Focus! Rating XAI Methods and Finding Biases

1st Anna Arias-Duart
Barcelona Supercomputing Center (BSC)
Barcelona, Spain
anna.ariasduart@bsc.es

2nd Ferran Parés
Barcelona Supercomputing Center (BSC)
ferran.pares@bsc.es

3rd Dario Garcia-Gasulla
Barcelona Supercomputing Center (BSC)
dario.garcia@bsc.es

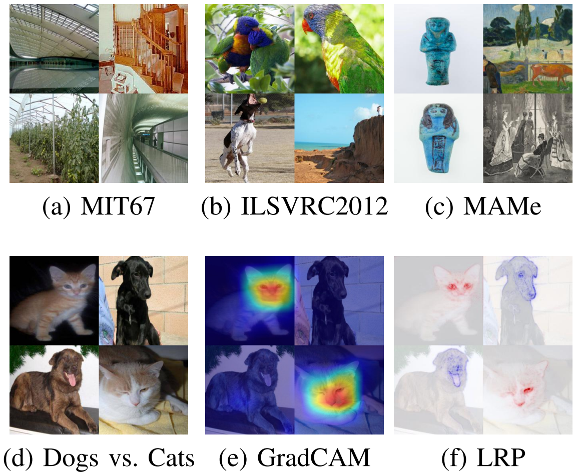
4th Victor Giménez-Ábalos
Barcelona Supercomputing Center (BSC)
victor.gimenez@bsc.es

Abstract*—AI explainability improves the transparency and trustworthiness of models. However, in the domain of images, where deep learning has succeeded the most, explainability is still poorly assessed. In the field of image recognition many feature attribution methods have been proposed with the purpose of explaining a model's behavior using visual cues. However, no metrics have been established so far to assess and select these methods objectively. In this paper we propose a consistent evaluation score for feature attribution methods—the *Focus*—designed to quantify their coherency to the task. While most previous work adds out-of-distribution noise to samples, we introduce a methodology to add noise from *within* the distribution. This is done through mosaics of instances from different classes, and the explanations these generate. On those, we compute a visual pseudo-precision metric, *Focus*. First, we show the robustness of the approach through a set of randomization experiments. Then we use *Focus* to compare six popular explainability techniques across several CNN architectures and classification datasets. Our results find some methods to be consistently reliable (LRP, GradCAM), while others produce class-agnostic explanations (SmoothGrad, IG). Finally we introduce another application of *Focus*, using it for the identification and characterization of biases found in models. This empowers bias-management tools, in another small step towards trustworthy AI.

I. Introduction

Explainability has become a major topic of research in Artificial Intelligence (AI), aimed at increasing trust in models such as Deep Learning (DL) networks. However, trustworthy models cannot be achieved with explainable AI (XAI) methods unless the XAI methods themselves can be trusted. This necessity gave rise to the assessment of XAI methods.

To evaluate XAI methods one may assess interpretability, a *qualitative* measure of how understandable an explanation is to humans [1]. While this is important to guarantee the proper interaction between humans and the model, interpretability generally involves end-users in the process [2], inducing strong biases. In fact, a qualitative evaluation alone cannot guarantee coherency to reality (*i.e.,* model behavior), as false explanations can be more interpretable than accurate ones. To enable trust on XAI methods, we also need *quantitative* and objective evaluation metrics, which validate the relation between the explanations produced by the XAI method and the behavior of the trained model under assessment. The challenge of quantitatively evaluating XAI methods lies in the absence of a ground truth: we cannot be sure of what a DL method is doing unless we understand the model parametrization itself (at which point we would not need a XAI method). Nonetheless, we still want to approximate the faithfulness [3] of XAI methods *w.r.t.* the underlying model, as this allows us to discern between accurate and misleading explanations. In this paper we propose a metric for that purpose, demonstrating its use on the evaluation of *feature attribution* methods when applied on image classification models.


(a) MIT67 (b) ILSVRC2012 (c) MAMe

(d) Dogs vs. Cats (e) GradCAM (f) LRP

Fig. 1. First row: sample of mosaics used by the evaluation methodology, obtained for: (a) MIT67 (b) ImageNet (ILSVRC2012) (c) MAMe. Second row: example of input mosaic from Dogs vs. Cats (d), and the explanations obtained by GradCAM (e) and LRP (f) for the target class *cat*.

The evaluation of XAI faithfulness is typically done by quantifying the change produced in the explanation when noise is added to the explained sample. This is necessary because no assumptions can be made regarding the *faithfulness* of samples without noise: explanations apparently inappropriate (*e.g.*, the background of a central object instead of object itself) may be an accurate portrait of the model's behavior, following a bias found and learnt from the data. The most popular approach to add noise is to visually alter samples [4], [5]. However, disturbed images become images outside of the original data distribution, which reduces the reliability of the analysis because of the effect it may cause on the activations of the model (*i.e.*, are bad explanations caused by a bad method or by the corruption inserted into the samples?).

In this paper we propose a novel evaluation score for *feature attribution* methods, described in §III. Our input alteration approach induces in-distribution noise into samples, that is, alterations on the input which correspond to visual patterns found within the original data distribution. To do so we modify the context of the sample instead of the content, leaving the original pixels values untouched. In practice, we create a new sample, composed of samples of different classes, which we call a *mosaic image* (see Figure 1). Using *mosaics* as input has a major benefit: each input quadrant is an image from the original distribution, producing blobs of activations in each quadrant which are consequently coherent. Only the pixels forming the borders between images, and the few corresponding activations, may be considered out of distribution.

