Quantifying Explainability of Saliency Methods in Deep Neural Networks with a Synthetic Dataset

Erico Tjoa, Guan Cuntai, *Fellow, IEEE*

Abstract—Post-hoc analysis is a popular category in eXplainable artificial intelligence (XAI) study. In particular, methods that generate heatmaps have been used to explain the deep neural network (DNN), a black-box model. Heatmaps can be appealing due to the intuitive and visual ways to understand them but assessing their qualities might not be straightforward. Different ways to assess heatmaps' quality have their own merits and shortcomings. This paper introduces a synthetic dataset that can be generated adhoc along with the ground-truth heatmaps for more objective quantitative assessment. Each sample data is an image of a cell with easily recognized features that are distinguished from localization ground-truth mask, hence facilitating a more transparent assessment of different XAI methods. Comparison and recommendations are made, shortcomings are clarified along with suggestions for future research directions to handle the finer details of select post-hoc analysis methods. Furthermore, mabCAM is introduced as the heatmap generation method compatible with our ground-truth heatmaps. The framework is easily generalizable and uses only standard deep learning components.

Impact Statement—We provide a synthetic dataset with clear, unambiguous features. We demonstrate that heatmap-based XAI methods are not necessarily producing heatmaps that correspond to human intuition of "explanation".

Index Terms—Blackbox, Computer Vision, Deep Neural Network, Explainable Artificial Intelligence

I. INTRODUCTION

EXplainable artificial intelligence (XAI) has been gathering attention in the artificial intelligence (AI) and machine learning (ML) community recently. The trend was propelled by the success of deep neural network (DNN), especially convolutional neural network (CNN) in image processing. DNN has been considered a blackbox because the mechanism underlying its remarkable performance is not well understood. XAI research has thus developed in many different directions. Among them is the saliency method, where heatmaps are generated to point at where an AI model is "looking at" as it makes a decision or prediction. The heatmaps have been treated as explanations, and they are desirable because are compatible with human's visual comprehension, easy to read and interpret.

In the most ideal scenario, ground-truth heatmaps are available, and they provide us with clues on how to fix an algorithm that produces wrong predictions. The sub-optimal heatmaps facilitate an analysis that reveals the problematic parts of the algorithm. Once these faulty parts are identified, the algorithm

This work is done under the Alibaba-NTU Talent Program.

Erico is affiliated to Alibaba HealthTech division and Nanyang Technological University at the time of writing.

Guan Cuntai is a professor at Nanyang Technological University, School of Computer Science and Engineering.

can be iteratively improved, and then new heatmaps converge closer towards the ground-truth heatmaps. However, ground-truth heatmaps are sometimes not available. Furthermore, many of the formulas used to generate heatmaps are given using heuristics, e.g. the baseline obtained by the gradient method [1] or by setting negative signals/weights to zeros [2], hence their relations to the groundtruth might not be clear.

Various XAI methods have been designed and evaluated in many different ways. However, evaluation methods do not necessarily reflect the relevant need for explanations, especially because the correctness of explanations is context dependent and can be subjective. In this paper, our goal is **to measure the correctness of explanation** as objectively as possible in the context of image classification. This is done by using a synthetic dataset with in-built ground-truth heatmaps whose interpretability matches human intuition. The novelty lies in the generation of unambiguous discrete-valued attribution heatmaps as explanation and the introduction of a stringent performance metric that counts "hits" and "misses" without ambiguity. The dataset is also customizable and augmented with noisy variations of basic shapes achievable by closed-form equations (such as circles, rectangles etc). Besides, ground-truth heatmaps are automatically generated alongside the image data and labels, avoiding the laborious process of manually marking heatmap features. Finally, we will introduce mabCAM, an easily generalizable heatmap generation method that is compatible the aforesaid dataset.

In section II, we review how XAI methods have been evaluated in the literature and focus on the challenges in their interpretation. In section III-A, we introduce the aforementioned synthetic dataset. For this paper, specifically a 10-class dataset is used to compare 9 different XAI methods. Each data sample consists of an object with a simple shape and its corresponding heatmap designed to be numerically unambiguous. The shapes and colours are computed through Algorithm 1 and 2 using modifications of standard mathematical formulas (equation of ellipse etc); github link is available for details. More precisely, the correctness of heatmaps can be verified in a simple and objective way through the computation of recall and precision. The rest of section III describes the implementation of neural network training for ResNet34 (henceforth ResNet), AlexNet and VGG, followed by validation and evaluation processes, followed by the description of *five-band-score*, a metric defined to capture quantities such as recall and precision that take into account the distinct meaningful regions in heatmaps. We have also introduced *mabCAM*, a generic framework to generate heatmaps akin to semantic segmentation process that is compatible with the framework described by our synthetic

heatmaps. Section IV discusses the recall-precision results and ROC curves. Finally, we conclude with recommendations and provide some caveats. Note: though heatmaps are sometimes interchangeably called saliency map, we only refer to them as heatmaps here because we want to distinguish them from XAI method whose name is Saliency.

II. RELATED WORKS

In this section, we provide a survey of several XAI papers and briefly discuss how their results are quantified.

A. Regarding Heatmap Ground-truths

Our idea of using synthetic data for evaluating XAI methods resembles the synthetic flower dataset in [3]. The flower dataset comes with ground-truth masks for discriminating features. The paper measures the correctness of heatmap explanations using IoU (Intersection over Union) between thresholded heatmaps and the ground-truths. Our dataset also provides ground-truth masks, but we provide different labels for two separate notions "discriminating features" and "localization" (respectively shown as dark red and light red regions in fig. 1). We assess the quality of heatmaps using precision, recall and ROC instead, though the quantities are slightly modified as we distinguish localization from feature discrimination.

The paper that introduces SmoothGrad [4] mentioned that, at the time of its publication, there was no ground-truth to allow for quantitative evaluation of heatmaps. It then proceeds with 2 qualitative evaluations instead. As of now, even though there are many different datasets available for AI and ML researches, the corresponding ground-truth explanations (such as heatmaps) are typically not available. To address this problem, we directly provide unambiguous ground-truth heatmaps.

B. XAI and their Evaluations

Heatmap-based XAI methods have been evaluated using several different metrics. Many of them suffer from various problems, especially correctness w.r.t human observation e.g. CHR problem (defined later), and, in an extreme case, *the pointing game*. By contrast, this work provides a very unambiguous way to measure the correctness of heatmaps as XAI *explanation*. This paper also serves as an instance to reiterate that existing XAI methods can be sensitive to datasets, thus cautioning users on their context dependence.

The experimental study [5] have explored many existing metrics, through which they compare different heatmap-based XAI methods on several DNN architecture and datasets. The metrics include faithfulness, sensitivity and stability, some with modifications.

Saliency, deconvolution and LRP (Layerwise Relevance Propagation) have been compared [6] using the "most relevant first" (MoRF) heatmap evaluation framework, where the quantity Area under Perturbation Curve (AOPC) is computed. Heatmap pixels are ordered according to importance $O = (r_1, r_2, \dots)$. The original image is perturbed by replacing the most important pixels starting from r_1, and then its AOPC is computed progressively with more perturbations. There is a

computational vs human relevance (CHR) problem, i.e. what is computationally most relevant may not correspond to what human finds relevant, especially when r_k can be a single isolated pixel. The correctness of ordering O is not addressed.

CAM [7] and GradCAM [8] heatmaps are shown to improve the localization on ILSVRC (ImageNet Large Scale Visual Recognition Challenge) datasets. By observing the change in the log odd scores after deleting image pixels, the relevance of image pixels to the decision or prediction of a model can be determined as well [9]. GradCAM paper demonstrates through human studies that its heatmaps help increase human's performance in categorical tasks. CAM paper and many other heatmap-based XAI papers do not report report similar human studies. CHR problem may be present, as only computational relevance is presented.

The earlier paper on layerwise relevance propagation (LRP) [2] introduces the epsilon-LRP. It displays heatmaps generated from many sample data, although many heatmaps do not appear to visually demonstrate good consistency in their pixel-wise assignment of values (different improvements have since been suggested). Tests were conducted on the effect of transformation on the images, for example, by flipping MNIST digits, and *mean prediction* is defined to assess the method after interchanging pixels systematically based on relevance computed by LRP. Still, the paper itself mentions that the analysis is semi-quantitative. Furthermore, quantitative results presented mainly analyzes computational relevance (CHR problem). Other methods that may suffer from CHR problem include ROAR (RemOve And Retrain) [10] (consider the correctness of ordering $\{e_i^o\}_{i=1}^N$) and SWAG [11] (see e.g. their black swan saliency).

Two simple sanity checks [12] have been proposed for saliency maps (though they are easily applicable for others): (1) test whether algorithm parameters affect the heatmap output (2) test whether data ordering affect the output. The first sanity check is performed by randomizing weights layer by layer and then computing similarity metrics such as structural similarity index (SSIM), histogram of gradients (HOGs) and rank correlations. It reports possibly severe problems with existing methods. For example, after a DNN's weight parameters are randomized in all layers, GradCAM still produces heatmaps with high similarity values when compared to heatmaps produced by the DNN with no or few randomization, as though the DNN itself is irrelevant. Besides, the paper also reports that epsilon-LRP, DeepLift [9] and integrated gradients [13] return a large part of the input image, possibly confusing inputs with the values relevant to explanations. Using DeepLIFT on our synthetic dataset, on the contrary, show more favorable results, as background noises appear to be filtered away to a good extent. This could indicate sensitivity to the dataset.

