Optimization Methods for Generating Saliency Masks in Image Classifiers

***Abstract*—The interpretability of artificial intelligence models is fundamental for confidence in their decisions, which is essential for the acceptance and adoption of the technology. In this context, this study investigates the performance of optimization-based methods to generate saliency masks in image classifiers, compared to approaches that use gradients. Several model-agnostic algorithms are proposed incorporating three main hypotheses: use of continuous pixel regions, application of inpainting techniques, and multi-objective optimization. Empirical results demonstrate that optimization-based methods generally outperform gradient approaches and that the three hypotheses, with emphasis on continuous regions, bring new potential to this line of research.**

***Index Terms*—interpretability, explainable artificial intelligence, multi-objective optimization, image classification, saliency maps**

I. Introduction

The use of artificial intelligence-based systems is becoming increasingly common in our society. Among its various uses, one that has already made significant progress in the literature and is increasingly valuable is image classification. Being one of the first types of AI to outperform humans in benchmarks [1], various models and types have been created throughout the development of the field, reaching its current peak with the use of deep neural networks. Its importance in our society lies in the increasingly constant use of automated systems that permeate and assist diverse areas such as facial recognition, engineering, vehicle sensors, geology, weather detection, satellite image recognition, agriculture, medicine, etc. [2].

However, a fundamental aspect for these systems to be feasible and adoptable is trust in the model's prediction. This trust, in turn, does not come only from the quantitative measures resulting from model training, but also involves attempting to interpret why the model behaves the way it does [3]. In neural network models, notably those used in image classification that employ dense hidden layers, this task becomes complex, and consequently, such models are classified as black-box models, since it is virtually impossible to understand their conclusion based solely on their operation. Beyond the essential attribute of trust in model transparency, another derived issue emerges, related to the ethical implications of possible biases that may exist in their decisions, which, due to the lack of understanding of why they were made, present difficulties in being detected [4].

To address this issue, AI interpretability methods have been developed to increase the transparency of these models. According to [5], interpretability is defined as the degree to which a human can understand the cause of a decision and is, therefore, fundamental for the reliable use of automated systems that use AI [6]. Several methods that attempt to dissect the decisions of the models have been created and categorized throughout the development of the area. For this study, which focuses on image classification models, the emphasis will be on developing a model-agnostic technique that provides local explanations. This is defined as attempting to explain the model's prediction for a single instance [7].

In the context of image classification, it is common for these local explanations to be based on finding which pixels would be considered important in the model's decision. Important pixels are those that contribute positively to that particular prediction. Methods such as Saliency Maps [8], GradCam [9], Guided Backpropagation [10] and Guided GradCam [9] perform this procedure in different ways, but all are specific to neural networks, since they depend on access to the model's internal gradients to identify which pixels are important. The studies [11] and [12], on the other hand, developed new optimization-based methods that search for these pixels without needing access to specific components of the model's architecture, characterizing them as agnostic, and ultimately obtained results similar to gradient-based methods.

Building on research into the use of optimization in identifying important (and unimportant) pixels, this work aims to investigate and propose new implementations based on hypotheses raised through the analysis of [11] and [12]. Such hypotheses stem from the analysis of possible improvement opportunities grounded, in summary, on the following criteria: forcing the optimization algorithm to search for regions of continuous pixels, considering that important pixels tend to form contiguous regions instead of being randomly dispersed across the image; using inpainting techniques during the pixel replacement process in optimization search, allowing the selected region of the image to maintain visual coherence; and, finally, employing multi-objective optimization, attempting to balance the selection of regions with smaller areas with the maximization of their impact on the model's prediction, aiming for more concise explanations.

Therefore, and with what will be presented in the following sections, the contributions of this work are:

- Description of new optimization-based methods to attempt to interpret local decisions made by image classification models, based on three hypotheses of improvements from previous work.
- Performing empirical testing comparing the developed methods with the aforementioned state-of-the-art methods, including both those using optimization and those

using gradient-based techniques.
- Evaluation of the proposed hypotheses and comparison among them, as well as between optimization-based and gradient-based methods.