By inducing in-distribution noise, *mosaic images* introduce a problem in which XAI methods may objectively err (*i.e.*, focus on something it should not be focusing on). On those composed mosaics we ask a XAI method to provide explanation for just one of the contained classes, and follow its response. In a sort of eye-tracking game, we measure how much of the explanation generated by the XAI is located on the areas corresponding to the target class, quantifying it through the *Focus* score. This score allows us to compare methods in terms of explanation precision, evaluating the capability of XAI methods to provide explanations related to the requested class (see §V for the comparison of six XAI methods). Using *mosaics* has another benefit. Since the noise introduced is in-distribution, the explanation errors identify and exemplify biases of the model. This facilitates the elimination of biases in models and datasets, potentially resulting in more reliable solutions. We illustrate how to do so in §VI.

II. Related Work

The evaluation of XAI faithfulness in current literature can be divided into two broad groups: qualitative and quantitative methods. On one side, qualitative evaluations are based on assumptions induced by the human understanding of perception [3], [6], hence an evaluation built on top of human cognitive biases, may not be necessarily aligned with the learning paradigm of DL. In contrast, quantitative evaluations avoid such human biases by excluding the human from the XAI assessment process. Quantitative evaluations have a primary barrier to overcome: the lack of a ground truth specifying what defines a correct explanation. Instead, these methods introduce noise to evaluate the output, assuming certain properties on their expected response. Such noise can be introduced both on data and model parametrization. We can separate noise inducing evaluation methods based on their generated response. While some produce categorical evaluation, others generate numerical ones. Works proposing categorical evaluations define axioms or tests that XAI methods must satisfy or fulfill. One of these works [7] discusses three axioms for XAI methods: *Sensitivity*, *Implementation invariance* and *Completeness*. *Sensitivity* checks that irrelevant features have no explanation attributed, *Implementation invariance* checks that functionally similar models produce equivalent attributions, and *Completeness* is satisfied when the difference between the sum of the attributions of an input and a baseline is equal to the change of the output. In [8], two types of tests are proposed: the *model parameter randomization* test and the *data randomization* test. The first aims to prove that if an explanation depends on the model, a randomized model should produce a different explanation. The second test checks if there exists a relation between the explanation and the labels, that is, if a regular dataset produces different results than one with randomly permuted labels.

While categorical evaluations can validate whether the XAI methods fulfill or not certain properties, they are limited in terms of comparison and ranking purposes. Numeric evaluations are more informative to that end, providing comparable scores that can be easily ranked. Examples of numeric evaluations are the Pixel Flipping algorithm [4] and the Average Drop % metric [5]. The first performs a semi-quantitative analysis by perturbing pixels from patches with the greatest relevance, and then assessing the impact on the prediction score. The second measures the drop percentage of the score when only the part of the image with attribution is shown *w.r.t.* the score with the full image. These numeric evaluations rely on disturbing input images. As said before, disturbed images fall out of the original distribution, reducing the reliability of the following analysis due to its effect on produced model activations.

A few XAI assessment methods use images from the original distribution without any perturbation on them. Since these methods lack a source of noise, they require of an assumption to numerically evaluate the produced explanations. Examples of these methods are those working with manually generated ground truth regions, assuming that the relevance produced by XAI methods should fall inside regions corresponding to a target class. In the work of Zhang *et al.* [9], authors introduce the *pointing game* technique to evaluate if the point of maximum relevance lies on the object of the target class. Similarly, Selvaraju *et al.* evaluate the localization capacity of methods by drawing bounding boxes from the explanation heatmap and calculating the error *w.r.t.* the correct bounding box [3]. However, as pointed out by different works [7], [10], the premise that objects, or any other part of the input, are the only relevant feature for the prediction cannot be presumed.

In this context, we define the *Focus* score. A quantitative, numerical, in-distribution noise inducing method to assess XAI methods and AI models (see §V and §VI). It is based on compositions of images from the dataset as a source of noise. Since each quadrant of the mosaic contains an undisturbed image, the network activations in each quadrant will fall inside the original distribution. Additionally, the *Focus* score does not expect relevance to be only centered on a pre-defined region, avoiding assumptions regarding the localization of explanation within each quadrant.

III. Methodology

In this section we define a metric—the *Focus*—intended to assess the explanations produced by *feature attribution* methods. This score involves three elements: an explainability method (§III-A), a trained model (§III-B), and a set of mosaic samples (§III-C). In the following subsections we discuss these in detail, before defining the *Focus* itself (§III-D).

A. Explainability methods

Throughout the paper we use and evaluate six *feature attribution* methods:

- Gradient-weighted Class Activation Mapping (Grad-CAM) [3], based on the implementation of Gildenblat *et al.*[1]. We compute the gradients of the logits of the class *w.r.t.* the feature maps of the final convolutional layer. That is, the 5th layer for AlexNet, the 13th for VGG16 and the last layer from the 5th block for ResNet-18 (also known as block E).
- Layer-wise Relevance Propagation (LRP) [4], based on the implementation of Nam *et al.* [11]. On the first layer we use the z^B-rule [12], on fully connected layers the LRP-ϵ [4], and on convolutional layers the LRP-$\alpha\beta$ [4] with $\alpha = 1$ and $\beta = 0$.
- SmoothGrad [13], based on the implementation of Nakashima *et al.* [2]. Explanations are obtained computing the gradient of the specific class score *w.r.t.* the input pixels and adding small perturbations on the input image (in our case Gaussian Noise).
- LIME [6], based on the implementation of Tulio *et al.*[3]. Each explanation is computed considering 1,000 samples and the final explanation only includes the five top features, that is, the five most relevant superpixels.
- GradCAM++ [5], based on the implementation of Gildenblat *et al.*[1]. We use the last convolutional layer to compute the GradCAM++ explanations.
- Integrated Gradients (IG) [7], based on the implementation of Kokhlikyan *et al.* [14]. We use the black image as the baseline image and 30 steps to approximate the integral.