Likewise, several heatmap methods are evaluated by measuring similarity w.r.t *sensitivity-n* property [14]. The sum of attribution is compared to the same sum but with n features removed. The paper presents several useful empirical observations from their XAI evaluations. We similarly provide more observations on heatmaps as explanations at the end of this paper. Similarity metrics are also used with the above-
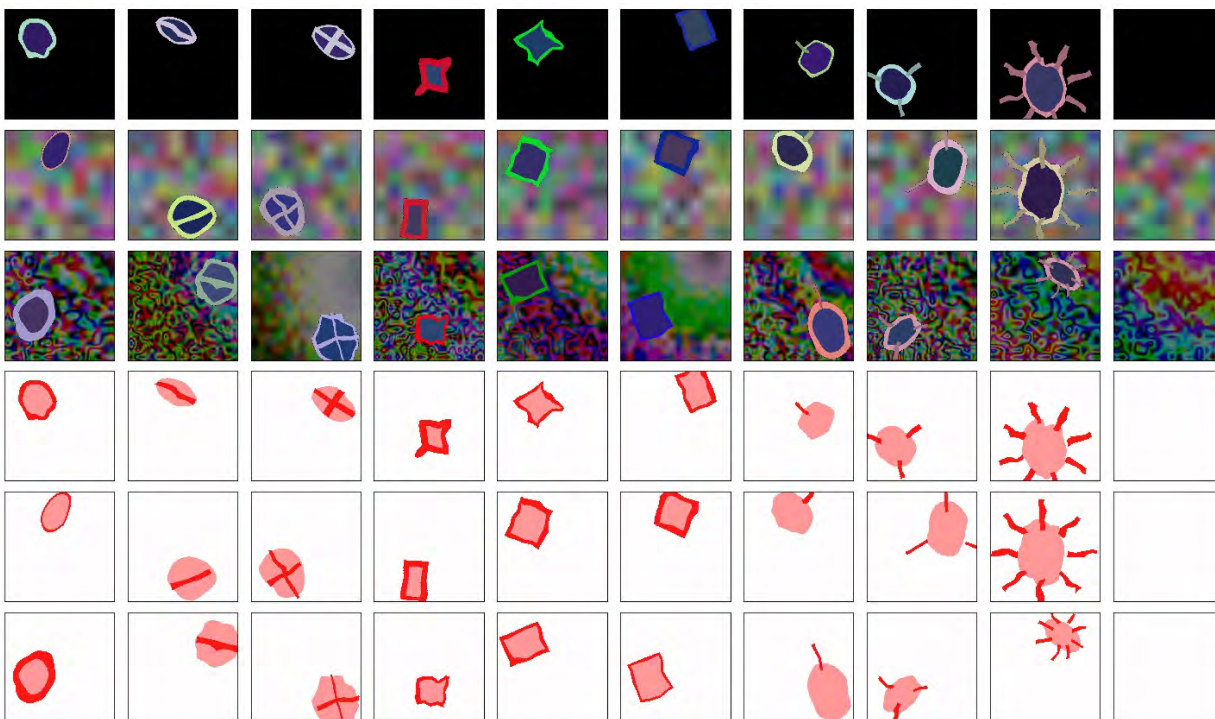
Fig. 1: The first row shows 10 different types of shapes that can be generated by our algorithm, placed in background type 1, i.e. dark background. We refer to the different types of cells as cells type 0 to 8, where 0,1,2 are circular, 3,4,5 are rectangular and 6,7,8 are circular with one, three and eight tails respectively. Their alternative names in the codes are CCell (0), CCellM (1), CCellP (2), RCell (3), RCellB (4), RCellC (5), CCellT (6), CCellT3 (7), CCellT8 (8) respectively. C denotes circular cell, R rectangular, T tails, M minus, P plus. The last column (or type 9) does not contain any cell. The second and third rows are similar to the first row, except they are placed on background type 2 and 3 respectively. Row 4, 5, 6 are the ground-truth heatmaps for row 1, 2, 3 respectively. The region colored light-red corresponds to localization information, while dark red region corresponds to distinguishing features. For example, column 1 and 2 can be distinguished by the presence of the bar across the circular cell. Columns 4, 5, 6 differ only in their dominant colors of their rectangular borders, and thus the distinguishing features are their borders.

mentioned sanity checks in [15].

The *pointing game* has been used for evaluating the performance of localization algorithm. In XAI evaluations, they are used in [16, 17]. Looking more closely, this metric is "extremely loose" because it considers a hit whenever the maximum value lies inside the desired ground-truth mask region (even a single pixel counts). Using a single pixel to evaluate the quality of explanations may be questionable, unless the mask area is always significantly small compared to the image size. By contrast, we measure recall and precision over a large regions of pixels in the images.

Another experimental study on the quantitative evaluations of heatmap-based methods [18] listed several issues in existing methods as well (e.g. faithfulness has out-of-distribution issues). It introduces IntIoSR, again another metric to compute overlap with ground-truth heatmaps; our paper again differs from such "localization only" approach, since we distinguish localization from discriminative features. Likewise, [19] performs such localization measurement using Relevance Mass Accuracy, though it also includes Relevance Rank Accuracy. Our metric is similar in the sense that we compute pixelwise hit/miss.

C. Other XAIs and Applications

XAI methods that are not focused on generating heatmaps have also been developed. This paper is mainly concerned with quantitative heatmaps comparison, but we may still benefit from different types of evaluations of XAI performance. Local interpretable model-agnostic explanation (LIME) [20] is introduced to find a locally faithful interpretable model that represents well the model under inspection, regardless of the latter's architecture (i.e. is agnostic). By comparing LIME with obviously interpretable models such as decision trees and sparse logistic regression, in particular using recall value, the quality of feature importance obtained using LIME can be assessed. Experiments on Concept Activation Vectors (section 4.3 of [21]) include quantitative comparison of the information used by a model when a ground-truth caption is embedded into the image. In some cases, the caption is used by the model for decision-making, but in other cases, only the image concept is used. Furthermore, human-subject experiments are also conducted to test the importance of the saliency mask, showing that heatmaps only marginally help human make decision. It also shows that heatmaps can even be misleading. There has also been other similar sentiments, for example in the caption
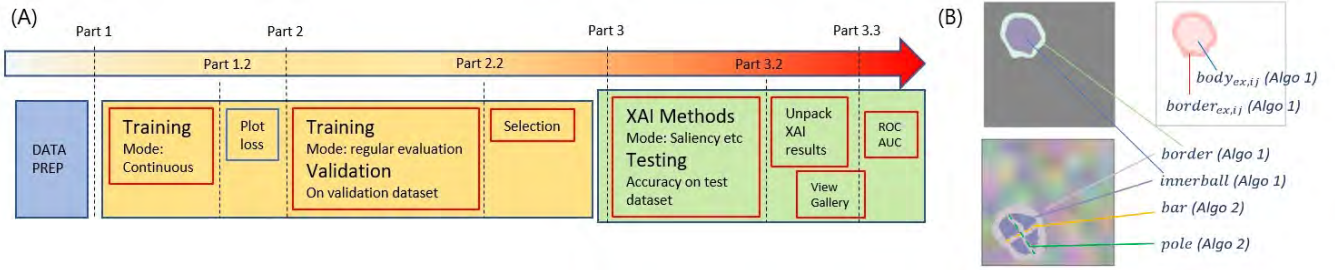
Fig. 2: (A) Workflow illustrating the process starting from data generation to the generation of heatmaps gallery. (B) Illustration showing how parts of the cells are generated in algorithms 1 and 2.

of fig. 2 in [22], where the utility of a heatmap is called into question. Network dissection frameworks [23, 24, 25] have been used to evaluate DNN by counting the number of *unique detectors* and measuring the IoU between feature maps and segmentation mask. There are other methods not covered here, and we refer the readers to survey papers like [26, 27, 28].

Applications of XAI methods have emerged in several fields e.g. in network traffic analysis [29, 30] and the medical field, where evaluation of heatmaps has been performed in different ways. A study on MRI-based Alzheimer's disease classification [31] computes the L2 norm between average heatmaps generated by different XAI methods and compares the performance of three other different metrics. Ground-truth heatmaps are sometimes available, for example in the diagnosis of lung nodules [32] where recall values can be directly computed between the reference features (ground-truth) and the heatmaps generated by different XAI methods. Different kinds of ground-truth have been obtained using specialized method, such as NeuroSynth in [33] for analyzing neuroimaging data. Some parts of the evaluation are qualitative (such as group-level evaluation), though the paper uses F1-score to evaluate the heatmap, thus naturally including recall and precision concepts in the evaluation. Other applications of XAI methods, especially heatmaps, in the medical field are for example [34, 35, 36, 37, 38, 39, 40, 41, 42].

III. DATA AND METHODOLOGY

This section describes the workflow starting from data generation, network training, network performance evaluation, heatmap generation and evaluating generated heatmaps with common quantities. The workflow is shown in fig. 2(A), closely following the sequence of commands run in the package of python codes [1] provided. Some details, such as the algorithms needed to generate each sample data, can be traced from the tutorials available as jupyter notebook included in the package of codes.

A. Dataset

We provide algorithms that can generate dataset on demand, illustrated in fig. 2(B). Sample data can be seen in fig. 1, where the top three rows show the image data and the last three rows are the corresponding ground-truth heatmaps. Ten

[1]https://github.com/etjoa003/explainable_ai/tree/master/xai_basic

---

Algorithm 1: build_basic_ball_body(x_0, y_0, r, t etc) for type 0 cell, see fig. 1. *More descriptions in appendix.