II. Methodology

Building upon [11] and [12], this study conducts experiments on the application of optimization in the interpretability of image classification models, in which attempts are made to find pixels that are important and unimportant for a classification instance. The definition of a pixel's importance in this context is given by how much it contributes to the prediction made by the classifier; if the pixel contributes positively, it is said to be important, and vice versa. In general, these methods follow a common framework in their implementation, as described:

1) The algorithm receives as input the classifier model and the image for which an explanation is desired, in order to discover the important and/or unimportant pixels.
2) Next, a single-objective optimization process is performed, in an attempt to find a mask that makes the model's prediction as accurate as possible for that image and label. Letting M and N be the width and height of the image, respectively, the set of possible states for this optimization is described as an M x N matrix with values x_{ij} that can be 0 or 1, where 1 indicates the presence of the mask. Applying the mask requires replacing the image pixels not described by the mask with something else, which will be called noise. This becomes another variable in the algorithm and can be a solid color like black and white or something more elaborate like the average color of the remaining pixels in the image. The studies differ in the algorithm used for optimization, with Hill Climbing being used by [11] and a generic genetic algorithm being used by [12].
3) The optimization objective function, which guides the entire process, is described by a single objective f_1, namely the score of the classifier model itself. Thus, the output of the optimization is a mask that, when applied to the image, will result in the highest score of the classifier found and, therefore, it is argued that this mask describes which pixels are most important to the model. Regarding the unimportant pixels, two different approaches are taken by the studies: either take the complementary pixels to the important pixels optimization, or performing a new optimization using the objective function in reverse.

Both studies presented promising results when compared to gradient-based methods, matching and even surpassing them in some cases. This study, therefore, aligns with this promising line of research and proposes the creation of new methods based on three hypotheses that could improve the effectiveness of the optimization algorithm: forcing pixel continuity; using pixel replacement methods considering the context of neighboring pixels; and performing multi-objective optimization that considers, in addition to the classification score, the masked area. Each of these hypotheses generated a new version of this method and will be explained in the following subsections.

A. Optimization method

A generic genetic algorithm was chosen as the optimization method for all versions implemented, following [12]. This is an optimization technique inspired by natural selection that iteratively evolves a population of potential candidates. Each individual is evaluated according to an objective function, and the best are selected for reproduction. Using genetic operators, such as crossover and mutation, new populations are created promoting diversity in the search space. After successive generations, the population converges towards better solutions, as more suitable traits are preserved and propagated.

B. Pixel continuity

The first hypothesis stems from the fact that previous methods create masks that do not necessarily have pixel continuity; that is, the pixels chosen to remain in the image could be anywhere in the original image, even at completely different points. This type of perturbation can create difficulties for the classifier model, possibly removing important context that was together and generating fragmented images with sporadic pixels. The addition of noise to the remaining pixels can also divide parts that were contextually important and deform the original image, creating a completely new context for the classifier, which will likely have its effectiveness compromised, given that such discontinuities filled with strange artifacts do not usually appear in its training.

In an effort to solve these problems, this work proposes forcing pixel continuity in the mask generation process. To this end, two new masking methods were developed and will replace previous works. The first method replaces the previous mask, in the form of an M x N matrix, with a simple sequence of four values [A_x,A_y,B_x,B_y] that will represent the coordinates of two pixels, A and B, which will describe a rectangle that will be used as a mask to represent the pixels to be evaluated. The second method is almost the same, but instead of generating a rectangle, it attempts to be more dynamic and generates a polygon that can have n $>$ 3 vertices/sides and is described by the sequence [A_x,A_y,B_x,B_y,C_x,C_y,...] which represents the n coordinates A, B, C, ..., forming the polygon. In both methods, the regions described by the sequences are transformed into masks, applied to the images, and the optimization process continues as previously described. Fig. 1 shows an example of applying these two mask models with solid black noise.