For all *feature attribution* methods we skip their custom post-processing for visualization purposes.

B. Models

To run a XAI method we need a model to explain, generated from an architecture, trained on a dataset, with a training configuration. In our experiments, we use the following:

- **Architectures:** AlexNet [15], VGG16 [16] and ResNet-18 [17].
- **Datasets:** the Dogs vs. Cats[4], the Museum Artworks Medium dataset (MAMe) [18], the MIT67 [19] and the ILSVRC 2012 [20] (hereafter ImageNet).

Training configurations: During training, AMSGrad [21] is used for optimizing weights and data augmentation is performed. Code needed to replicate trainings and experiments of this paper can be found in [5]. For the ImageNet dataset, we use the pre-trained models in the subpackage *torchvision.models*[6,7,8]. For Dogs vs. Cats and MAMe datasets, we take the ImageNet pre-trained models and perform training on top of them. Finally, in the case of the MIT67 dataset, we train the model on top of pre-trained Places365-Standard dataset [22] (models available in the official repository [9]).

C. Mosaic construction

The last element required to compute the *Focus* metric is the mosaic, an image composed by four different samples disposed in a two by two grid. Samples from the training set are never used for mosaics. To formalize mosaics, and later *Focus*, let us define a dataset $\mathbb{D}$ composed by a set of images $\mathbb{I} = \{img_1, img_2, ..., img_N\}$ and a set of classes $\mathbb{C} = \{c_1, c_2, ..., c_K\}$, where N is the number of total images and K is the number of total classes. Every image in $\mathbb{I}$ has assigned a unique class from $\mathbb{C}$: $c(img)$. From here we build a set of mosaics $\mathbb{M} = \{m_1, m_2, ..., m_J\}$ where J is the total number of mosaics in $\mathbb{M}$. A mosaic m is composed by four images $m = \{img_1, img_2, img_3, img_4\}$ and characterized by a target class $tc = c(m)$, the specific class the XAI method is expected to explain. While two images of the mosaic belong to the target class $c(img_1) = c(img_2) = c(m)$, the other two are randomly chosen among the rest of classes $c(img_3) \neq c(m); c(img_4) \neq c(m)$. Mosaics are implemented as two by two, non-overlapping grid, with the position of each image being random. Samples of mosaics from different datasets can be seen in Figure 1.

For the sake of keeping the same resolution of the visual patterns seen by models during their training, and thus keeping most of the noise added within the training distribution, all XAI evaluation experiments use 448×448 mosaics. That is, four times the size of the inputs the models were trained with. AlexNet and VGG16 architectures were not input-agnostic when originally proposed, being limited by design to an input size of 224×224 pixels. Nowadays, these architectures employ an Adaptive Pooling Layer to circumvent this problem.

D. The Focus metric

Before starting with the *Focus*, let us introduce its foundations as well as its motivation. When a *feature attribution* method is applied to an image to explain the model's prediction regarding a chosen class, it typically produces a map from pixels to real values, referred to as relevance. While some *feature attribution* methods also provide negative relevance, this is not generalized. For the scope of this paper we *focus* on positive relevance only. For XAI methods providing both

[1] https://github.com/jacobgil/pytorch-grad-cam

[2] https://github.com/kazuto1011/grad-cam-pytorch

[3] https://github.com/marcotcr/lime

[4] https://www.kaggle.com/c/dogs-vs-cats/overview

[5] https://github.com/HPAI-BSC/Focus-Metric

[6] https://download.pytorch.org/models/alexnet-owt-4df8aa71.pth

[7] https://download.pytorch.org/models/vgg16-397923af.pth

[8] https://download.pytorch.org/models/resnet18-5c106cde.pth

[9] https://github.com/CSAILVision/places365

positive and negative relevance, only the positive relevance is used, while negative values are treated as 0.

Intuitively, the output of a method is reliable (not necessarily understandable) when higher values of relevance lie on pixels of the image that are evidence of the chosen class. We consider *visual evidence* any set of pixels used by the model to distinguish the chosen class from other classes of the task. Formally, we introduce a probability distribution $\mathcal{P}_{tc}$ over all pixels given a target class tc. The probability of sampling a pixel from $\mathcal{P}_{tc}$ is proportional to the pixel's relevance toward tc attributed by an explainability method $\mathcal{A}$ and a model θ. Then, we define the formal reliability $Re(\mathcal{A}, \theta, tc)$ as the probability that a pixel sampled from the distribution $\mathcal{P}_{tc}$ lies within visual evidence corresponding to tc.

The definition of $Re(\mathcal{A}, \theta, tc)$ over a method-model-class triplet can be extended to evaluate a method-model pair as $Re(\mathcal{A}, \theta)$. To do so, we take the expectancy of reliability over all classes $\mathbb{C}$: $Re(\mathcal{A}, \theta) = \mathbb{E}_{tc \in \mathbb{C}}[Re(\mathcal{A}, \theta, tc)]$. More accurate models and better *feature attribution* methods will result in $Re(\mathcal{A}, \theta)$ values closer to 1. The lower bound of $Re(\mathcal{A}, \theta)$ is the probability that any pixel lies within evidence, which is proportional to the number of pixels lying on visual evidence.

In order to obtain the $Re(\mathcal{A}, \theta)$ metric, we would require a ground truth of which pixels are evidence toward a class. A way to bypass this limitation is to take the assumption that evidence toward a class is more prevalent in images labelled with that class, this being the main assumption of the proposed approach. We thus define the *Focus* as an estimator of the reliability computed over a dataset. The *Focus* evaluates the expected probability that a pixel sampled from $\mathcal{P}_{tc}$ lies on an image of tc. Notice the *Focus* underestimates the reliability, as evidence toward a class can be present on samples of a different class of the dataset. We leverage this to our advantage in §VI, using it to detect biases in models and dataset (be it desirable or undesirable biases).

