Combining Similarity and Adversarial Learning to Generate Visual Explanation: Application to Medical Image Classification

Martin Charachon
Incepto Medical - Université Paris-Saclay, CentraleSupélec, MICS

Céline Hudelot
Université Paris-Saclay, CentraleSupélec, MICS

Paul-Henry Cournède
Université Paris-Saclay, CentraleSupélec, MICS

Camille Ruppli
Incepto Medical

Roberto Ardon
Incepto Medical

Abstract—**Explaining decisions of black-box classifiers is paramount in sensitive domains such as medical imaging since clinicians confidence is necessary for adoption. Various explanation approaches have been proposed, among which perturbation based approaches are very promising. Within this class of methods, we leverage a learning framework to produce our visual explanations method. From a given classifier, we train two generators to produce from an input image the so called similar and adversarial images. The similar image shall be classified as the input image whereas the adversarial shall not. Visual explanation is built as the difference between these two generated images. Using metrics from the literature, our method outperforms state-of-the-art approaches. The proposed approach is model-agnostic and has a low computation burden at prediction time. Thus, it is adapted for real-time systems. Finally, we show that random geometric augmentations applied to the original image play a regularization role that improves several previously proposed explanation methods. We validate our approach on a large chest X-ray database.**

Index Terms—**Deep learning, Classification, Interpretation, Explainable AI, Medical Images, Adversarial Example**

I. Introduction

Deep neural models have enabled to reach high performances on various applications and in particular in medical image analysis [1]. In this field, additionally to high accuracy, it is critical to provide interpretative explanations of the system decision since the clinicians' confidence in the system is at stake [2]. Thus, several methods were proposed to address the visual explanation problem, ranging from saliency maps [3]–[6] and class activation mapping methods [7], [8] to perturbation maps [9], [10]. However the problem is still open since there is no general consensus on their performances. Independently, under the motivation of model safety, several methods were proposed to generate "fake" images that "fool" classification algorithms. Such "fooling" images are generated either by the addition of a small perturbation on top of the input image [11]–[13], or as a complete new image, very close to the input [14]. Most of these works point out that adversarial generation allows to study the model robustness and fragility but very few make links with the explainability problem.

In this work, we propose to leverage adversarial generation methods to produce interpretative explanation of classifier's decision. Inspired by [10] and [14] who both train model to generate respectively explanation masks and adversaries, we propose to train a model to generate images that capture discriminative structures with the following key contributions:

- A new framework of explanation is proposed. We define the explanation of a classifier's decision as the difference between a regularized adversarial example and the "projection" of the original image into the space of adversarial generated samples.
- We introduce a new optimization workflow that combines the training of an adversarial generator and of a similar generator that "projects" the original image into the adversarial space.
- We propose a new method that greatly improves the use of several explanation methods as means to localize decisive objects in the image: Namely, averaging of registered explanation results built upon random geometric augmentations of the input image.

We validate our algorithm on a binary classification problem (pathological/healthy) of a large database of X-ray chest images. We compare our technique to state of the art approaches such as Gradient [3], GradCAM [8], BBMP [9], Mask Generator [10].

II. Related Work

Explaining classifiers decisions via some visual map has been the subject of several contributions. We here classify in three categories a selected few.

A. Back-propagation-based methods

These methods leverage, for neural networks and for a given image, back-propagation of small variations of the model's prediction [3]. While providing interesting results, these methods tend to produce noisy explanation maps since any variations of the model's output is considered. Many contributions are thus focused on building sharper and smoother explanation maps [4], [6], [15], [16]. In [7], explanation

maps are produced by upsampling activations from the last convolutional layer to the input image size, GradCAM [8] (or its application in medical domain [17]), builds on this work by computing the gradient of the output with respect to the last convolutional layer (and not only with respect to the model's prediction) generating compelling results. For an exhaustive review, the reader can refer to [18] and [19]. As a limitation, these approaches are not model-agnostic (limited to neural networks) and need access to intermediate layers.

B. Iterative perturbation-based methods

The principle of this approaches is to exploit the effects of perturbations to the input image on the model's output [15]. For instance, LIME [20] proposes a local explanation by perturbing random segments of the input image and training a linear classifier to predict the importance of each segment for the classifier's prediction.