$x, y \leftarrow meshgrid$
$d \leftarrow \sqrt{x^2 + (y/y_s)^2} + noise$
$border \leftarrow (d \le r) * (d \ge r - t)$
$innerball \leftarrow (d < r - t)$
rotate by θ
shift center to (x_0, y_0)
$ball \leftarrow border + innerball$
make_explanation(){
for all pixels (i, j) **do**
  $border_{ex,ij} \leftarrow \frac{1}{3}\sqrt{\Sigma_c(border)^2_{ij,c}} \ge th_d$
  $body_{ex,ij} \leftarrow \frac{1}{3}\sqrt{\Sigma_c(inner)^2_{ij,c}} \ge th_l$
  $body_{ex,ij} \leftarrow body_{ex,ij} * (1 - border_{ex,ij})$
  $heatmap_{ij} \leftarrow border_{ex,ij} * 0.9 + body_{ex,ij} * 0.4$
end for
}

Algorithm 2: build_ccell_body(x_0, y_0, r, t, t_b etc) for type 1 or 2 cell, see fig. 1. *More descriptions in appendix

build_basic_ball_body(x_0, y_0, r, t etc)
$x, y \leftarrow x + noise, y + noise$
create_skeleton(){
  $bar_x = (x - x_0 \le r) * (x - x_0 \ge -r)$
  $bar_y = (y - y_0 \le t_b/2) * (y - y_0 \ge -t_b/2)$
  $bar = bar_x * bar_y$
  if type 2 **then**
    $pole_{y1} = (y - y_0 \le r * v_s)$
    $pole_{y2} = (y - y_0 \ge r * v_s)$
    $pole_{x1} = (x - x_0 \le t_p/2)$
    $pole_{x2} = (x - x_0 \ge -t_p/2$
    $pole = pole_{y1} * pole_{y2} * pole_{x1} * pole_{x2}$
    $pole_{pos} = (pole \ge 0) * (1 - bar \ge 0)$
    $bar = bar + pole * pole_{pos}$
  end if
}
rotate ball and bar by θ
shift center of ball and bar to (x_0, y_0)
make_explanation()

different classes of cells are shown along the columns. Types 0,1,2 are circular cells with border (algo. 1), with a bar (or minus sign) and with a plus sign (algo. 2) respectively. Types 3,4,5 are rectangular cells with different dominant colors. Types 6,7,8 are circular cells with one, three and eight tails respectively. The last class does not contain any cell. Three types of backgrounds are given to increase the variation of dataset, as shown separately in the first three rows of the same figure.

The algorithms are generally designed with simple formulas that correspond to mathematically well-defined shapes, as illustrated in fig. 2(B). For example, in Algorithm 1, type 0 cell is a simple coloured elliptical cell with border coloured differently. We can see the cell body reflected as $innerball = d < r - t$, describing the inner part of a circle. In algorithm 2, bar and pole reflect the shapes of minus and plus inside cells type 1 and 2. Noises etc will distort the shape further while retaining their main features. In both algorithm 1 and 2, make_explanation() function is defined to highlight the features (dark red) and the localization (light red). They are easily obtained since the cells are obtained part by part.

The ground-truth heatmaps h_0 have been designed to mark features that distinguish all the classes in a way that is as unambiguous as possible, subject to human judgment. Admittedly, there may not exist a unique unambiguous way of defining them. Where appropriate, the heatmaps could be readjusted by editing the heatmap generator classes in the package of codes. The heatmaps are shown in fig. 1 row 4 to 6. With this dataset, fair comparison between heatmaps generated by different XAI methods can be performed. In this particular implementation, each ground-truth heatmap h_0 is normalized to $[-1, 1]$, thus heatmaps to be compared to h_0 are expected to be normalized to $[-1, 1]$ as well. Each h_0 consists of an array of values of size (H, W) with three distinct regions (1) regions with value 0 (shown as white background) corresponding to regions that should not contribute to the neural network prediction, or non-features (2) regions with value 0.4 corresponding to localization (shown as light red region) and (3) regions with value 0.9 to delineate discriminative feature (shown as dark red region), where discriminative feature is also qualitatively considered part of localization. In this paper, two other distinct regions are defined symmetrically. They are -0.4 and -0.9 regions, to accommodate the fact that some heatmap methods have been interpreted in such a way that negative regions (shown as blue color in this paper) are considered as regions contributing *against* a given decision or prediction [6]. For our dataset, the ground-truth does not contain any such information that contributes negatively, although, as will be shown later, some XAI methods still do generate negative values.

For this paper, training, validation and evaluation datasets are prepared in 32, 8 and 8 shards respectively, each shard containing 200 samples uniform randomly drawn from the 10 classes. In another words, in total, the datasets contain 6400, 1600 and 1600 samples respectively. The dataset is prepared in shards for practical purposes, for example, to prevent full restart in case of interruption of data downloading and caching, and to facilitate more efficient process of training in evaluation

mode indicated in fig. 2(A) as part 2.

B. Training

After the data is cached or saved, the process starts with *training in continuous mode*, indicated as part 1 of the workflow fig. 2(A). In this mode, pre-trained models are first downloaded from Torchvision and modified for compatibility with pytroch Captum API. The three pre-trained models used are AlexNet [43], ResNet34 [44] and VGG [45], corresponding to workflow 1, 2 and 3 in the codes. In this phase, training proceeds continuously for the purpose of fine-tuning the models to our current data. The number of epochs and batch size are specified in table I. Adam optimizer is used with learning rate $lr = 0.001$ for ResNet but $lr = 0.0001$ for AlexNet and VGG, and the same weight decay 10^{-5} is used for all. Plot of losses against training iterations (not shown in this paper) is saved as a figure in part 1.2 of the workflow. We refer to the model trained after this phase as the *base model*.

The next phase is the *training in regular evaluation mode*, indicated as part 2 in fig. 2(A). The training uses the same optimizer as the previous phase, and the number of epochs used are also shown in table I. Evaluation is performed every 4 training iterations; more accurately, this part is known as validation in machine learning community, separate from the final evaluation. Each validation is performed on a shard randomly drawn from the 8 shards of the validation dataset. We set the *target accuracy* to 0.96. If during validation, the accuracy computed on that single shard exceeds the target accuracy, the training is stopped and evaluation on all validation data shards is performed. The total validation accuracy is used to ensure that the validation accuracy on a single shard is not high by pure chance. While the total validation accuracy can be slightly lower, our experiments so far indicate that there is no such problem. Furthermore, only ResNet attained the target accuracy within the specified setting. For AlexNet and VGG, 0.96 is not exceeded throughout and early stopping mechanism is triggered to prevent unnecessarily long, unfruitful training; note that, fortunately, the total accuracy when evaluated on the final evaluation dataset is still very high, as shown in table I. The early stopping mechanism is as the following. Whenever validation on a single shard does not achieve the target accuracy but (1) if there is no improvement in the validation accuracy, then early stopping counter n_{early} is increased by one (2) if there is improvement in the validation accuracy, then $n_{early} \to n_{early} \times r_f$, where $r_f < 1$ is the refresh fraction, so that the process is given more chance to train longer. If n_{early} becomes equal to the early stopping limit, n_l, training is stopped.

We repeat the above process of training in regular evaluation mode 4 other times starting from the base model, and thus we have a total of 5 branch models. Note that r_f and n_l are set so that AlexNet and VGG can be trained for longer period (shown in table I), since they both achieve lower accuracy performance than ResNet if given the same number of epochs, n_l and r_f. This is possibly because (1) larger batch size means fewer iterations per epoch and (2) improvement in accuracy is inherently slower, considering that ResNet has been known
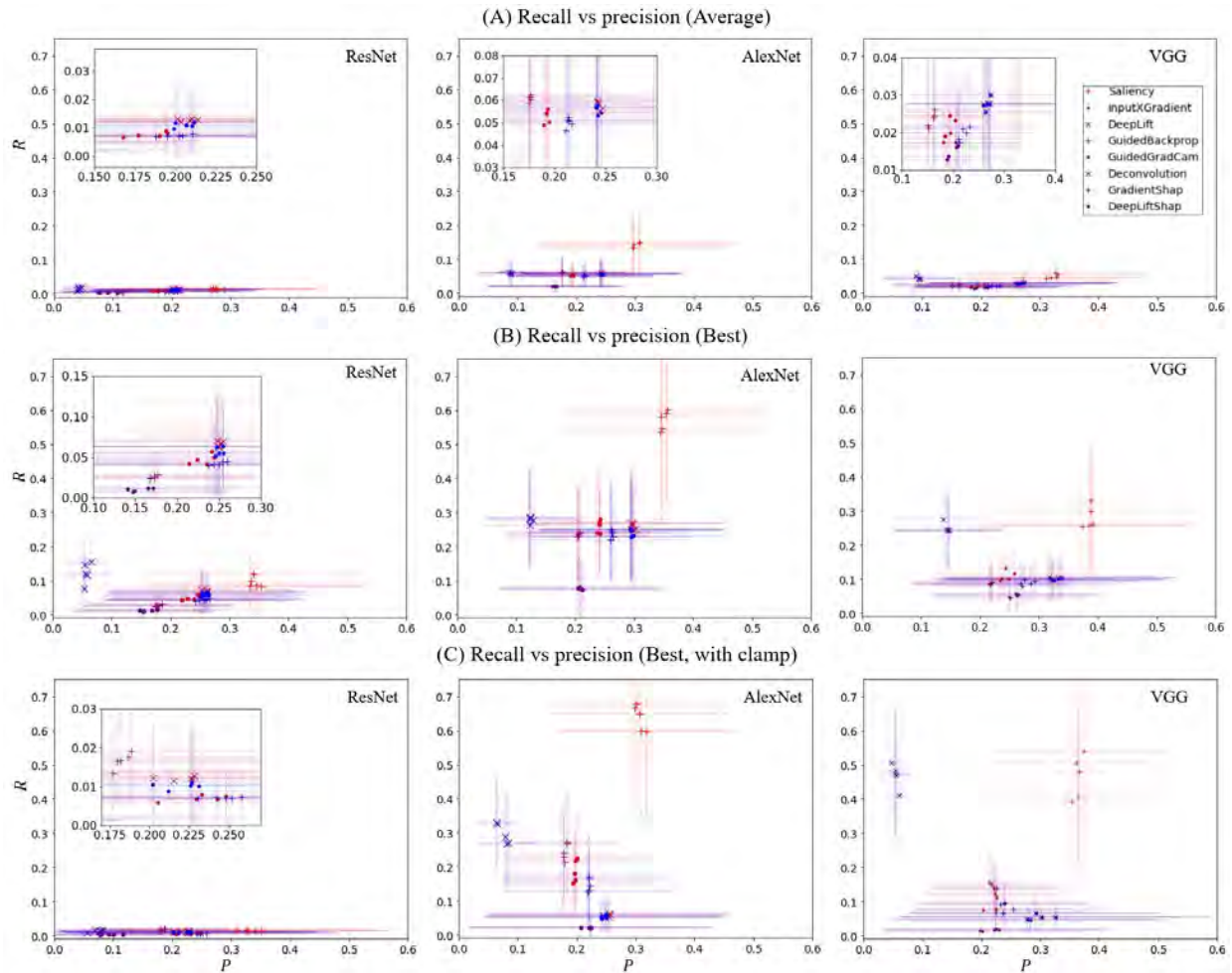
Fig. 3: Recall and precision scores of five-band stratified heatmaps compared to ground-truth for ResNet, AlexNet and VGG and 8 different XAI methods. For all, higher recall and precision values are better, i.e. points located towards top-right are better. (A) Average and (B) maximum values of recalls and precisions (over soft five-band thresholds) of each sample of evaluation dataset are collected, then averaged over all these samples to be shown as individual points (P, R) in the plot. (C) is the same as (B), except the scores are obtained after clamping process. Vertical and horizontal bars are the standard deviations over all samples of evaluation dataset correspondingly. Insets show zoom-in on selected regions.
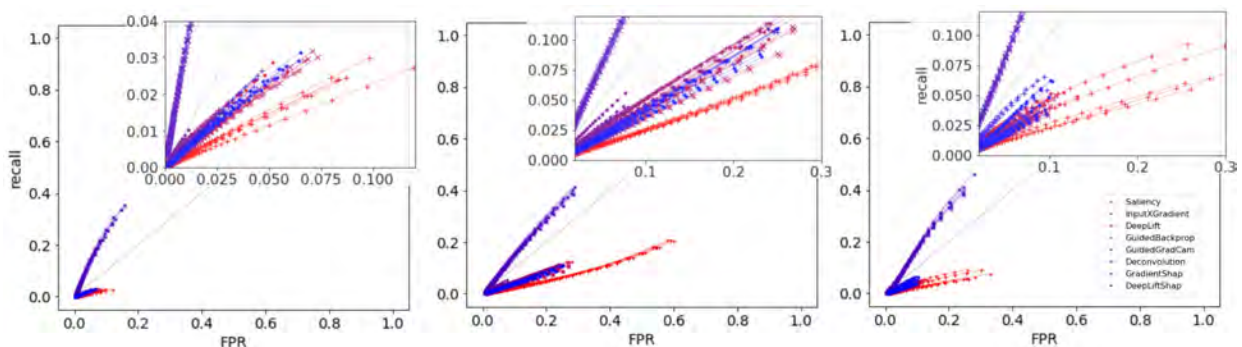