Since this new score only requires image labelling instead of pixel labelling, we transform the dataset into a set of mosaics as introduced in §III-C. The number of samples composing each mosaic could be altered. In this case we use four, as it provides a good balance between robustness (small mosaics are more noisy) and complexity (large mosaics are more computationally expensive). Therefore, we compute *Focus* on subsets of four images (*i.e.*, each image composing the mosaic is labeled) to estimate the *Focus* of a method and a model on the whole dataset. In this context, the *Focus* metric estimates the reliability of XAI method's output as the probability of the sampled pixels lying on an image of the target class of the mosaic $c(m)$. This is equivalent to the proportion of positive relevance lying on those images:

$$F_{\mathcal{A},\theta}(m) = \frac{R_{c(m)}(img_1) + R_{c(m)}(img_2)}{R_{c(m)}(m)} \tag{1}$$

where $R_c(r)$ is the sum of positive relevance toward class c on the region of the mosaic r.

This probability can be interpreted as a precision of the relevance. In an sort of eye-tracking game, the *Focus* metric asks to the XAI method "*Why does mosaic m belong to class $c(m)$?*" on a mosaic m which contains both samples belonging and not belonging to the target class $c(m)$. Given the previous question and a good underlying model, a reliable *feature attribution* method should be able to concentrate most of its explanation relevance on the two appropriated images of the mosaic (*i.e.*, img_1 and img_2).

As explainability becomes more reliable, the *Focus* will grow. As with reliability, the theoretical upper-bound of the *Focus* score is 1, but this is unrealistic: *visual evidence* of a class appearing exclusively on images of that class is seldom true. On the other hand, in the case of uninformed relevance attribution (*i.e.*, unreliable explanations), the expected value of *Focus* is 0.5, since the probability of picking a pixel of the correct class is just the prior probability of picking one of the pixels of img_1 or img_2, which amount to half of the total pixels in the mosaic.

IV. Randomization test

Current evaluations of XAI methods frequently rely on qualitative assessments. These include humans in the loop, thus introducing a significant subjective bias. This is further complicated by the fact that XAI methods are typically designed to focus on prominent, central and/or high contrast areas on the input. When this happens, a XAI method may become more dependant on the input sample than on the underlying model supposedly generating the explanations. To verify this is not an issue for the *Focus* score, we run a set of randomization tests.

First, we conduct a randomization experiment to assess and decide the exact position of the target class (tc) images within the two by two grid of the mosaic. This experiment uses GradCAM on top of a VGG16 model trained for the Dogs vs. Cats dataset (pre-trained on ImageNet). The six possible configurations of the two by two grid were tested, plus a seventh for random positioning. For each configuration 2,812 mosaics were created, using *cat* class as tc. The resulting *Focus* distributions are shown in Figure 2. Clearly, the positioning of target samples has an effect on the *Focus* distribution. Configurations where the two target class images (img_1 and img_2) are arranged contiguously tend to be better. While this may be partially the result of explanation relevance spilling over samples, it happens more prominently when correct samples are placed on top. Meanwhile, the left-right configurations show a smaller gain when placing the correct samples on the right. We hypothesize that such variance in *Focus* performance is independent from the underlying XAI method, and is instead caused by particularities of the dataset and/or task. Since we cannot guarantee that these properties will hold among target classes, datasets or models, we decide to use a sampling approach hereafter. That is, the exact position samples within the composed grid is chosen randomly for every mosaic.

The second randomization test aims at evaluating the effect of model randomization on the *Focus* score. For that, we start using two different models. A VGG16 pre-trained on

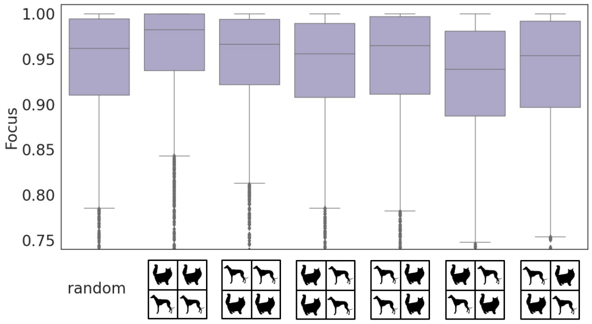


Fig. 2. *Focus* obtained by GradCAM on a VGG16 trained for Dogs vs. Cats dataset (pre-trained on ImageNet), using different mosaic configurations. Each box plot shows the distribution of *Focus* obtained from evaluating 2,812 samples for each configuration (the cat being the target class).

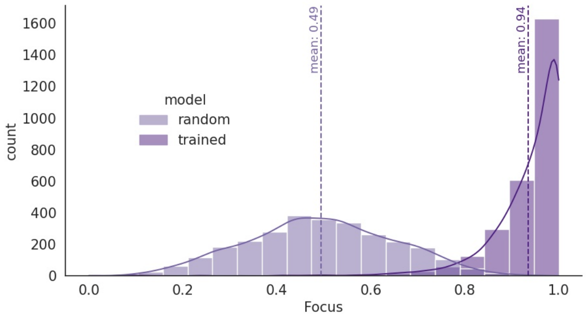


Fig. 3. Histogram of *Focus* scores obtained by GradCAM from 2,812 mosaics, using a VGG16 trained on Dogs vs. Cats and a randomized VGG16 model. The corresponding PDF estimation is represented by a contour line on top.