C. Pixel replacement with context

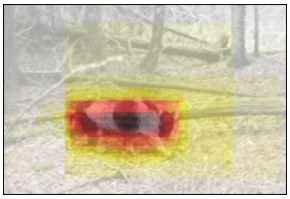
The second hypothesis of this work concerns the use of noise during the optimization method. As mentioned earlier, this pixel replacement causes the insertion of extraneous artifacts, generating unnatural images that deviate the optimization algorithm from its normal classification. The intuition behind this hypothesis is that by using a pixel replacement

(a) Original

(b) Rectangular mask

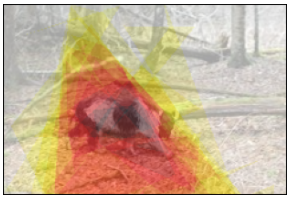
(c) Poligonal mask

Fig. 1: Comparison of continuous pixel masks

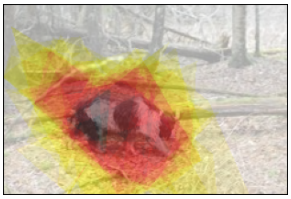


(a) Rectangle with black noise

(b) Rectangle with inpainting



(c) Polygonal with black noise



(d) Polygonal with inpainting

Fig. 3: Heatmap examples

technique that considers the context of local neighboring pixels, the images become more natural to the classifier by eliminating these possible artifacts, which ultimately increases the effectiveness of the optimization algorithm in its search. The technique used here for this is based on [13], with an implementation available at [14].

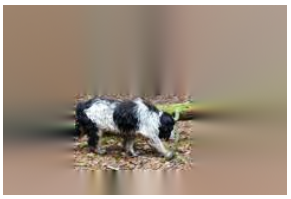
An example comparison of this application can be seen in Fig. 2, which shows the result of an example mask with the application of solid black noise compared to an application that uses the inpainting technique, which replaces pixels respecting the context of their neighbors. It is possible to observe that the addition of the black mask adds a strange context to the image that obviously does not belong to it and that, therefore, will negatively influence the model and may divert the optimizer from the region of interest. The substitution with context, in turn, maintains the semantics of the image and allows the classifier, which is normally trained on datasets without artificial black regions or strange bodies, to perform naturally.

D. Minimizing the mask area as a new objective function

The third hypothesis stems from an aspect that emerged during the initial stages of this study. By forcing the pixels to be continuous, it was noticed that sometimes the optimization algorithm generated an area much larger than half of the original image, not necessarily focusing on the classified object, but encompassing it with a considerable margin. Thus, to address this problem, the addition of another objective to the optimization algorithm was considered, thereby making it multi-objective. Consequently, in addition to the function f_1, which would be the same as in the aforementioned algorithms, another function f_2 is added representing the mask's area, such that the smaller the area, the better the mask will be evaluated by the optimization process.

Transforming the optimization algorithm into a multi-objective algorithm brings with it some challenges. Since it is not possible to determine a single best solution for a multi-objective algorithm, as there is a trade-off between the two functions (the smaller the mask area, the lower the classifier's score tends to be for the image), a solution set that aims to be close to the Pareto frontier will be found. This set will contain various masks that will vary in size but will fill a region of the image. The idea is that by being forced to generate these various masks with limited area sizes, the optimizer will tend to find a common region among them, indicating that this region is indispensable in its solutions because the classifier needs this region to make its correct classification. Fig. 3 shows examples of overlaps of various masks found after this multi-objective optimization process in various combinations. It is possible to notice that despite several different solutions being found, most converge to an image region that can, intuitively, be said to represent an indispensable part for the classifier and is therefore considered important.

(a) Black noise



(b) Inpainting

Fig. 2: Comparison of black noise vs. use of inpainting technique

III. Implementation and Results

A. Parameterization

Given these hypotheses, variations of the algorithms were implemented, combining one or more of them as identified in Table I, with identifying names that will serve for future reference. The implementations were based on [12], which employs a genetic algorithm, with the same crossover and selection processes. Mutation was adapted for non-binary

values, selecting a number between the maximum and minimum allowed, an approach chosen given the narrow total interval between these values. For multi-objective methods, the structure of the single-objective algorithms was maintained, with the significant change occurring in the storage of non-dominated solutions constituting the search result set and their incorporation into the selection process. The general parameterization used was: population of 50 individuals, 1000 iterations, 10 individuals in the selection, and mutation probability of $pm = 0.9$. The initial population was generated randomly. For performance economy, early stopping was implemented in 35 iterations with no improvement in the best individual(s).