Fig. 4: ROC curve for ResNet (left), AlexNet (middle) and VGG (right) for 8 different saliency methods (see legend, bottom-right), each XAI method applied on test datasets separately for 5 branch models. Both FPR and recall do not necessarily reach 1.0 as the thresholds are adjusted in multi-dimensional space. Most methods under our the experimental conditions specified here lie under traditionally poor ROC region.

TABLE I: Training settings and performances on the ten-classes data. TC and TR denote trainings in continuous and regular evaluation mode respectively. $< Acc. >$ denotes average accuracy over 5 models branched from the base model. n_{batch} is the batch size, n_{ep} no. of epochs. Image shapes are (H, W) where $H = W$.

			TC	TR			Eval
	H,W	n_{batch}	n_{ep}	n_{ep}	n_l	r_f	$< Acc. >$
ResNet	512	4	2	4	24	0.4	0.951
AlexNet	224	16	16	16	48	0.3	0.980
VGG	224	16	4	4	48	0.3	0.986

to generally perform better. Here, comparing accuracy of prediction in a precise manner is not very meaningful, as we are focusing on the heatmaps later. Perfect accuracy is not attained and there is no guarantee that DNN will ever attain a perfect performance. However, we take this opportunity to compare heatmaps between correct and wrong predictions (see discussion section later). Finally, note that there is no need for k-fold validation here since the validation dataset is completely separate from the training dataset.

C. Evaluation and XAI implementation

This part corresponds to part 3 of fig. 2(A), where heatmaps h are computed using the following XAI methods available in pytorch Captum API: Saliency [1], Input*Gradient [46] (reference based on Pytorch Captum's page), DeepLift [9], GuidedBackprop [47], GuidedGradCam [8], Deconvolution [48], GradientShap [49], DeepLiftShap [49]. Integrated-Gradients [13] has been excluded as it is comparatively inefficient with ResNet. Note also that the original implementation of ϵ-Layerwise Relevance Propagation (LRP)[2] [2] has been shown to be equivalent to gradient*input or DeepLIFT depending on a few conditions [9]. For all heatmaps, we compute the heatmaps derived from the predicted values, not the true values (for some XAI methods, explanation can be extracted from the probability of predicting not only the correct class, but also other classes). The following is the sequence of processing leading to the final results.

Channel adjustments. Each heatmap h, which has (C, H, W) shape (C=3 for 3 color channels), is compressed along the channels to (H, W) by *sum-pixel-over-channels*, where the values are summed pixel-wise along all channel, i.e. $s_c(h_{ij}) = \Sigma_{c=1,2,3} h_{ij,c}$ when written component-wise. This is so that it can be compared with h_0 of shape (H, W). Normalization to $[-1, 1]$ is also performed in the following *channel adjustment* process: $h \rightarrow s_c(h) \rightarrow h/max(|h|)$. $max(|h|)$ is the maximum absolute value over all pixels in that single heatmap. The practice of summing over channels can be seen, for example, in LRP tutorial site [50] and [6].

Five-band stratification. Adjusted heatmaps h will subsequently be evaluated using *five-band score*, where each pixel needs to be assigned one of the five values $0.9, 0.4, 0, -0.4, -0.9$ described in section III-A. The value 2 is designated for discriminative feature, 1 for localization, 0 for irrelevant background, while -1 and -2 are symmetrically
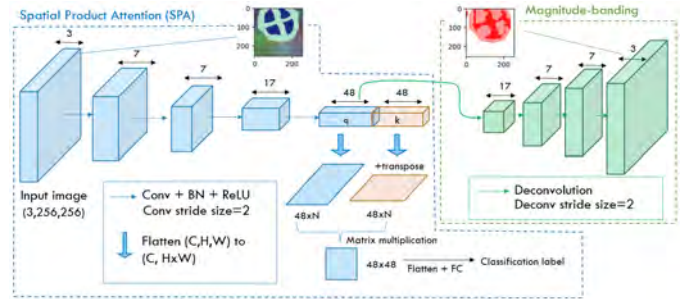
Fig. 5: Spatial Product Attention (SPA) and Magnitude-banding. SPA is a classification model with query q and key k abstraction (like transformer). Magnitude-banding is achieved through deconvolution layers on query q (green dotted box).

defined for negative contribution to model prediction or decision. Recall that our ground-truth heatmaps h_0 pixels have been assigned one of the following values 0, 0.4 and 0.9. Regardless of the intermediate processing of the heatmap h, the mapping for h_0 is always such that $0.9 \rightarrow 2$, $0.4 \rightarrow 1$ and 0. To map h, which has been normalized to $[-1, 1]$ by now, a threshold of the form $t = [-t_{(2)}, -t_{(1)}, t_{(1)}, t_{(2)}]$ is used, so that for each pixel h_{ij}, a transformation we refer to as *five-band stratification* is performed in the following manner: $h_{ij} \rightarrow 2$ if $h_{ij} > t_{(2)}$, to 1 if $h_{ij} \in (t_{(1)}, t_{(2)}]$, to 0 if $h_{ij} \in (-t_{(1)}, t_{(1)}]$, to -1 if $h_{ij} \in (-t_{(2)}, -t_{(1)}]$ and to -2 if $h_{ij} \leq -t_{(2)}$. Bracketed sub-script here is used to denote the component of t if it is regarded as a vector for notational convenience later. Up to this point, we have $S^5(h_0), S_t^5(h)$ where S^5 denotes the five-band stratification.

Five-band score. After stratification, for each heatmap, we compute accuracy A, precision $P = \frac{TP}{TP+FP+\epsilon}$, $R = \frac{TP}{TP+FN+\epsilon}$, where accuracy is the fraction of correctly assigned h_{ij} pixel over the total number of pixels, TP is the number of true positive pixels, FP false positives, FN false negatives and $\epsilon = 10^{-6}$ for smoothing. TP is slightly different from TP used in binary case. We only count TP when $h_{0,ij} \neq 0$ and $h_{ij} = h_{0,ij}$, i.e. we use the stringent condition where the labels for localization and features must be correctly hit to achieve a true positive. Likewise, FP is counted when $h_{0,ij} = 0$ and $h_{ij} \neq 0$ plus $h_{0,ij} \neq 0$ and $h_{ij} \neq h_{0,ij}$ whereas FN when $h_{0,ij} \neq 0$ and $h_{ij} = 0$. To plot receiver operating characteristics (ROC), false positive rate $FPR = \frac{FP}{FP+TN+\epsilon}$ is also computed, where TN is the number of true negatives $h_{ij} = h_{0,ij} = 0$.