The authors of [9] take this idea further by defining their explanation maps as the result of an optimization procedure over the input image and the model to explain. Considering a fixed perturbation, their approach consists in learning the maps that maximally impact the model or on the contrary the maps that enable to preserve its performance. Similarly, building on [21] for the medical imaging domain, [22], [23] adopt a similar optimization formulation but focus on the perturbations. They use generative methods to, respectively, perturb pathological images by local in-painting healthy tissues within pathological images or completely reconstructing a healthy image.

As noisy outputs is a major concern within these methods, some authors focus on regularization terms [24] and others filter gradients during back-propagation [25]. Different optimization formulations were also introduced [26], [27]. In [28], an explanation is generated through features perturbation at different levels of the neural network. In [12], [29] the optimization problem is revisited as an adversarial example generation, where the adversarial perturbation is sought in a regularized and restricted space.

Note that all these methods have in common the necessity to solve, for each image, an optimization problem in order to produce an explanation map. This translates into a high computational cost, as several iterations are needed for convergence (often inappropriate when a real time response is expected). Moreover, since the optimization problem is solved for every image, over-fitting issues arise. Explanation maps often contain features not linked to the models' behaviour but only to the image being processed. Strong and elaborated regularization is thus a necessity.

C. Trained perturbation-based methods

In order to alleviate computational needs of iterative perturbation-based methods, [10] proposes to evolve to an optimization problem on the whole database, thus learning a masking model. In medical imaging, [30] uses the same approach on a single class problem. Despite the benefits of this optimization strategy, two main drawbacks remain. First, perturbations are provided as parameters to be set and adapted manually. Their choice is impacted not only by the database and the considered classifier but also by the training of this classifier (e.g. a random noise perturbation has no effects on models trained to be robust to this noise). Since perturbations are manually selected, residual adversarial artefacts (without any link to the explanation) are still generated. Second, a costly hyperparameters tuning is needed to control the size of the generated explanation masks.

III. METHODOLOGY

We present our methodology to generate explanations for image classification outcomes. As for the methods exposed in section II, our explanation is given as a visual explanation map where higher values code for more important areas on the image w.r.t to the classifier decision. For the sake of simplicity, we present the rationale behind our approach in the case of a binary classification problem, the extension to the multi-class case being presented in section III-C. Let f_c be the studied classifier outputting a classification score in the range $[0, 1]$. Without loss of generality, we assign label 1 (resp. label 0) to an input image if its classification score (f_c value) is over 0.5 (resp. under 0.5). In the case of a different threshold one can apply for instance a piece-wise linear transformation to f_c to satisfy this condition.

A. Explanation through similar and adversarial generations

1) Adversarial naive formulation: A naive, yet novel, approach to reach our objective is to combine a trainable masking model [10] with adversarial perturbations for visual explanations [12]. The visual explanation map is then given by the difference between the input image and its adversary. This method is no longer dependent on the choice of a perturbation function since the adversarial sample "learns" this perturbation. For an input image x we define the ***naive*** explanation as

$$E_{f_c}^{naive}(x) = |x - \bar{g}_a(x)| \tag{1}$$

where $\bar{g}_a$ is a model obtained via a training process with the goal of "fooling" the classifier f_c while producing an image "very close" to x, written as follows:

$$\bar{g}_a = \underset{g_a}{\operatorname{argmin}} \quad \mathbb{E}_x \left[\|x - g_a(x)\|_2 \right] \\ s.t.\ f_c(g_a(x))) = 1 - f_c(x) \tag{2}$$

The mean value is taken over a training data set. Generating a visual explanation using (1) and (2) effectively counterbalances drawbacks of [10] & [12] but despite the regularization expected from the learning process, visual explanations are often corrupted by noise, highlighting regions which clearly should have no impact on the classifiers decision (see section V-B).