TABLE I: Methods description

#	Name	Description
1	ga	Implementation of [12]
2	hc	Implementation of [11]
3	ga_rect	Rectangular masks
4	ga_polygon	Polygonal masks
5	ga_rect_i	Rectangular masks and inpainting
6	ga_polygon_i	Polygonal masks and inpainting
7	ga_rect_m	Rectangular masks and multi-objective
8	ga_polygon_m	Polygonal masks and multi-objective
9	ga_rect_m_i	Rectangular masks, multi-objective and inpainting
10	ga_polygon_m_i	Polygonal masks, multi-objective and inpainting
11	gradcam	Implementation of [9]
12	saliency	Implementation of [8]
13	guidedbackprop	Implementation of [10]
14	guidedgradcam	Implementation of [9]

Some of the created versions receive extra individual parameters. Algorithms using a polygonal mask used n = 5 vertices; Inpainting algorithms used the implementation of [14], with a context radius = 3; while for multi-objective algorithms, a threshold parameter t = 0.3 was defined, indicating that only pixels that appear in at least 30% of solution masks are considered for final evaluation. All of these parameters were chosen empirically. Finally, the noise values used followed the two previous works, limited to the solid colors white and black, Gaussian noise, and the normalized average of the image itself.

Gradient-based algorithms were implemented using [15]. In this implementation, the only extra parameter needed, besides the model and image, is the number of pixels to be chosen as relevant. To maintain a fair comparison with other methods, an area of 50% was chosen. The selection of unimportant pixels in these algorithms was done using the complement of the important pixel mask.

To test all methods, a classifier model and a set of images to be explained are necessary. The model used in the experiment was ResNet-18 [16] and the image set used was the CIFAR-10 [17] test set, a widely used dataset in the literature, containing sixty thousand 32x32 images, divided into ten distinct classes. Fig. 4 shows an example image from the dataset and the results obtained for important pixels in each evaluated method. The code implementation is available at https://github.com/racs4/optimization-ieee.

B. Metric

Properly parameterized, all methods were executed for the selected images, saving the values of the original score and the mask scores for important and unimportant pixels. All these values are in the range [0, 1]. With them, it is possible to use the Feature Impact Index (FII) metric [11] defined by $abs(S_{original} - S_{selection})$, where $S_{original}$ is the classifier score for the label being explained with respect to the original image, and $S_{selection}$ is the score with respect to the masked image. This metric, therefore, evaluates how close the image with the mask applied approximates to the original prediction; consequently, for important pixels, values close to 0 are desired, while for unimportant pixels, values close to 1 are desired. Tables II and III present the average and standard deviation results of this metric, grouped by methods and noise, for important and unimportant pixels. The lines are ordered in ascending and descending based on general mean, respectively, in order to show the methods that had the best averages first.

C. Results

Regarding the task of identifying which pixels the classifier considers relevant, it was observed that *ga_rect_m_i* method, which uses a rectangular mask, is multi-objective and performs inpainting during optimization, was the algorithm implemented with particularly promising performance. Its result is quite similar to GradCam, which obtained the best average and, as can be seen in Fig. 4, is the gradient algorithm that preserves the largest continuous region. The methods that do not use inpainting follow closely, but with a much wider distribution and similar to the subsequent *ga* and *hc*, but with lower averages than these. The remaining gradient-based algorithms and those that use inpainting appear to complete the methods with averages worse than the others. In comparing optimization approaches and gradient-based methods, it is noted that optimization-based algorithms demonstrate greater overall effectiveness, with the only discrepancies being the methods that use inpainting, which performed significantly worse than the others, and GradCam, which ranked best among all and positioned well ahead of the other gradient-based methods.