Soft five-band scores. As seen, the threshold defined above is sharp, and the value near any of the thresholds $\pm t_{(i)}$ might not be properly accounted for. We thus instead use soft five-band scores, where the metrics are collected for different thresholds. More precisely, for the k-th data sample, we obtain $(A, R, P)_{t_m}^{(k)}$ for $t_m = [-0.5 + md, -0.3 + md, 0.3 - md, 0.5 - md]$ where $d = 0.005$, $m = 0, 1, \cdots, n_{soft}$, and $n_{soft} = 55$ after comparing the stratified ground-truth $S^5(h_0)$ with $S_{t_m}^5(h)$, where h has undergone *channel adjustment* process previously described. The best and average values of $X = A, R, P$ for sample k over the different thresholds, $X_{avg}^{(k)} = \frac{1}{n_{soft}} \Sigma_{t_m} X_{t_m}^{(k)}$ and $X_{best}^{(k)} = max_{t_m}\{X_{t_m}^{(k)}\}$ respec-

tively, are then saved sample by sample into a csv file in the XAI result folder for analysis in the discussion section. These values are identified by their positions among the shards, the predicted class and the true class. Note: the choice of t_m is chosen such that localization is found at the "middle range" between $\sim [0.3, 0.5]$ and discrimination at the "higher range" i.e. stronger signals.

Receiver operating characteristic. To compare the performances of different XAI methods mentioned above, ROC is also obtained as shown in fig. 4. For each threshold t_m, mean values of $\{FPR_{t_m}^{(k)}\}$ and $\{R_{t_m}^{(k)}\}$ over all samples in the evaluation datasets contribute to a single point in the figure. Unlike the usual binary ROC, changing thresholds in the multi-dimensional space we defined does not guarantee the change from FP to TP (or vice versa). For example, a point that begins as FN that predicts label 0 can become TP or FP if the true value are 1 and 2 respectively when the 0 thresholds are lowered. Hence, we will not always obtain a curve that starts with $(0, 0)$ and ends with $(1, 1)$ in the ROC space, unlike the usual ROC curve. Regardless, by simple understanding of rate of change of FPR and recall, the usual rule of thumb that assigns steeper increase in recall to better ROC quality should still hold. Mathematically, the more optimal ROC curve lies nearer the top-left vertices of the convex hull formed by the points. There has been studies on multi-dimensional ROC curve with its "area under volume" [51, 52], though the difficulty of observing them makes them unsuitable for visual comparison here. With the definition of TP, FP, TN, FN above, we have instead created pseudo-binary conditions.

D. Magnitude-banded Class Activation Mapping

The central idea in this paper has been highlighted in the previous sub-section: the quality of XAI methods is measured in a straightforward manner, specifically by measuring pixel-wise hit/miss against ground-truth heatmaps/attributions using metrics such as recall and precision. We have proposed five-band score to quantify different levels of explanation, and we have assigned 0.4 as the value corresponding to local-ization and 0.9 to more prominent, distinguishing features, i.e. pixelwise attribution is banded by magnitude. In this sub-section, we present the mabCAM (Magnitude-banded Class Activation Mapping), a general arrangement of deep neural network modules intended to produce high-quality magnitude-banded heatmaps.

Before going into details, we briefly describe the components used to generate mabCAM heatmaps. A series of deconvolution layers (denoted mab) will be applied to feature maps at the end of a convolution layer. In this paper, mab is applied to feature maps from the last convolution layer, similar to the original CAM which computes the weighted sum of features in the last convolution layer. Our deconvolution layers are not attached to activation functions or batch normalization layers, and each layer has stride 2. The goal is to expand the dimensions of feature maps to approximately the same height and weight as the original input, just like the decoding part of a generic variational auto-encoder. The number of channels in deconvolution layers correspond to the number of channels in the convolution layers preceding them (ideally, we wish to use as few as possible parameters just like CAM, but it did not work well enough). The final deconvolution layer has 3 channel outputs corresponding to 3 classes of labels: non-feature, localization and features, corresponding to heatmap values $0, 0.4, 0.9$ respectively. This arrangement allows optimization via the minimization of cross-entropy loss similar to semantic segmentation.

Some remarks: (1) ideally, there are 5 channel outputs corresponding to $0.9, 0.4, 0, -0.4, -0.9$ rather than 3 channels, the two missing classes being the "negative attributions". However, in our dataset, the design does not incorporate any features that contribute in a "negative" way, not to mention the fact that it is not yet clear what role the negative attribution values play in existing XAI methods, at least w.r.t the negative values (blue patches) that we observe in heatmaps produced by these methods on our dataset. (2) Semantic segmentation can be very costly, especially when there are many classes. There need to be as many output channels as the class labels, and each ground-truth label may need to be marked manually. Our procedure automatically generates similar pixelwise label, which is convenient. Note that using our magnitude-banding framework on other datasets still requires some manual la-beling of features and localization areas. Unlike semantic segmentation, our magnitude banding only requires 3 classes (at most 5, if the negative attribution is included), thus the manual labeling effort is not as taxing.

The mabCAM design. The implementation in this paper is geared towards a small model with only 82376 parameters (compared to ResNet50 with over 20 million parameters). Regardless, the idea is general and widely applicable, as the following. First, we introduce the Spatial Product Attention (SPA), as shown in fig. 5. SPA is a series of convolution layers set up to quickly downsize the feature maps, which are then split to yield query q and key k abstractions, like the attention mechanism in a transformer encoder [53], analogous to a retrieval system. In SPA, q is designed as the components that will later be fine-tuned to produce our attribution heatmaps. For image classification, the key and query are flattened, followed by a matrix multiplication qk^T, resulting in a 48 by 48 feature map (note that each of its pixel is already influenced by all pixels in the input image upstream). With this arrangement, the size of SPA's pre-FC feature map is independent of the input image size, as long as the input image size is not too small; thus we do not need global average pooling commonly used in a CNN.

The component that generates mabCAM heatmap is mab, or the the magnitude-banding module, as seen in dotted green box of fig. 5. It is a series of deconvolution layers, and no special novelty is required here. Each layer also has stride=2, resulting in heatmap dimensions approximately equal to the input image. As a whole, $y, h = mabCAM(x)$ where x is the input image, y the raw predicted class label and h the raw predicted heatmap of shape $(3, h', w')$. The predicted feature maps is computed by, first, $h_1 = argmax(h, dim = 0)$ and finally $attr = (h1 == 1) * 0.4 + (h1 == 2) * 0.9$. To train the model, cross entropy loss is minimized for both class labels and magnitude-banding labels, i.e. $loss = CELoss(y, y0) +$

$\lambda_{mab}CELoss(h, h0)$. Note: magnitude-banding labels mapping is $0 \rightarrow 0, 0.4 \rightarrow 1, 0.9 \rightarrow 2$. We use Adam optimizer with learning rate 0.001 and $\beta = (0.5, 0.999)$, batch size 16 and mab regularization $\lambda_{mab} = 5$. No special hyperparameter tuning is performed. Training data consists of 4096 samples uniformly drawn from the 10 classes of the synthetic cells previously described, while validation and test data consist of 1024 samples each. Additionally, smaller 3-class demo is also available in the appendix. We allow training to last for at most 128 epochs, though the training is terminated if the validation accuracy reaches 0.975. Once training is completed, five-band scores are computed similar to the way we compute them for other XAI methods. The difference lies in the soft thresholding: since mabCAM is designed to be compatible with the discrete heatmap values, we do not need to perform soft-thresholding across 56 soft-steps. The results from mabCAM will be discussed at the end of the next section.

IV. DISCUSSION

A. Recall vs Precision

We provide recall vs precision scores as shown in fig. 3. Each point in the plot corresponds to an XAI method, for example Saliency, applied on a single branch of the corresponding model trained from the base model. There are 5 points per method as we have trained 5 branches per architecture. Naturally, the higher P and R are, the better is the XAI method. Each point can be denoted by (R_{stat}, P_{stat}), where $X_{stat} = \frac{1}{N}\Sigma_{k=1}^{N}X_{stat}^{(k)}$, $X = R, P$, $stat = avg, best$ and N is the number of data sample in the evaluation dataset. Thus fig. 3(A) is a plot of R_{avg} vs P_{avg} for ResNet, AlexNet and VGG respectively. Likewise, fig. 3(B) is R_{best} vs P_{best}. After qualitative assessment of some of the generated heatmaps, we perform similar analysis by applying clamping to heatmap values after the first normalization process, following roughly the idea in [54]. In other words, the *channel adjustment* process described in the previous section is changed to $h \rightarrow h/max(|h|) \rightarrow C_{[c_1,c_2]}(h) \rightarrow s_c(h) \rightarrow h/max(|h|)$ where $C_{[c_1,c_2]}(h_{ij}) = c_2$ if $h_{ij} \geq c_2$, $C_{[c_1,c_2]}(h_{ij}) = c_1$ if $h_{ij} \leq c_1$ and otherwise $C_{[c_1,c_2]}(h_{ij}) = h_{ij}$. A different set of soft thresholds has been used to match the clamping process as well, with $t_m = [-0.9+md, -0.5+md, 0.5-md, 0.9-md]$ where $d = 0.01$, $m = 0, 1, \cdots, n_{soft}$, and $n_{soft} = 40$ with the clamping threshold given by $[c_1, c_2] = [-0.1, 0.1]$. Fig. 3(C) is thus the same as fig. 3(B), except with clamping process applied, where we do observe some changes in the precision and recall scores, more notably for AlexNet and VGG, though not necessarily better.