2) Similar-Adversarial formulation: Why does the ***naive*** formulation generate incoherent visual explanations? We argue that the flaw resides in the definition of explanation as expressed in equation (1). Comparing the original image with its generated adversarial sample exposes the method to a risk of reconstruction error. Some details of the original image can be

absent from the generated adversarial sample. However these details are not discriminative for the classifier in the sense that their sole presence would not change the classification score. More formally, the adversarial sample belongs to the target space of $\bar{g}_a$ (χ_a) which is different from the space of original images (χ_o). The comparison between x and $\bar{g}_a(x)$ inherits from the differences between χ_o and χ_a and these differences are not explicitly linked to the explanation problem by Equation (2). Since we do not have control on the original image space χ_o, we propose to mitigate this reconstruction risk by defining the explanation mask as the difference between the adversary $\bar{g}_a(x)$ and the closest element to x in χ_a on which f_c returns the same value as x. We call this element the *similar* image and it is denoted by $\bar{g}_s(x)$. $\bar{g}_s$ is the function mapping images to their similar counterparts. The rationale is to reduce the reconstruction error so that $E_{f_c}(x)$ only contains values related to the classifiers' decision and reads

$$E_{f_c}(x) = |\bar{g}_s(x) - \bar{g}_a(x)| \tag{3}$$

Denoting χ_s the target space of $\bar{g}_s$, both $\bar{g}_s$ and $\bar{g}_a$ are built through a joint optimization process aiming to make χ_s and χ_a as "close" as possible while satisfying $f_c(\bar{g}_s(x)) = f_c(x)$ and $f_c(\bar{g}_a(x)) = 1 - f_c(x)$:

$$
(\bar{g}_s, \bar{g}_a) = \underset{g_s, g_a}{\arg\min} \mathbb{E}_x
\begin{pmatrix}
d_{\chi_o, \chi_s}(x, g_s(x)) + \\
d_{\chi_o, \chi_a}(x, g_a(x)) + \\
d_{\chi_s, \chi_a}(g_s(x), g_a(x))
\end{pmatrix}
+ d(g_s, g_a)
$$
$$
s.t
$$
$$
f_c(g_s(x))) = f_c(x)
$$
$$
f_c(g_a(x))) = 1 - f_c(x)
$$
$$\tag{4}$$

where d_{χ_o, χ_s}, d_{χ_o, χ_a} and d_{χ_s, χ_a} are distance functions between elements of the different image spaces while $d(g_s, g_a)$ is a measure between the two functions. Henceforth, we refer to $\bar{g}_s$ and $\bar{g}_a$ as the similar and adversarial generators respectively.

B. Weaker formulation: Objective function

We propose to solve a weak formulation of the previous constrained optimization problem (4). We search for both similar and adversarial generators as minimizers of the following unconstrained problem

$$
(\bar{g}_s, \bar{g}_a) = \underset{g_s, g_a}{\arg\min}
\left\{
\begin{array}{l}
\mathbb{E}_x \begin{pmatrix}
L_d(x, g_s(x), g_a(x)) \quad + \\
L_{f_c}(x, g_s(x), g_a(x)) \quad + \\
L_{reg}(x, g_s(x), g_a(x))
\end{pmatrix} \\
\\
+ \quad L_{s,a}(g_s, g_a)
\end{array}
\right\}
\tag{5}
$$

L_d is a similarity loss that accounts for the term $d_{\chi_o, \chi_s} + d_{\chi_o, \chi_a} + d_{\chi_s, \chi_a}$ in equation (4) and enforces the proximity between x, $g_s(x)$ and $g_a(x)$. L_{f_c}, the classification loss, is a weak formulation of the classification constraints in (4) enforcing the similarity between $f_c(x)$ and $f_c(g_s(x))$ and their dissimilarity with $f_c(g_a(x))$. $L_{s,a}$ enforces the similarity between g_s and g_a ($d(g_s, g_a)$). In addition to the terms of (4), L_{reg} acts on the difference $(g_s(x) - g_a(x))$ to enforce regularity. An embodiment of optimization problem (5) when