The results for irrelevant pixels once again show better performance from the search algorithms. The standout algorithms are *ga_polygon*, *ga_polygon_m_i*, *ga_rect*, and *ga*, which presented averages closest to the maximum possible value. The other algorithms showed similar results, with close numbers. The *ga_rect_m_i* and GradCam methods, which performed better in finding important pixels, had difficulties searching for irrelevant pixels, and therefore no algorithm achieved good performance in both tasks. For gradient-based methods, the results suggest a moderate performance, remaining in intermediate positions between the best and worst optimization algorithms, but without particularly significant results.

D. Discussion

Based on the results presented, it is possible to see that there are promising factors in using all the techniques mentioned

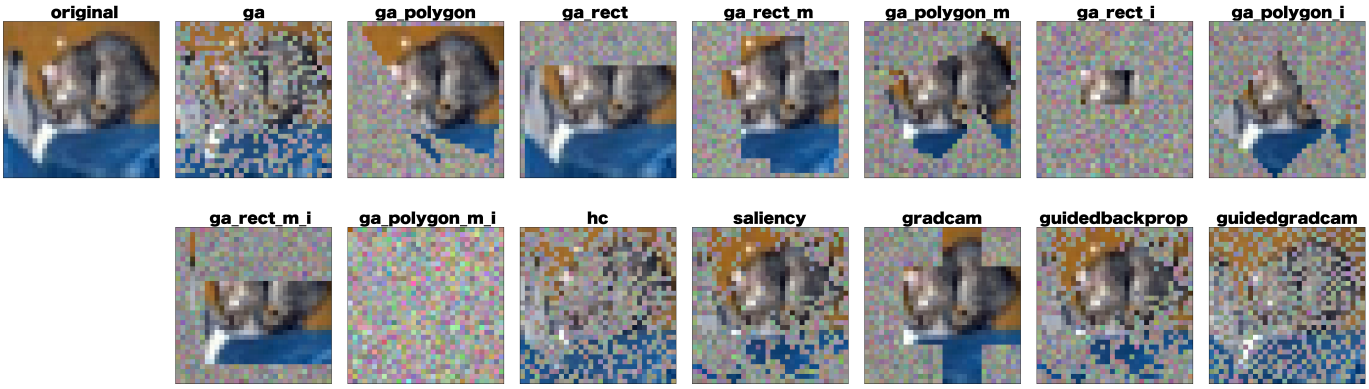


Fig. 4: Example of important pixels found for an image with Gaussian noise.

TABLE II: Important pixels results

	black	gaussian	norm_mean	white
gradcam	**0.2093 0.3519**	**0.1928 ± 0.3353**	**0.2195 ± 0.3598**	0.236 ± 0.3715
ga_rect_m_i	0.2097 ± 0.3662	0.2275 ± 0.3794	0.225 ± 0.3781	**0.2296 ± 0.3788**
ga_rect	0.3539 ± 0.4426	0.2606 ± 0.4027	0.3271 ± 0.4331	0.349 ± 0.4321
ga_polygon	0.4373 ± 0.452	0.3762 ± 0.4424	0.4298 ± 0.4517	0.411 ± 0.4491
ga_rect_m	0.4261 ± 0.4475	0.44 ± 0.447	0.4266 ± 0.4479	0.4485 ± 0.4517
ga_polygon_m	0.4943 ± 0.4428	0.5097 ± 0.4457	0.5297 ± 0.4435	0.5453 ± 0.4437
ga	0.6168 ± 0.4177	0.6743 ± 0.4077	0.5713 ± 0.431	0.4398 ± 0.4315
hc	0.5949 ± 0.4608	0.7066 ± 0.4221	0.6364 ± 0.4499	0.6558 ± 0.4432
guidedbackprop	0.6692 ± 0.4063	0.7203 ± 0.3916	0.6744 ± 0.403	0.6975 ± 0.3906
saliency	0.6692 ± 0.4062	0.7208 ± 0.3912	0.6744 ± 0.403	0.6975 ± 0.3906
ga_rect_i	0.761 ± 0.3786	0.7537 ± 0.385	0.766 ± 0.3737	0.7615 ± 0.3597
guidedgradcam	0.7573 ± 0.3611	0.7915 ± 0.3453	0.7742 ± 0.348	0.7888 ± 0.3307
ga_polygon_i	0.8048 ± 0.3467	0.7887 ± 0.3638	0.8074 ± 0.3479	0.8068 ± 0.3516
ga_polygon_m_i	0.8631 ± 0.3126	0.8378 ± 0.3273	0.8615 ± 0.306	0.8456 ± 0.232

