, and usability," in *Proceedings of The 18th International Conference on Availability, Reliability and Security (ARES'23*, (New York, NY, USA), p. 12, ACM.

[23] F. Bodria, F. Giannotti, R. Guidotti, F. Naretto, D. Pedreschi, and S. Rinzivillo, "Benchmarking and survey of explanation methods for black box models." arXiv preprint arXiv:2102.13076.

[24] A.-K. Dombrowski, C. J. Anders, K.-R. Müller, and P. Kessel, "Towards robust explanations for deep neural networks," *Pattern Recognition*, vol. 121, p. 108194, 2022.

[25] A. Holzinger, A. Saranti, C. Molnar, P. Biecek, and W. Samek, "Explainable ai methods - a brief overview," in *xxAI - Beyond Explainable AI. xxAI 2020* (A. Holzinger, R. Goebel, R. Fong, T. Moon, K. Müller, and W. Samek, eds.), Lecture Notes in Computer Science(), vol 13200, Cham: Springer.

[26] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM computing surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[27] W. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu, "Definitions, methods, and applications in interpretable machine learning," in *Proceedings of the National Academy of Sciences 116*, vol. 44, p. 22071–22080.

[28] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199.

[29] B. Jia, C. Dong, Z. Chen, K.-C. Chang, N. Sullivan, and G. Chen, "Pattern discovery and anomaly detection via knowledge graph," in *2018 21st International Conference on Information Fusion (FUSION*, p. 2392–2399, IEEE.

[30] M. Du, F. Li, G. Zheng, and V. Srikumar, "Deeplog: Anomaly detection and diagnosis from system logs through deep learning," in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, p. 1285–1298.

[31] N. AfzaliSeresht, "Explainable intelligence for comprehensive interpretation of cybersecurity data in incident management," *Ph.* D. Dissertation.

[32] M. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you?": Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, p. 1135–1144.

[33] S. Lundberg and S. Lee, "A unified approach to interpreting model predictions," in *Advances in neural information processing systems*, p. 4765–4774.

[34] R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, p. 618–626.

[35] J. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net." arXiv preprint arXiv:1412.6806.

[36] M. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*, (Cham), p. 818–833, Springer.

[37] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *Proceedings of the 34th International*.

[38] M. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-precision model-agnostic explanations," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1).

[39] "What-if tool," google, accessed on may 14." Online]. Available:.

[40] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International Conference on Machine Learning*, p. 3319–3328, PMLR.

[41] "Xai methods - integrated gradients."

[42] "Tensorflow tutorial on integrated gradients."

[43] R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," in *Proceedings of the IEEE international conference on computer vision*, p. 3429–3437.

[44] G. Erion, J. D. Janizek, P. Sturmfels, S. M. Lundberg, and S.-I. Lee, "Improving the performance of deep learning models with axiomatic attribution priors and expected gradients," *Nature machine intelligence*, vol. 3, no. 7, p. 620–631.

[45] A. Kapishnikov, T. Bolukbasi, F. Viégas, and M. Terry, "Xrai: Better attributions through regions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, p. 4948–4957.

[46] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "*

= equal contribution) imagenet large scale visual recognition challenge," *IJCV*. paper — bibtex — paper content on arxiv — attribute annotations.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 770–778.

[48] F. Chollet, "Keras," *GitHub*. Retrieved from.

[49] P. Dabkowski and Y. Gal, "Real-time image saliency for black box classifiers," in *Advances in neural information processing systems*, p. 6967–6976.

[50] J. Li, W. Li, J. Pan, J. Wu, and X. Zhu, "Visualizing the effects of image transformations on deep neural networks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 11, p. 3066–3075.

[51] S. Hooker, D. Erhan, P.-J. Kindermans, and B. Kim, "A benchmark for interpretability methods in deep neural networks," *Advances in neural information processing systems*, vol. 32, 2019.

[52] V. Arya, R. K. Bellamy, P.-Y. Chen, A. Dhurandhar, M. Hind, S. C. Hoffman, Q. L. Stephanie Houde, R. Luss, and A. Mojsilović, "One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques." arXiv preprint arXiv:1909.03012.

[53] D. A. Melis and T. Jaakkola, "Towards robust interpretability with self-explaining neural networks," *NeurIPS*.

[54] B. Chazelle, "On the convex layers of a planar set," *IEEE Trans. Inf. Theory*, vol. 31, pp. 509–517, 1985.