using neural networks is given in Figure 1 (see section IV-C). We next specify the choices made in our method for each of the terms in Equation (5).
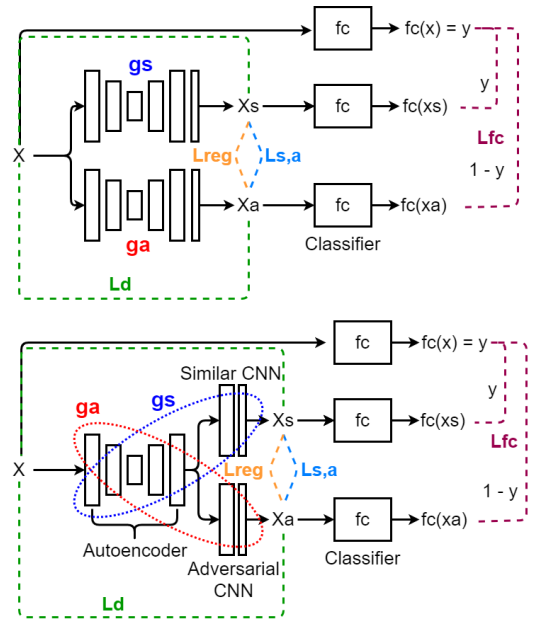

Fig. 1. Overview of Duo AE (Top) and Single AE (Bottom)

1) *Similarity Loss L_d:* Is defined as

$$
L_d(x, g_s(x), g_a(x)) =
\begin{array}{l}
\alpha_1 \|x - g_s(x)\|_2 + \\
\alpha_2 \|x - g_a(x)\|_2 + \\
\alpha_3 \|g_s(x) - g_a(x)\|_2 + \\
\alpha_4 \|g_s(x) - g_a(x)\|_1
\end{array}
\tag{6}
$$

where parameters α_1, α_2, α_3, $\alpha_4 \in \mathbb{R}$ adjust the importance attached to the different terms. Combining L_1 and L_2 norms to enforce similarity between $g_s(x)$ and $g_a(x)$ produces better results experimentally (as in [14]).

2) *Classification Loss L_{f_c}:* Is defined as

$$
L_{f_c}(x, g_s(x), g_a(x)) =
\begin{array}{l}
\beta_1 L_{bce}(f_c(x), f_c(g_s(x)) + \\
\beta_2 L_{bce}(1 - f_c(x), f_c(g_a(x))
\end{array}
\tag{7}
$$

where β_1, $\beta_2 \in \mathbb{R}$ are weighting parameters and L_{bce} is the binary cross entropy loss. This term accounts for the weak formulations of constraints in (4), favoring classifier f_c to act on $g_s(x)$ as it acts on x and in the opposite manner for $g_a(x)$.

3) *Generator Loss $L_{s,a}$:* Is a measure of the distance between the two generators. In the particular case (see section IV-C) where they both are neural networks (parameterized by w_s and w_a respectively) we used the following metric

$$
L_{s,a}(g_s(., w_s), g_a(., w_a)) = \gamma \left\| \sum_k w_s^k - w_a^k \right\|_2
\tag{8}
$$

where we assume generators g_s and g_a to have the same architecture. $\gamma \in \mathbb{R}$ is a weighting parameter. Note that metrics used in GANs to measure discrepancies between distributions [31] may also be considered.

4) Regularization Loss L_{reg}: Is defined as

$$L_{reg}(x, g_s(x), g_a(x)) = \lambda \sum_{i \in \mathbb{R}^d} \left\| \nabla \left(g_s^i(x) - g_a^i(x) \right) \right\|_2 \quad (9)$$

where parameter $\lambda \in \mathbb{R}$ controls the relative importance of L_{reg} with respect to the other terms of L (5) and d is the dimension of the output space of the generators. This term acts as $L_{s,a}$ favoring the proximity of g_s and g_a and regularizes the explanation map (3).

C. Multi-class situation

Weak optimization problem (5) can be adapted to the multi-class problem by modifying L_{f_c} to account for a vector valued $\mathbf{f_c} = [f_{c_i}]_{i \in [|1, \cdots N|]}$ function. This boils down to modifying term (7) adapting CW loss of [12], [14] into

$$L_{f_c} = \begin{array}{l} \beta_1 \max(\max\limits_{i \neq l}(f_{c_i}(g_s(x))) - f_{c_l}(g_s(x)), -\kappa) + \\ \beta_2 \max(f_{c_l}(g_a(x)) - \max\limits_{i \neq l}(f_{c_i}(g_a(x))), -\kappa) \end{array} \quad (10)$$

where index l is defined by $\operatorname{argmax}\limits_{i}([f_{c_i}(x)])$ corresponding to the class selected by the classifier on input x. κ is a strictly positive margin.