here. The hypothesis of pixel continuity can be empirically confirmed, both for unimportant pixels, where the best results were obtained by the proposed methods, and for important pixels, since the *ga_rect_m_i* method was equivalent to GradCam, and the latter, despite using gradients, also generates masks with continuous pixel regions. In comparing the proposed methods regarding pixel continuity, a consistent pattern is observed: for relevant pixels, the methods using rectangular masks were generally superior to their version with polygon masks; for irrelevant pixels, the opposite occurs. The intuition behind this may be related to the metric calculation, since rectangular masks tend to be more regular than polygonal ones, influencing a more positive evaluation of the classifier, which benefits the metric for important pixels and harms it for unimportant pixels. In addition, it is noticeable that optimizing polygonal masks always involves more variables to be optimized and, therefore, demands more from the optimizer's search process.

Regarding the use of the inpainting technique, except for the good results of *ga_rect_m_i* (important pixels) and *ga_polygon_m_i* (unimportant pixels), all algorithms that employed this technique obtained inferior results, ranking at the bottom of both tests. To explain this outcome, it is important to note that the algorithms that used this technique are the only ones that do not optimize directly with the noise that will be applied for metric evaluation, whereas other methods perform the search with the noise already applied to unmasked pixels. Another significant point refers to the dataset used, composed only of 32x32 images, which may provide insufficient context for the inpainting algorithm used. Nevertheless, despite this disadvantage and mixed results, the presence of this technique in two algorithms among the top-performing ones demonstrates its potential for future investigations using different tests and parameters.

Something similar can also be said for the use of the multi-objective algorithm. It is interesting to note that both algorithms that performed well using the inpainting technique also used the aspect of the third hypothesis. But, in general, the methods employing it either underperformed or matched their single-objective counterparts. As with the second hypothesis, this indicates that there are also specific aspects of using the multi-objective algorithm that can be explored to seek better performance of these metrics. In general, it is noticeable that the methods using optimization performed better than the methods using gradients, and that those employing continuous regions were more efficient. However, optimization methods require more computational time to deliver their results, which should be taken into account as a trade-off. Finally, the difference between the noise values was negligible across all algorithms, indicating that pixel replacement, regardless of the

TABLE III: Unimportant pixels results

	black	**gaussian**	**norm_mean**	**white**
ga_polygon	**0.876 ± 0.291**	0.804 ± 0.3503	0.858 ± 0.3074	**0.8745 ± 0.2933**
ga_polygon_m_i	0.8626 ± 0.3105	**0.8376 ± 0.3272**	**0.8608 ± 0.3023**	0.8453 ± 0.2324
ga_rect	0.8198 ± 0.3434	0.7902 ± 0.3617	0.8171 ± 0.3421	0.8469 ± 0.317
ga	0.7814 ± 0.3561	0.8165 ± 0.3357	0.7985 ± 0.3446	0.853 ± 0.2953
guidedgradcam	0.7462 ± 0.3663	0.774 ± 0.3561	0.7533 ± 0.3603	0.7679 ± 0.3418
gradcam	0.7477 ± 0.3738	0.6777 ± 0.3925	0.7594 ± 0.3685	0.7615 ± 0.3671
guidedbackprop	0.6902 ± 0.3919	0.744 ± 0.3739	0.7011 ± 0.3853	0.7207 ± 0.3748
saliency	0.6902 ± 0.3919	0.7411 ± 0.3748	0.7011 ± 0.3854	0.7207 ± 0.3748
ga_rect_m_i	0.7204 ± 0.4046	0.6691 ± 0.4158	0.7088 ± 0.4099	0.7249 ± 0.3887
ga_polygon_i	0.6522 ± 0.4262	0.6453 ± 0.4309	0.6562 ± 0.4274	0.678 ± 0.4217
ga_polygon_m	0.657 ± 0.4041	0.5841 ± 0.4188	0.6439 ± 0.4101	0.6764 ± 0.3991
hc	0.5754 ± 0.4653	0.6778 ± 0.4358	0.6176 ± 0.4552	0.6356 ± 0.4505
ga_rect_m	0.5949 ± 0.4431	0.5009 ± 0.4408	0.6007 ± 0.4418	0.6238 ± 0.434
ga_rect_i	0.453 ± 0.4534	0.483 ± 0.4567	0.4551 ± 0.4529	0.4694 ± 0.4467