ImageNet and then trained for Dogs vs. Cats, and a totally randomized VGG16 model. The experiment computes the *Focus* metric on the cat target class ($tc = cat$) for the 2,812 mosaics with random layout. The distribution of *Focus* achieved by GradCAM on both models is shown as histograms in Figure 3. While the mean of the *Focus* obtained with the pre-trained model reach a remarkable 0.94, the random model mean score is 0.49, that is roughly 50% of the relevance lays on the wrong class quadrants. To take the randomization analysis further, we replicate the experiment of Adebayo *et al.* [8]. In it, the authors qualitatively pointed at how visual explanations can be compelling to the eye even when randomizing one or more layers of the underlying model. In this experiment, layers are randomized in cascade, starting with only the top layer, and increasingly randomizing more layers one by one until obtaining a fully randomized model. We use GradCAM on InceptionV3 [23] (like [8]) adding as well VGG16 and ResNet-18. Our results are straight-forward: simply randomizing the top layer (or any other set of layers) makes the *Focus* drop to a 50% mean, the same score obtained by a purely random XAI method. This illustrates how resistant the *Focus* score is to misleading explanations.

V. Evaluation of XAI methods

Let us now put *Focus* into practice. We evaluate GradCAM, LRP, SmoothGrad, LIME, GradCAM++ and IG, using three architectures (AlexNet, VGG16 and ResNet-18) and four target datasets (Dogs vs. Cats, MAMe, MIT67 and ImageNet). For the Dogs vs. Cats dataset, the MAMe dataset and the MIT67 dataset we use 100 mosaics per target class, a total of 200, 2,900 and 6,700 mosaics respectively. In the ImageNet experiments a total of 10,000 mosaics are used (10 per target class). Since the LIME method is computationally expensive, the experiments with this method have been restricted to the Dogs vs. Cats (200 mosaics) and MAMe datasets (2,900 mosaics). For each experiment, Table I depicts the mean and the standard deviation of the *Focus* distribution. For further insides, Figure 4 shows these distributions as box plots. Overall, *Focus* seems to be correlated with model accuracy. As models get better, the mean *Focus* goes up and the standard deviation goes down. However, there are exceptions to this rule, as the ResNet-18 outperforms the *Focus* of others consistently. This indicates that certain architectures produce more precise explanations than others.

GradCAM results are the best in average. Reaching a mean *Focus* above 81% in all experiments but one, it is best in 2/3 of the experiments conducted. This XAI method is particularly robust to noisy models, performing competitively even with 36% accuracy models (AlexNet on ImageNet). GradCAM++ scores significantly lower in every experiment we conducted, being the 3rd or 4th in the overall ranking. Still, its explanations are well above random behavior.

LRP gets the second best *Focus* in 8 of 12 experiments, and wins in 3 of the remaining 4. As LIME, performs very well on the high accuracy models of Dogs vs. Cats, outperforming GradCAM. But on the other models it is able to beat the mean of GradCAM only once, while variance grows significantly. The worst results of LRP are produced in the MIT67 experiment, for the AlexNet and VGG16 models. Notice these models where pre-trained on the Places365-Standard dataset [22], which is noticeably narrower than ImageNet (434 vs 1,000 classes). According to these results, LRP is a very good methodology for XAI, when applied to very accurate models.

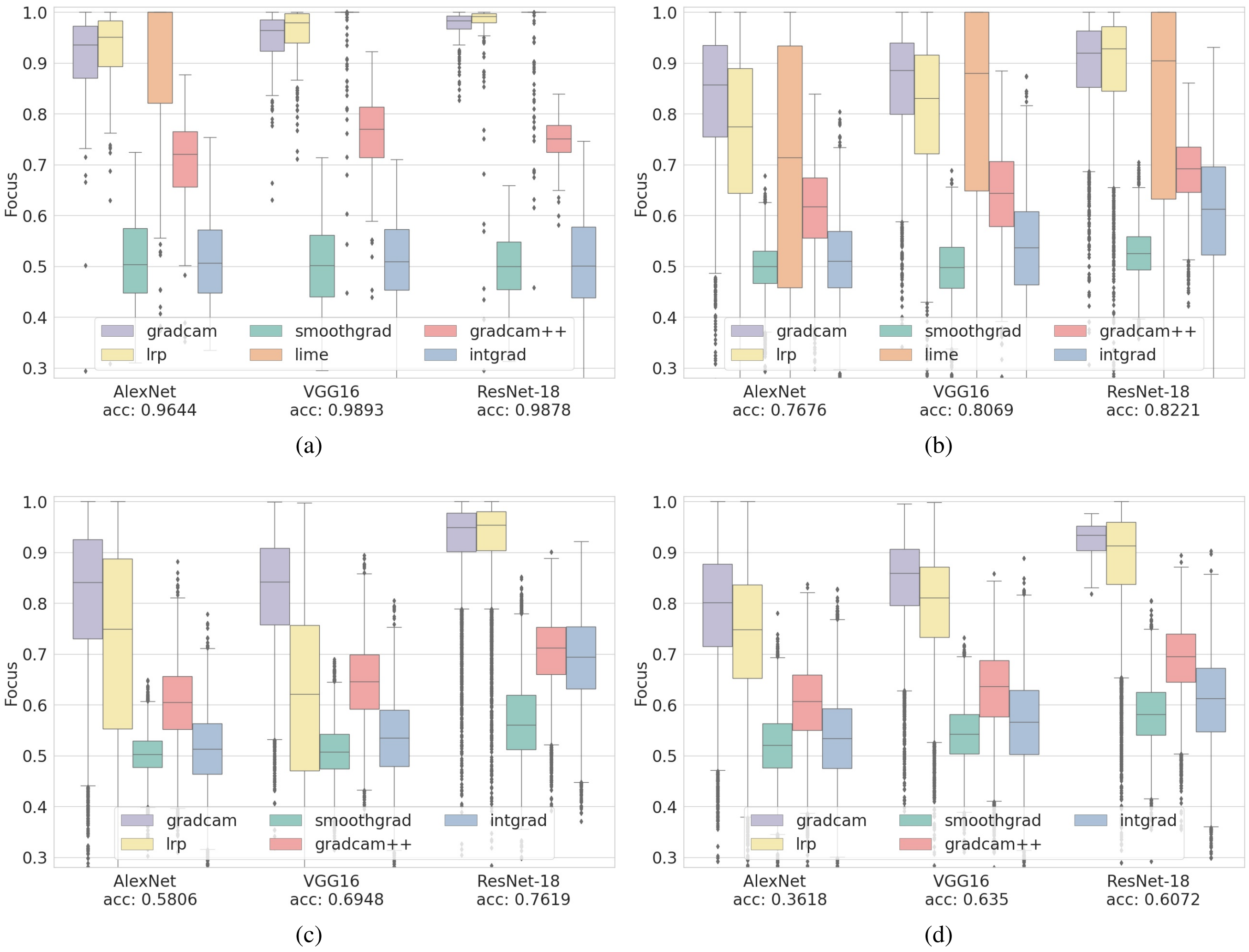
LIME performs remarkably well for the Dogs vs. Cats models, the ones with the highest accuracy (pre-trained with ImageNet), and the only two-class classification task. For lower accuracy models (AlexNet in this task, and all in MAMe task), LIME becomes less reliable. Its mean *Focus* drops, and its standard deviation becomes the largest of all XAI methods. The lack of hyperparameter tuning (which is impractical) may have penalized the results for MAMe.