D. Explanation and augmentations

As our visual explanation is defined as the difference between two generated images, we suggest to regularize the output of our explanation method by averaging all outputs on random geometrical transformations of the input image. Thus, discriminative regions against reconstruction errors are further enforced. This average reads:

$$\overline{E}_{f_c}(x) = \frac{1}{N+1} \left[E_{f_c}(x) + \sum_{i=1}^{N} \psi_i^{-1} \left(E_{f_c}(\psi_i(x)) \right) \right] \quad (11)$$

where ψ_i are random geometric transformations such as rotations, translations, zoom, axis flip, etc. This particular regularization can be applied to all other visual explanation techniques (see section V-B).

In the following sections, we denote by $x_s = \bar{g}_s(x)$, $x_a = \bar{g}_a(x)$ the output of similar and adversarial generators respectively.

IV. EXPERIMENTS

A. Datasets

We tested our approach on a publicly available Chest X-rays dataset for a binary classification problem. The Chest X-rays dataset comes from the RSNA Pneumonia Detection Challenge dataset which is a subset of 26,684 exams in dicom taken from the NIH CXR14 dataset [32]. We only extracted healthy and pneumonia cases from the original dataset. The resulting database is composed of 14,863 exams: 6,012 pneumonia - 8,851 healthy. We split the dataset into 3 random groups (80%, 10%, 10%) : train (11,917) - validation (1,495) - test (1,451). X-rays exams with opacities contain bounding box ground truth annotations.

B. Classifier Set Up

The classification model whose decisions need to be explained consists of a ResNet50 [33]. We adapt the last layers of the ResNet50 network in order to tackle a binary classification task (healthy/pathology). We transfer the pre-trained backbone layers from Imagenet [34] to our binary classifier. Then, the network is trained on the whole training set for 50 epochs with a batch size of 32. We use the Adam optimizer [35] with an initial learning rate of 1e-4. Original X-rays are resized from 1024x1024 to 224x224 and normalized to $[0, 1]$. We also used zoom, translations, rotations and vertical flips as random data augmentations. The binary classifier achieves an AUC of 0.974 on the test set.

C. Generative Explanation Model

For the similar and adversarial generators, as in [14], [36], generators roughly follows the UNet architecture [37]. We propose two different types of generators: (i) *Duo AE* (Figure 1 - Top): g_s and g_a are two separated UNet auto-encoders. (ii) *Single AE_i* (Figure 1 - Bottom): g_s and g_a share a same auto-encoder part that captures image structure for both generators. They differ by two identical convolutional neural networks connected at the end of the common autoencoder. Index i indicates the number of convolutional layers in the separated CNN.

Generators take as input the same image as the classifier with 3 channels and dimensions 224x224. Both generators are trained simultaneously for 70 epochs with a batch size of 8 for *Single AE_i* and 4 for *Duo AE* , with the same augmentations used for the classifier. Adam optimizer is used with an initial learning rate of 1e-4, and we reduce the learning rate by 3 each time the loss does not decrease after 3 epochs. Through trial an error we selected the objective loss function (5) parameters providing the best results and summarized them in Table I.

TABLE I
SELECTED PARAMETERS FOR COUPLES OF GENERATORS

Model name	α_1	α_2	α_3	α_4	$\beta_{1,2}$	γ	λ
Duo AE (TV)	1	1	1	0	0.001	0	0.2
Duo AE (W,TV)	1	1	1	0	0.001	0.1	0.2
Single AE_1 (TV)	1	1	1	0	0.001	0	0.2
Single AE_1 (W)	3	1	1	0.2	0.001	0.1	0
Single AE_1 (W,TV)	1	1	1	0.2	0.001	0.1	0.2
Single AE_2 (W)	3	1	1	0.2	0.001	0.2	0
Single AE_2 (W, TV)	3	1	1	0.2	0.001	0.2	0.2