Recall scores are generally low in most of the points in fig. 3, indicating high FN. The first obvious cause is the fact that most XAI methods in all architectures appear to assign 0 values to regions that contain either localization pixels or discriminative features. For example, fig. 6 shows the heatmaps from different channels R, G and B (extracted before summing pixel over channels). The heatmaps generally appear granular and non-continuous, having many white pixels in between the red pixels, thus contributing to false negatives. Furthermore, most of the inner body of the cells (represented
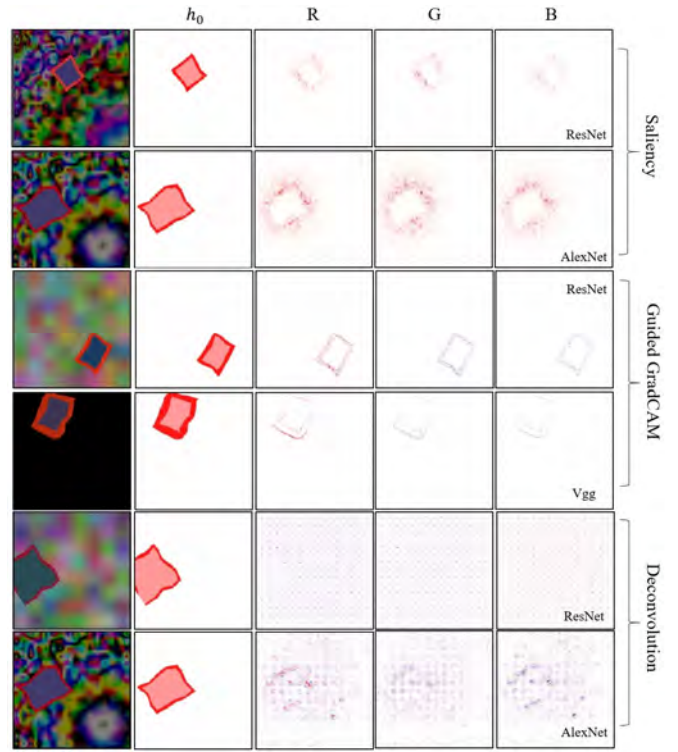

Fig. 6: Visual comparison of heatmaps generated by Saliency, Guided GradCAM and Deconvolution. Different color-channel responses are shown under R, G and B columns respectively, with the original image in the left-most column and ground-truth h_0 in the second left-most column. All heatmaps above are obtained from correctly predicted samples.

by light red color in the ground-truth h_0) is completely unmarked by most of the XAI methods, contributing to very large amount of false negatives. The highest recall values in fig. 3(A) are attained by Saliency applied on AlexNet. This is consistent with visual inspection of the heatmaps across different methods and architectures, because Saliency assigns a lot more red pixels in relevant regions while other methods often assign blue pixels (negative values) in unpredictable manner and highlight only the edges.

Similar to the heatmaps shown in fig. 6 produced by Guided GradCAM applied on VGG, many of the XAI methods only highlight the edges of the cell borders, sometimes faintly. As such, comparatively high recall values for Saliency can be qualitatively accounted for by the halo of high-valued heatmap encompassing the relevant area, although not in a very precise and compact manner. Deconvolution, on the other hand, has relatively higher recall scores due to the large amount of artifact pixels. The quality of its heatmaps has been therefore undermined, reflected as low precision score. Other methods such as guided GradCAM are still capable of highlighting some of the relevant regions, and, to reiterate, many of them tend to highlight the edges as seen in the heatmaps in the supp. material. Also, AlexNet tends to produce denser heatmaps than the other two, giving rise to slightly higher recall scores than VGG, while the average scores for ResNet are very low. Depending on the context, different XAI methods can
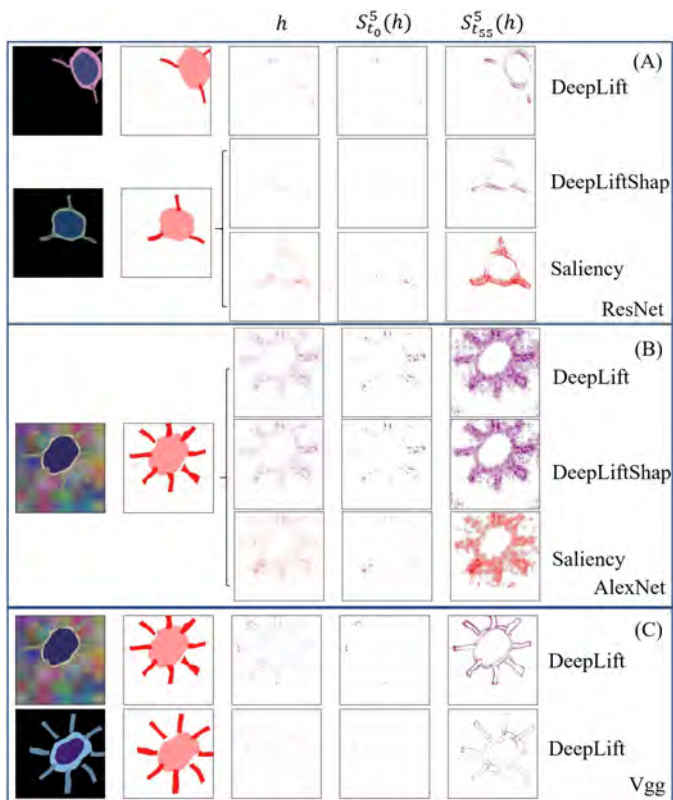
Fig. 7: Visual comparison of heatmaps generated by DeepLift and DeepLiftShap, with Saliency for comparison. Column h is obtained after *summing pixel over channels*. Columns $S_{t_0}^5(h)$ and $S_{t_{55}}^5(h)$ are obtained after five-band stratification using the first and last thresholding described in section III-C, where $t_0 = [-0.5, -0.3, 0.3, 0.5]$ and $t_{55} = [-0.225, -0.025, 0.025, 0.225]$. (A) Visualization of how DeepLift and DeepLiftShap on ResNet generally score slightly lower in recall scores than Saliency (B) DeepLiftShap and DeepLift appear to produce similar heatmaps for VGG and ResNet, though the SHAP variant appears to remove some artifacts in Alexnet. Negative attribution values (blue) are difficult to interpret. (C) Heatmaps for correct prediction of cells from the same class, cell type 8, generated using DeepLift applied on VGG. See more in appendix.

be the better choice based on their strength and weaknesses. More ideally, existing interpretations may need to be defined w.r.t context of the problems solved by the particular DNN e.g. when DNN performs image classification, users need to be informed whether localization is considered "part of explanation" or specific parts of an image contributes to the explanation.

Differences in responses to color channels are also observed. Saliency method appears as positive values (red) in all channels as shown in fig. 6, although type 3 cell has only 1 color channel whose input signal is strong because it has border whose pre-dominant color is red; in the implementation, the normalized border color is roughly $(0.8, 0.1, 0.1)$ with small uniform random perturbation. On the other hand, Guided GradCAM marks green and blue channels with negative val-

ues. If they are to be interpreted as negative contribution, the interpretation will be consistent. But when the heatmaps are summed over channels as we have done, the offsetting effect of the negative values become questionable. In other methods, such color responses are variable. For example, input*gradient for AlexNet do appear to exhibit color responses as well (not shown), although the quality is highly variable too. It is thus difficult to strongly recommend any one method specializing on color-detection, even for guided GradCAM.

Interpretation of heatmap values. Fig. 7 shows in column h the heatmaps obtained after *summing pixel over channels*, one of the earlier processes in the previous section. The figure shows the effect of soft five-band stratification as well, which demonstrates that the appropriate selection thresholding does affect the scores. In previous section, we addressed this by distinguishing between the best and average of recall and precision values over the soft thresholds, which is the main purpose of fig. 3(B). The effect of threshold change is variable across different XAI methods. If we focus on recall scores, from visual inspection of fig. 7(B), the XAI community may need to revise the idea of negative values in heatmaps. Clearly, DeepLift and DeepLiftShap examples show that they will score much better recall if we take the absolute values of the heatmaps and apply the same process from stratification to the computation of five-band scores.

SHAP and DeepLift. When SHAP is applied to DeepLift, the effect appears to be background artifact removals, thus confining non-zero heatmap pixel values to more relevant regions (fig. 7(B) and supp. materials). Blue pixels (negative values) mark some of the correct areas that we regard as discriminative features, but interpreting the blue pixels for these methods as negative contribution seems to be inappropriate. Applying absolute value to the negative pixels may improve its recall scores etc; generally, more investigation on the signals activated by the background may be necessary. Furthermore, we need to point out that heatmap shapes could be inconsistent even if the same classes of cells are correctly predicted, as shown in fig. 7(C). The figure shows two heatmaps of different qualities generated by DeepLift for VGG for cell type 8 that are correctly predicted. It may be tempting to make guesses regarding possible reasons, for example, backgrounds might play a large role in distorting the heatmaps.

From observing fig. 3 and many heatmaps, for examples the figures in the appendix, it is tempting to deduce that deeper networks (AlexnNet shallowest, followed by VGG, then ResNet deepest) tend to produce heatmaps that are more sensitive to the edges but cover less thoroughly the bulk of discriminative features and localization regions. To test this, we conduct a test on AlexNet modified by systematically adding more and more convolutional layers, trained and then evaluated for its precision vs recall in the same manner as before. The number of layers added are 1, 2, ..., 8, and the plot is made by computing mean values of recall and precision like before, except that the points are collected separately based on the predicted values (whereas in the previous section, averages are taken over all test samples regardless of predicted values). The expectation is for the precision and recall values to be nearer to 1 (more towards top right of the plot) for the modified

AlexNet with less additional layers. However, as shown in appendix fig. 1, this does not appear to be the case.

ROC curve. ROC plots in fig. 4 show that most heatmap methods tested lie on traditionally poor ROC regions. There appears to be trade-offs between higher recall values (which is good) and higher FPR (which is bad), most prominently shown by Saliency method. Deconvolution appears to be the best, as it has the greatest rate of increasing recall compared to FPR. However, this is misleading, since deconvolution starts with many FP predictions in all three architecture, as shown by the grid-like artifacts in fig. 6. This causes FP to change more quickly, and the ROC fails to make good comparison between deconvolution and other methods. Saliency tends to "over-assign" the heatmap pixels around the correct region; consider fig. 6, 7 and appendix, compare it to, for example, Guided GradCAM, DeepLift. Unlike DeepLift and DeepLiftShap, Saliency ROC shows higher recall because of more correct assignments of positive (red) values, but also higher FPR because of the assignment of positive values in supposedly white regions. Guided GradCAM appears to have some difficulty improving through the change of thresholds. Considering the way *sum-pixel-over-channels* is performed and its color-channel sensitivity, its performance might have suffered through incompatible heatmap pre-processing. ROC for other methods do not provide sufficiently distinct trends that favor the adoption of one method over another. There may be a need to investigate the different ways soft-thresholding can be performed for specific XAI methods to at least bring the ROC curves to traditionally favorable regions.