SmoothGrad generally obtains a *Focus* around 50%, showing close to random precision in all experiments. Since this method uses the gradient of the output *w.r.t.* to the input pixels, misleading attribution scores could be caused by discontinuous gradients or by saturation of gradients, as previously suggested [24]. The IG method tries to overcome these drawbacks and, while its mean score is always better than the SmoothGrad, it remains quasi random in general. The cause behind these noisy explanations may be the domination of gradients in saturated areas, as shown by Miglani *et al.* [25].

TABLE I

MEAN AND STANDARD DEVIATION (IN PARENTHESIS) OF THE FOCUS DISTRIBUTION OBTAINED BY DIFFERENT XAI METHODS (COLUMNS) ON ARCHITECTURES TRAINED FOR DIFFERENT DATASETS (ROWS). THE ACCURACY SHOWN BESIDES EACH MODEL (*acc*) CORRESPONDS TO THE MEAN PER CLASS ACCURACY ON THE VALIDATION SET. BEST MEAN FOCUS PER ROW IN BOLD.

		GradCAM	LRP	SmoothGrad	GradCAM++	IntGrad	LIME
Dogs vs. Cats	AlexNet - acc: 0.9644	0.9101 (± 0.0903)	**0.9230 (± 0.1018)**	0.5092 (± 0.0840)	0.7041 (± 0.0872)	0.5113 (± 0.0858)	0.8883 (± 0.1797)
	VGG16 - acc: 0.9893	0.9446 (± 0.0577)	0.9526 (± 0.0877)	0.5035 (± 0.0854)	0.7574 (± 0.0777)	0.5108 (± 0.0849)	**0.9724 (± 0.1024)**
	ResNet-18 acc: 0.9878	0.9725 (± 0.0320)	**0.9741 (± 0.1018)**	0.4970 (± 0.0677)	0.7484 (± 0.0456)	0.5037 (± 0.0976)	0.9735 (± 0.0809)
MAMe	AlexNet - acc: 0.7676	**0.8292 (± 0.1346)**	0.7237 (± 0.2359)	0.4962 (± 0.0515)	0.6117 (± 0.0879)	0.5138 (± 0.0825)	0.6695 (± 0.2819)
	VGG16 - acc: 0.8069	**0.8556 (± 0.1123)**	0.7827 (± 0.2015)	0.4957 (± 0.0626)	0.6401 (± 0.0932)	0.5354 (± 0.1050)	0.7951 (± 0.2459)
	ResNet-18 - acc: 0.8220	**0.8941 (± 0.0938)**	0.8864 (± 0.1268)	0.5257 (± 0.0521)	0.6874 (± 0.0665)	0.6076 (± 0.1213)	0.7937 (± 0.2533)
MIT67	AlexNet - acc: 0.5806	**0.8133 (± 0.1401)**	0.6864 (± 0.2545)	0.5017 (± 0.0415)	0.6037 (± 0.0773)	0.5121 (± 0.0736)	—
	VGG16 - acc: 0.6948	**0.8230 (± 0.1088)**	0.6033 (± 0.1978)	0.5079 (± 0.0522)	0.6441 (± 0.0776)	0.5340 (± 0.0809)	—
	ResNet-18 - acc: 0.7619	**0.9248 (± 0.0818)**	0.9162 (± 0.1265)	0.5682 (± 0.0807)	0.7027 (± 0.0702)	0.6892 (± 0.0865)	—
ImageNet	AlexNet - acc: 0.3618	**0.7866 (± 0.1179)**	0.7345 (± 0.1442)	0.5194 (± 0.0644)	0.6018 (± 0.0797)	0.5342 (±0.0867)	—
	VGG16 - acc: 0.6350	**0.8426 (± 0.0881)**	0.7914 (± 0.1140)	0.5425 (± 0.0566)	0.6279 (± 0.0814)	0.5637 (± 0.0924)	—
	ResNet-18 - acc: 0.6072	0.8792 (± 0.0849)	**0.8814 (± 0.1068)**	0.5827 (± 0.0608)	0.6885 (± 0.0711)	0.6081 (± 0.0897)	—