D. Augmentation during generator's prediction

During generator's prediction, for each image x, we generate 10 augmented images $(x_i)_{i \in [|1,10|]}$ with random geometric transformations of parameters described in Table II

E. Method Evaluation

Generators Evaluation
The evaluation is achieved on the classifier's test set. For similar and adversarial generators $\bar{g}_s$ and $\bar{g}_a$, we respectively evaluate the similarity between x, x_s and x_a. Structural

Transformation	value(s)
Rotations range (°)	$[-5, 5]$
Height shift range (pixels)	$[-10, 10]$
Width shift range (pixels)	$[-10, 10]$
Zoom range	$[0.9, 1]$
Random horizontal flip	(True, False)
Random vertical flip	(True, False)

Similarity Index (SSIM), as well as the Peak Signal to Noise Ratio (PSNR) are used to evaluate the similarity between pair of images. For the classification purpose, we compute the area under the ROC curve between the rounded value of the classifier predictions $R(f_c(x))$ (resp. $R(1 - f_c(x))$) and $f_c(x_s)$ (resp. $f_c(x_a)$).

Interpretability Evaluation
In state of the art methods and more specifically in medical imaging, a visual explanation is considered as interpretable if: (i) The highlighted regions coincide with discriminative regions for humans. In our classification problem, salient regions should overlap opacity regions where the pathologies are found. (ii) The highlighted regions coincide with context regions that are also discriminative for humans. We can quantitatively assess the overlap between explanation map and ground truth annotations by conducting a weak localization experiment. We use two metrics to evaluate the localization performance: the intersection over union (IOU) and an estimated area under the curve (AUC_{Loc}). We compute the intersection over union between the ground truth mask M_{GT} and the thresholded explanation mask M_{Ei}, as defined in (12).

$$IoU_i = \frac{M_{GT} \cap M_{Ei}}{M_{GT} \cup M_{Ei}} \qquad (12)$$

where M_{GT} is the binary mask included inside the ground truth bounding box annotation, and M_{Ei} is the binary mask obtained when we threshold the explanation mask E_{f_c} at the i-th percentile p_i:

$$M_{Ei} = \begin{cases} 1 & E_{f_c} \geq p_i \\ 0 & \text{otherwise} \end{cases} \qquad (13)$$

We also measure the precision and the sensitivity of the localization for different thresholds p_i in order to compute the area under the precision and recall curve as introduced in [38]:

$$AUC_{Loc} = \sum_i P_i(R_i - R_{i-1}) \qquad (14)$$

where $P_i = \frac{M_{GT} \cap M_{Ei}}{M_{Ei}}$, $R_i = \frac{M_{GT} \cap M_{Ei}}{M_{GT}}$ and $i \in [|1, 100|]$. Our estimation of AUC_{Loc} differs from [38] as we only compute the metrics over the hundred values of percentile instead of all sorted values of the explanation map.
We also compute a partial AUC_{Loc} for percentiles between 80 and 100 as it is more representative of the volume occupied by the ground truth mask M_{GT}. We show some statistics of the bounding box annotations in Table III.

We compare our proposed method to the **naive** one (see section III-A) and to the following state of the art approaches: Gradient [3], Smooth-Gradient [4], Input Gradient [5], Integrated Gradient [6], GradCAM [8], BBMP [9], Mask Generator [10] and Perceptual Perturbation [12]. The best BBMP results are reached when looking for a mask at 56x56 and with Gaussian blur perturbation. The mask Generator follows the UNet architecture described in [10], but we remove the class selector and adapt the objective function to a single class problem. The best results are obtained when we generate a mask at size 112x112 and then upsample it to image dimensions. For Perceptual perturbation which is not model-agnostic, we regularize the first ReLU layer of each convolution block of the ResNet50 classifier. We also adapt the optimization to a single class problem.