Other observations. For images of type 9 (no cell present), generally we see heatmaps in the form of artifacts appearing as well-spaced spots, forming lattice (see appendix fig. 40 etc), similar to the heatmaps from deconvolution method in fig. 6. In some cases, for example see orange boxes in fig. 8 (from appendix fig. 16 row 2), we can see that DeepLIFT is able to "provide" the correct reasoning for wrong prediction. In that figure, type 2 cell that has shape that almost looked like a single tail was mistaken as type 6, though some similar wrong predictions are not highlighted in similar manner. In many other cases of wrong predictions (see the heatmaps in the appendix), it is unclear what the highlighted regions mean.

Fig. 8: One example from DeepLift with wrong prediction (from appendix fig. 16).

B. Results with mabCAM

Here we discuss the main results obtained from the 10-class experiments stored in checkpoint/project02 folder (to find it, follow the instructions in our updated github) while all other results from experiments with the smaller 3-class data will be briefly discussed in the appendix. Training lasted the entire 128 epochs without early termination, reaching validation accuracy
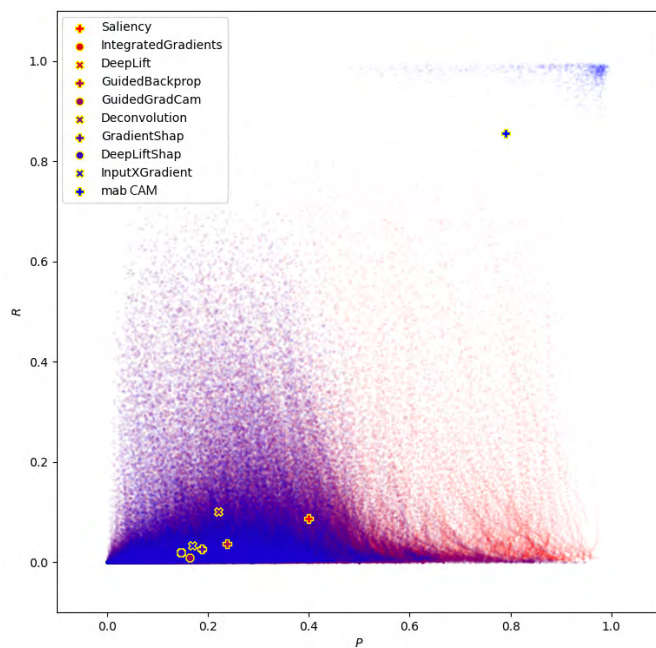

Fig. 9: Recall versus precision scores of five-band stratified heatmaps compared to ground-truth for mabCAM. Unlike previous figures, we scatter the scores of all individual test samples in the background. The mean values are presented as opaque coloured marker with yellow border highlight. *Zoomed-in view on an electronic media is highly recommended.*

of 92.19% at epoch 121 (starting at epoch 0). The test accuracy is 90.04%.

As expected, since mabCAM is finetuned against ground-truth heatmaps, it produces significantly better recall and precision compared to all other XAI methods, as shown in fig. 9(A). Clearly, our magnitude-banded heatmaps (mab) show much higher recall and precision scores with mean values beyond 0.8 for both, while other methods at most barely crossed 0.4 precision. Some samples of heatmaps generated by mabCAM are shown in fig. 10, demonstrating high quality (although imperfect) heatmaps. This is useful because it demonstrates that explanations can be tailored to the context, especially by matching them to ground-truth user-defined explanations. Naturally, users will have to define what is considered the suitable explanation beforehand, and a few-shot training on explanation data is a possible follow up for future works. This framework ensures that no ambiguity will arise, assuming that users agree on the pre-defined explanations.

Finally, we collected some samples of wrongly predicted images as shown in fig. 9(B). We see that mabCAM confuses class 0 and 6 often, which is not surprising, given that they differ by a single tail that sometimes blend into the noisy background colour or is too small. Other samples also demonstrate how background noise colours might be mistaken as tails e.g. second row of fig. 10(B) highlights the edge of the cell even though there is no tail. The heatmaps with wrong classifications generally show inconsistency, where features (patches with value 0.9, dark red) are mistakenly served as
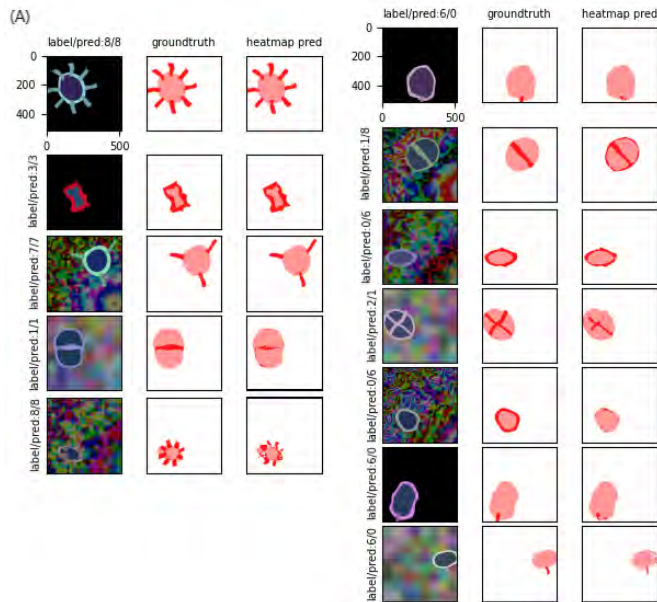
explanations.

Fig. 10: (A) From left to right: cell images, ground-truth and predicted magnitude-banded heatmaps. All labels are correctly predicted. (B) similar to (A), except all labels are wrongly predicted.

V. CONCLUSION

Recommendations and Caveats. Regardless of the imperfect performance, relative comparisons between the XAI methods can be made.

- Saliency method appears to highlight the relevant regions in the most conservative way, which is more suitable for localization in the case false positives are not important. In particular, AlexNet is scoring the highest recall.
- If only the edges of features are needed, then VGG and ResNet with input*grad, DeepLift, DeepLiftShap seem to be the reasonable choices. The same heatmap methods for Alexnet seem to produce heatmaps that go beyond capturing just the edges in rather inconsistent ways. Compared to Saliency, these methods may be more useful to detect small, hard to observe discriminative features, e.g. from medical images and other dense images.
- The heatmaps produced by ResNet appear to be sparsest, followed by VGG then AlexNet. Input size and depth of networks may be the reasons.
- A research into the role of negative values in the heatmaps may be necessary. If we continue with the interpretation that negative values correspond to negative contribution of prediction, some XAI methods such as DeepLift and DeepLiftShap may be completely incomprehensible.
- There are phenomena where large continuous patches or areas are ignored and where red and blue signals are mixed in close proximity. More investigations may be needed, e.g. on how to find the best *channel adjustment*.

In practice, it may not be easy to handle various noises produced by XAI methods. However, it is not impossible. Here we recommend that, in practical applications, every stakeholder should first agree on what constitutes an explanation. Our framework advocates the use of localization and distinguishing features as explanations. However, localization is sometimes sufficient as the explanation in computer vision problems, which is acceptable if agreed upon by all stakeholders. Otherwise, a major feature is to be selected as the main explanation.

We have provided an algorithm to produce synthetic data that we hope can be a baseline framework for testing XAI method, especially in the form of saliency maps or heatmaps. Some XAI methods appear to be more suitable for localization, while others are more responsive to the edges of features. In addition, mabCAM has been introduced as the framework compatible with the computation of multi-banded attribution heatmaps such as our synthetic data. The process is similar to semantic segmentation, a task which DNN has proven capable of performing.

Modifications required to boost the explainability power of existing XAI methods might differ across methods, making fair comparison a possibly difficult task. At least, for each application of XAI method, we should attempt to find a clear, consistent interpretation under the same context of study. For example, if negative values need to be treated with absolute values in some applications, at least an accompanying experiment is needed to show the effects and implications of performing such transformation. As of now, XAI still remains a challenging problem. However, it does exhibit good potential to improve the reliability of black-box models in the future.

Future directions. Synthetic data with clear-cut shapes is useful as a yardstick to measure the quality of XAI methods in an objective manner. Our synthetic dataset allows readers to clearly see the pros and cons of using certain XAI methods. We have also included rich features in our dataset enough to simulate the variety found in real life dataset, including background noises. We hope that the practice of including magnitude-banded heatmaps can be expanded in the future, and it may be useful to craft explanation ground-truth labels with magnitude banding on real datasets. Further studies on the performance of mabCAM on other datasets will help improve the objectivity of heatmap as explanation framework as a whole.

APPENDIX

The links to our appendix section is available in our github.

ACKNOWLEDGMENT

REFERENCES

[1] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2022.3328834

13

AI	Artificial Intelligence
AOPC	Area under Perturbation Curve
CAM	Class Activation Mapping
CHR	Computational vs human relevance
CNN	Convolutional Neural Network
DNN	Deep Neural Network
FC	Fully connected
FN	False Negative
FP	False Positive
GradCAM	Gradient CAM
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
IoU	Intersection over Union
LIME	Local Interpretable model-agnostic explanation
LRP	Layerwise Relevance Propagation
mab	magnitude-banded
MoRF	most relevant first
ROAR	RemOve and Retrain
ROC	Receiver OPerating Characteristics
SHAP	Shapley Additive exPlanation
SPA	Spatial Product Attention
TN	True Negative
TP	True Positive
XAI	eXplainable AI

Workshop at International Conference on Learning Representations, 2014.

[2] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7):1–46, 07 2015. doi: 10.1371/journal.pone.0130140. URL https://doi.org/10.1371/journal.pone.0130140.