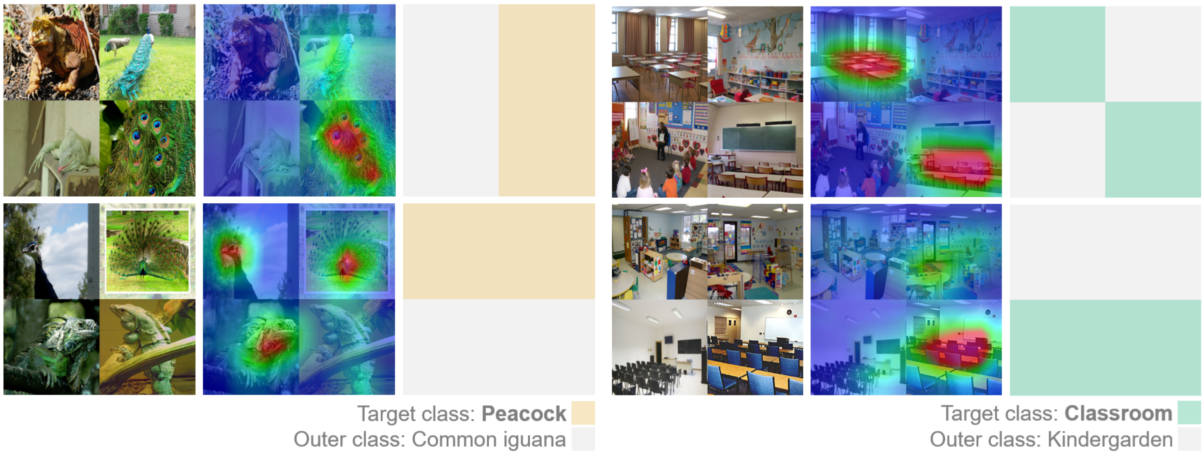
(a)

(b)

(c)

(d)

Fig. 4. *Focus* distribution boxplot for different XAI methods applied to models trained for different datasets. The accuracy (*acc*) shown under each model corresponds to the mean per class accuracy on the validation set of the corresponding dataset. These datasets are: (a) Dogs vs. Cats dataset, (b) MAMe dataset, (c) MIT67 dataset and (d) ImageNet dataset. LIME is only present in (a) and (b).


Target class: **Peacock**
Outer class: Common iguana
Target class: **Classroom**
Outer class: Kindergarden

(a)

(b)

Fig. 5. GradCAM explanations obtained on the ResNet-18 trained with (a) ImageNet and (b) MIT67. Two examples of mosaics are shown in the first column. The second column shows the corresponding GradCAM explanations for the target class. The third column specifies the positions of the classes within the mosaic.(a) The target class is the *Peacock* class and the outer class is the *Common iguana* class. The example above obtains a high *Focus* (0.818) and the one below a lower one (0.494).(b) The target class is the *Classroom* class and the outer class the *Kindergarden*. The *Focus* scores are 0.847 and 0.596 respectively.

VI. Bias detection

Explainability has been used to validate biases in models before. For example, the GradCAM authors use their method to visually validate the existence of gender bias in a model [3]. However, this approach typically relies on a human identifying the bias beforehand. With *Focus* we can go beyond, automating the bias identification process as well, while providing visual validation to the user. This is possible because mosaics induce in-distribution noise, where *Focus* errors directly correspond to visual biases of the model.

In this section we illustrate how mosaics and *Focus* together can be used to identify sources of bias in a model. The proposed procedure is as follows. First, for a better detection of biases between pairs of classes, we use mosaics with two classes. Therefore, in the mosaics used for this section, samples different from the target class actually belong to the same class: $c(img_3) = c(img_4) \neq c(m)$. We concentrate on the most relevant biases by finding the pairs of classes obtaining the lowest mean *Focus* in their joint mosaics. For each of these pairs we extract the mosaics with highest and lowest *Focus*, and present them to a human evaluator who must review the explanations produced. The role of the evaluator is to interpret the rationale behind the explanations (both correct and incorrect) and its degree of generalization for the task. Based on that assessment, corrective measures can be implemented, as later discussed.

For this experiment we use the GradCAM method and the ResNet-18 architecture, a particularly robust configuration in our experiments. A few samples are shown in Figure 5, top ones corresponding to high *Focus* and the bottom ones to low *Focus*. For the example from the ImageNet dataset (see Figure 5 (a)), the model is able to correctly attribute relevance to the *Peacock* images on the upper mosaic, while, for the bottom mosaic, some of the relevance incorrectly fall on the head of the *Common iguana*. The fact that most of the incorrect relevance in the *Common iguana* falls in the subtympanic shield (*i.e.*, the characteristic circle in its jowl) seems to be related with its visual similarity with the ocellus of the *Peacock* (*i.e.*, the circular spot in the feathers). Notice the iguana's subtympanic shield is hardly visible in the top mosaic. For the example from the MIT67 dataset (see Figure 5 (b)), the model correctly attributes the relevance to the two target class images on the top mosaic, both belonging to the *Classroom* class. For the lower mosaic, the model struggles to find the evidence in the *Classroom* image when no tables are present. These patterns are consistent found in several mosaics for the classes studied. After reviewing several cases as the ones described above, one can identify at least two types of biases responsible for decreasing the *Focus* score. These are:

1) Shared bias: A visual evidence of the target class is found in an outer class image (*e.g.*, the ocellus shape found in the *Common iguana* class).
2) Missing bias: A visual evidence of the target class is not found in an image of the same target class (*e.g.*, the tables in the *Classroom* class).