chosen method, made no difference in the metric results for this image set.

IV. Conclusion

This study demonstrated that optimization-based methods are competitive and, in some cases, superior to gradient-based methods for generating saliency masks in image classifiers. The continuous regions hypothesis proved effective, with the *ga_rect_m_i* method achieving performance equivalent to GradCam. Inpainting and multi-objective techniques yielded mixed results, yet their potential was evidenced by the algorithms that employed them and ranked among the top-performing methods. Future research directions include investigating these techniques behavior on higher-resolution images, exploring alternative parameters for inpainting and multi-objective, and assessing their application in critical domains, such as medical images.

References

[1] N. Maslej et al., 'Artificial Intelligence Index Report 2025', arXiv [cs.AI], 20-Nov-2025.

[2] A. A. Elngar et al., 'Image classification based on CNN: A survey', Journal of Cybersecurity and Information Management, p. PP. 18-50, 2021.

[3] Z. Hossain et al., "Implementing Explainable AI to Enhance Business Decision Making & Bridging the Trust Gap," 2025 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), pp. 0072–0077, Feb. 2025.

[4] E. Barnes and J. Hutson, 'Navigating the Complexities of AI: The Critical Role of Interpretability and Explainability in Ensuring Transparency and Trust', 06 2024.

[5] O. Biran and C. V. Cotton, 'Explanation and Justification in Machine Learning: A Survey Or', 2017.

[6] U. Ehsan, Q. V. Liao, M. Muller, M. O. Riedl, and J. D. Weisz, 'Expanding Explainability: Towards Social Transparency in AI systems', in Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, Yokohama, Japan, 2021.

[7] C. Molnar, Interpretable Machine Learning: A Guide for Making Black Box Models Explainable, 3rd edn. 2025.

[8] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," arXiv.org, Apr. 19, 2014.

[9] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," 2017 IEEE International Conference on Computer Vision (ICCV), pp. 618–626, Oct. 2017.

[10] Jost Tobias Springenberg, Alexey Dosovitskiy, T. Brox, and M. A. Riedmiller, "Striving for Simplicity: The All Convolutional Net" arXiv (Cornell University), Jan. 2014.

[11] A. Oliveira, Cleber Zanchettin, S. Silva, L. N. Matos, and Paulo Novais, "A Hybrid Post Hoc Interpretability Approach for Deep Neural Networks," Lecture notes in computer science, pp. 600–610, Jan. 2021.

[12] Marcelo, C. L. Oliveira, Santos, Paulo Novais, L. N. Matos, and A. Britto, "Genetic Algorithm to Understand Image Classification," Lecture notes in networks and systems, pp. 541–553, Nov. 2025.

[13] A. Telea, 'An Image Inpainting Technique Based on the Fast Marching Method', Journal of Graphics Tools, vol. 9, 01 2004.

[14] G. Bradski, 'The OpenCV Library', Dr. Dobb's Journal of Software Tools, 2000.

[15] N. Kokhlikyan et al., 'Captum: A unified and generic model interpretability library for PyTorch', ArXiv, vol. abs/2009.07896, 2020.

[16] He, K., et al, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[17] A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Toronto, ON, Canada, Tech. Rep., 2009.