[3] Jose Oramas, Kaili Wang, and Tinne Tuytelaars. Visual explanation by interpretation: Improving visual feedback capabilities of deep neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1ziPjC5Fm.

[4] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825, 2017. URL http://arxiv.org/abs/1706.03825.

[5] Xiao hui Li, Yuhan Shi, H. Li, Wei Bai, Y. Song, Caleb Chen Cao, and Li-Chiou Chen. Quantitative evaluations on saliency methods: An experimental study. *ArXiv*, abs/2012.15616, 2020.

[6] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K. Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660–2673, 2017.

[7] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, June 2016. doi: 10.1109/CVPR.2016.319.

[8] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016. URL http://arxiv.org/abs/1610.02391.

[9] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. volume 70 of *Proceedings of Machine Learning Research*, pages 3145–3153, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/shrikumar17a.html.

[10] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 9737–9748. Curran Associates, Inc., 2019.

[11] Thomas Hartley, Kirill Sidorov, Christopher Willis, and David Marshall. Swag: Superpixels weighted by average gradients for explanations of cnns. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 423–432, January 2021.

[12] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9505–9515. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8160-sanity-checks-for-saliency-maps.pdf.

[13] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319–3328. JMLR.org, 2017.

[14] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Sy21R9JAW.

[15] Leon Sixt, Maximilian Granz, and Tim Landgraf. When explanations lie: Why many modified bp attributions fail. In *ICML*, 2020.

[16] R. C. Fong and A. Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3449–3457, 2017. doi: 10.1109/ICCV.2017.371.

[17] S. A. Rebuffi, R. Fong, X. Ji, and A. Vedaldi. There and back again: Revisiting backpropagation saliency methods. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8836–8845, 2020. doi: 10.1109/CVPR42600.2020.00886.

[18] Xiao-Hui Li, Yuhan Shi, Haoyang Li, Wei Bai, Caleb Chen Cao, and Lei Chen. An experimental study of quantitative evaluations on saliency methods. New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383325. doi: 10.1145/3447548.3467148. URL https://doi.org/10.1145/3447548.3467148.

[19] Leila Arras, Ahmed Osman, and Wojciech Samek. Clevr-xai: A benchmark dataset for the ground truth evaluation of neural network explanations. *In-*

formation Fusion, 81:14–40, 2022. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2021.11.008. URL https://www.sciencedirect.com/science/article/pii/S1566253521002335.

[20] Marco Tulio Ribeiro, Sameer Singh, and Carlos 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*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

[21] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda B. Viégas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In Jennifer G. Dy and Andreas Krause, editors, *ICML*, volume 80 of *JMLR Workshop and Conference Proceedings*, pages 2673–2682. JMLR.org, 2018. URL http://dblp.uni-trier.de/db/conf/icml/icml2018.html.

[22] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, May 2019. ISSN 2522-5839. doi: 10.1038/s42256-019-0048-x. URL https://doi.org/10.1038/s42256-019-0048-x.

[23] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3319–3327, 2017.

[24] B. Zhou, D. Bau, A. Oliva, and A. Torralba. Interpreting deep visual representations via network dissection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9):2131–2145, 2019.

[25] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, page 201907375, Sep 2020. ISSN 1091-6490. doi: 10.1073/pnas.1907375117. URL http://dx.doi.org/10.1073/pnas.1907375117.

[26] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82 – 115, 2020. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2019.12.012. URL http://www.sciencedirect.com/science/article/pii/S1566253519308103.

[27] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89, 2018.

[28] Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence (xai): Toward medical xai. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–21, 2020. doi: 10.1109/TNNLS.2020.3027314.

[29] Christian Callegari, Pietro Ducange, Michela Fazzolari, and Massimo Vecchio. Explainable internet traffic classification. *Applied Sciences*, 11(10), 2021. ISSN 2076-3417. doi: 10.3390/app11104697. URL https://www.mdpi.com/2076-3417/11/10/4697.

[30] Alfredo Nascita, Antonio Montieri, Giuseppe Aceto, Domenico Ciuonzo, Valerio Persico, and Antonio Pescapé. Xai meets mobile traffic classification: Understanding and improving multimodal deep learning architectures. *IEEE Transactions on Network and Service Management*, 18(4):4225–4246, 2021. doi: 10.1109/TNSM.2021.3098157.

[31] Fabian Eitel and Kerstin Ritter. Testing the robustness of attribution methods for convolutional neural networks in mri-based alzheimer's disease classification. In Kenji Suzuki, Mauricio Reyes, Tanveer Syeda-Mahmood, Ben Glocker, Roland Wiest, Yaniv Gur, Hayit Greenspan, and Anant Madabhushi, editors, *Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support*, pages 3–11, Cham, 2019. Springer International Publishing. ISBN 978-3-030-33850-3.

[32] Peifei Zhu and Masahiro Ogino. Guideline-based additive explanation for computer-aided diagnosis of lung nodules. In Kenji Suzuki, Mauricio Reyes, Tanveer Syeda-Mahmood, Ben Glocker, Roland Wiest, Yaniv Gur, Hayit Greenspan, and Anant Madabhushi, editors, *Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support*, pages 39–47, Cham, 2019. Springer International Publishing. ISBN 978-3-030-33850-3.

[33] A. W. Thomas, H. R. Heekeren, K. R. Müller, and W. Samek. Analyzing Neuroimaging Data Through Recurrent Deep Learning Models. *Front Neurosci*, 13:1321, 2019.

[34] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn L. Ball, Katie S. Shpanskaya, Jayne Seekins, David A. Mong, Safwan S. Halabi, Jesse K. Sandberg, Ricky Jones, David B. Larson, Curtis P. Langlotz, Bhavik N. Patel, Matthew P. Lungren, and Andrew Y. Ng. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *CoRR*, abs/1901.07031, 2019. URL http://arxiv.org/abs/1901.07031.

[35] Guannan Zhao, Bo Zhou, Kaiwen Wang, Rui Jiang, and Min Xu. Respond-cam: Analyzing deep models for 3d imaging data by visualizations. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 485–492, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00928-1.

[36] Xiaoxiao Li, Nicha C. Dvornek, Juntang Zhuang, Pamela Ventola, and James S. Duncan. Brain biomarker inter-

pretation in asd using deep learning and fmri. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 206–214, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00931-1.

[37] Magdalini Paschali, Sailesh Conjeti, Fernando Navarro, and Nassir Navab. Generalizability vs. robustness: Investigating medical imaging networks using adversarial examples. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 493–501, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00928-1.

[38] Heather D. Couture, J. S. Marron, Charles M. Perou, Melissa A. Troester, and Marc Niethammer. Multiple instance learning for heterogeneous images: Training a cnn for histopathology. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 254–262, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00934-2.

[39] Ziqi Tang, Kangway V. Chuang, Charles DeCarli, Lee-Way Jin, Laurel Beckett, Michael J. Keiser, and Brittany N. Dugger. Interpretable classification of alzheimer's disease pathologies with a convolutional neural network pipeline. *Nature Communications*, 10 (1):2173, 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-10212-1. URL https://doi.org/10.1038/s41467-019-10212-1.

[40] Zachary Papanastasopoulos, Ravi K. Samala, Heang-Ping Chan, Lubomir Hadjiiski, Chintana Paramagul, Mark A. Helvie M.D., and Colleen H. Neal M.D. Explainable AI for medical imaging: deep-learning CNN ensemble for classification of estrogen receptor status from breast MRI. In Horst K. Hahn and Maciej A. Mazurowski, editors, *Medical Imaging 2020: Computer-Aided Diagnosis*, volume 11314, pages 228 – 235. International Society for Optics and Photonics, SPIE, 2020. doi: 10.1117/12.2549298. URL https://doi.org/10.1117/12.2549298.

[41] Yao Qin, Konstantinos Kamnitsas, Siddharth Ancha, Jay Nanavati, Garrison W. Cottrell, Antonio Criminisi, and Aditya V. Nori. Autofocus layer for semantic segmentation. *CoRR*, abs/1805.08403, 2018. URL http://arxiv.org/abs/1805.08403.

[42] Hyebin Lee, Seong Tae Kim, and Yong Man Ro. Generation of multimodal justification using visual word constraint model for explainable computer-aided diagnosis. In Kenji Suzuki, Mauricio Reyes, Tanveer Syeda-Mahmood, Ben Glocker, Roland Wiest, Yaniv Gur, Hayit Greenspan, and Anant Madabhushi, editors, *Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support*, pages 21–29, Cham, 2019. Springer International Publishing. ISBN 978-3-030-33850-3.

[43] Alex Krizhevsky. One weird trick for parallelizing convolutional neural networks. *CoRR*, abs/1404.5997, 2014. URL http://arxiv.org/abs/1404.5997.

[44] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[45] S. Liu and W. Deng. Very deep convolutional neural network based image classification using small training sample size. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 730–734, 2015.

[46] Avanti Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje. Not just a black box: Learning important features through propagating activation differences. *ArXiv*, abs/1605.01713, 2016.

[47] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net, 2014.

[48] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10590-1.

[49] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[50] Tutorial LRP. Lrp tutorial, accessed August 16, 2020. URL http://heatmapping.org/tutorial/.

[51] Ashwin Srinivasan and Ashwin Srinivasan. Note on the location of optimal classifiers in n-dimensional roc space. Technical report, 1999.

[52] César Ferri, José Hernández-Orallo, and Miguel Angel Salido. Volume under the roc surface for multi-class problems. In Nada Lavrač, Dragan Gamberger, Hendrik Blockeel, and Ljupčo Todorovski, editors, *Machine Learning: ECML 2003*, pages 108–120, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. ISBN 978-3-540-39857-8.

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[54] Erico Tjoa, Guo Heng, Lu Yuhao, and Cuntai Guan. Enhancing the extraction of interpretable information for ischemic stroke imaging from deep neural networks, 2019.