After the identification of biases, and an assessment of their impact, one could try to mitigate their relevance for the model. For casuistry (1), shared bias, more images of the target class without the characteristic pattern found in the outer class could be added to the training set (*e.g.*, *Peacocks* images where the ocellus is not visible). Similarly, more images of the outer class where the characteristic pattern is present (*e.g.*, *Common iguana* images where the subtympanic shield is visible) could be added. In either case, the dependency of the target class *w.r.t.* the shared bias would be reduced, increasing the robustness of the model. For casuistry (2), missing bias, more samples without the identified visual patterns of the target class could be added to the training set (*e.g.*, *Classrooms* samples without tables). Again, this would reduce the dependency of the target class *w.r.t.* the missing bias.

VII. Conclusion

For the quantitative evaluation of XAI methods, we introduce a novel metric—the *Focus*—to assess the consistency of the method under the existence of in-distribution noise. First, we show the methodology to be consistent and resilient to misleading explanations. When applied to SmoothGrad or IG, *Focus* finds quasi-random explanations *w.r.t.* the model. In contrast, LRP and GradCAM are both found consistently reliable. GradCAM performs well on all experiments conducted, even when the underlying model is not particularly well fit to the task. LRP performs very well for high performing models, but it becomes more unreliable on less accurate models. This also seems to be the case of LIME, which suffers from an even larger variance. Furthermore, LIME computational complexity and need for hyperparameter tuning limits its practical application. GradCAM++ performs better than random, but not as well as GradCAM and LRP. Remarkably, the *Focus* results are rather consistent across tasks and architectures, providing strong empirical evidence of their performance.

The consistency of *Focus* is likely related with the type of noise it induces. By altering the context and not the content of samples, *Focus* adds and exploits in-distribution noise. Unlike out-distribution noise, this is less prone to arbitrary model behavior. Through in-distribution noise mosaics and the *Focus* score visually characterize bias in the model, and can be directly used as an automated bias identification and exemplification tool. This opens the door to use mosaics and *Focus* to improve models, datasets and explanations.

Focus is related with the precision metric (*i.e.*, $\frac{TP}{TP+FP}$). While *Focus* is not precision (it lacks the ground truth needed to specify TP from FP), it approximates it by implicit labeling of mosaic quadrants. That is, that all positive lays somewhere within the target quadrants, and all negative somewhere within the other two quadrants. A similar assumption could be made to define an analogous recall metric (*i.e.*, $\frac{TP}{TP+FN}$) using, for example, the negative relevance provided by some *feature attribution* methods. This remains as future work.

Acknowledgments

This work is supported by the European Union – H2020 Program under the “INFRAIA-01-2018-2019 – Integrating Activities for Advanced Communities”, Grant Agreement n.871042, “SoBigData++: European Integrated Infrastructure for Social Mining and Big Data Analytics” and by the Dept. de Recerca i Universitats of the Generalitat de Catalunya under the Industrial Doctorate Grant DI 2018-100. We thank Marta González Mallo for her insightful comments and suggestions.

References

[1] 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)*. IEEE, 2018, pp. 80–89.

[2] S. Mohseni, N. Zarei, and E. D. Ragan, “A multidisciplinary survey and framework for design and evaluation of explainable ai systems,” *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 11, no. 3-4, pp. 1–45, 2021.

[3] 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,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[4] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, “On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation,” *PloS one*, vol. 10, no. 7, 2015.

[5] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, “Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks,” in *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2018, pp. 839–847.

[6] M. T. Ribeiro, S. Singh, and C. Guestrin, “” why should i trust you?” explaining the predictions of any classifier,” in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[7] M. Sundararajan, A. Taly, and Q. Yan, “Axiomatic attribution for deep networks,” in *International Conference on Machine Learning*. PMLR, 2017, pp. 3319–3328.

[8] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, “Sanity checks for saliency maps,” *Advances in neural information processing systems*, vol. 31, 2018.

[9] J. Zhang, S. A. Bargal, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff, “Top-down neural attention by excitation backprop,” *International Journal of Computer Vision*, vol. 126, no. 10, pp. 1084–1102, 2018.

[10] W. Samek and K.-R. Müller, “Towards explainable artificial intelligence,” in *Explainable AI: interpreting, explaining and visualizing deep learning*. Springer, 2019, pp. 5–22.

[11] W.-J. Nam, S. Gur, J. Choi, L. Wolf, and S.-W. Lee, “Relative attributing propagation: Interpreting the comparative contributions of individual units in deep neural networks,” 2019.

[12] G. Montavon, S. Lapuschkin, A. Binder, W. Samek, and K.-R. Müller, “Explaining nonlinear classification decisions with deep taylor decomposition,” *Pattern Recognition*, vol. 65, pp. 211–222, 2017.

[13] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, “Smoothgrad: removing noise by adding noise,” *arXiv preprint arXiv:1706.03825*, 2017.

[14] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, and O. Reblitz-Richardson, “Captum: A unified and generic model interpretability library for pytorch,” 2020.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[16] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[17] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[18] F. Parés, A. Arias-Duart, D. Garcia-Gasulla, G. Campo-Francés, N. Viladrich, E. Ayguadé, and J. Labarta, “The mame dataset: on the relevance of high resolution and variable shape image properties,” *Applied Intelligence*, pp. 1–22, 2022.

[19] A. Quattoni and A. Torralba, “Recognizing indoor scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 413–420.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[21] S. J. Reddi, S. Kale, and S. Kumar, “On the convergence of adam and beyond,” *arXiv:1904.09237*, 2019.

[22] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, “Places: A 10 million image database for scene recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[23] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[24] A. Shrikumar, P. Greenside, and A. Kundaje, “Learning important features through propagating activation differences,” in *International Conference on Machine Learning*. PMLR, 2017, pp. 3145–3153.

[25] V. Miglani, N. Kokhlikyan, B. Alsallakh, M. Martin, and O. Reblitz-Richardson, “Investigating saturation effects in integrated gradients,” *arXiv preprint arXiv:2010.12697*, 2020.