Metrics	Height (pixels)	Width (pixels)	Area Ratio (%)
Min	13	13	0.5
Max	171	91	25.3
Mean	71.8	47.5	7.3
Median	67.8	46.8	6.3

V. RESULTS & DISCUSSION

A. Generator evaluation

For the different architectures and optimization tested, both generators g_s and g_a reach high performance in term of classification. As shown in Table IV, similar images are almost all classified as the original ones, as the AUC_s almost reaches 1. Adversarial images achieve better adversarial attacks either when the network (Single) or the weights regularization (W) causes the generators g_s and g_a to be close to each other (Table IV). They even outperform the **naive** approach (Adv. AE (TV)) where g_a is trained without g_s.

Explanation method	AUC_s	AUC_a
Naive	-	0.939
Duo AE (TV)	1.0	0.905
Duo AE (W,TV)	1.0	0.958
Single AE_1 (TV)	1.0	0.961
Single AE_1 (W)	0.998	0.952
Single AE_1 (W,TV)	0.997	0.944
Single AE_2 (W)	0.998	0.949
Single AE_2 (W, TV)	0.998	0.952

For the similarity, both generators produce samples visually highly similar to original images (see Figure 2 and Table V). Similar images x_s best perform for both SSIM and PSNR when generators are not constrained by weight regularization. At the opposite, adversarial images x_a increase their similarity to both x and x_s when generators are constrained, and it even outperforms the **naive** adversarial generator trained on

its own. In our case, the objective is to produce x_s and x_a as close as possible in order to reduce non discriminative differences, while having x_s very close to x. As shown in Table V, Single AE_2 regularized with weights proximity (W) produce highly similar samples x_s and x_a, while maintaining a strong similarity between x_s and x.
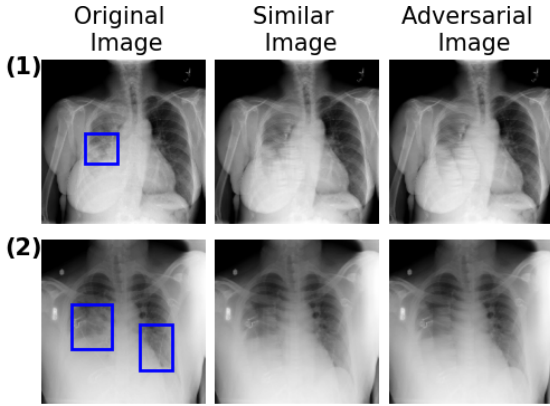

Fig. 2. Examples of original images with respective similar and adversarial generated images. **Case (1)**: $f_c(x) = 1.0$ - $f_c(x_s) = 0.999$ - $f_c(x_a) = 0.051$ - PSNR between orignal and similar image $PSNR_{os} = 43.18$ - PSNR between orignal and adversarial image $PSNR_{oa} = 42.07$ - PSNR between similar and adversarial image $PSNR_{sa} = 52.75$. **Case (2)**: $f_c(x) = 0.978$ - $f_c(x_s) = 0.986$ - $f_c(x_a) = 0.34$ - $PSNR_{os} = 46.30$ - $PSNR_{oa} = 45.49$ - $PSNR_{sa} = 56.39$

TABLE V
SIMILARITY METRICS BETWEEN GENERATED AND ORIGINAL IMAGES

Explanation method	$x \leftrightarrow x_s$		$x \leftrightarrow x_a$		$x_s \leftrightarrow x_a$	
Metrics	ssim	psnr	ssim	psnr	ssim	psnr
Adv. AE (TV)	-	-	0.994	41.92	-	-
Duo AE (TV)	0.996	44.07	0.987	39.47	0.994	43.89
Duo AE (W,TV)	0.995	41.99	0.987	39.08	0.995	44.26
Single AE_1 (TV)	**0.997**	**44.57**	0.989	40.67	0.996	45.25
Single AE_1 (W)	0.994	42.73	0.993	41.85	0.999	52.59
Single AE_1 (W,TV)	0.992	41.79	0.991	41.35	**0.999**	**54.55**
Single AE_2 (W)	0.995	43.61	0.994	42.42	0.999	52.26
Single AE_2 (W, TV)	0.995	43.88	**0.994**	**42.63**	0.999	51.93

B. Weak localization evaluation

As shown in Table III, bounding box annotations of the test set occupy from 0.5 to 25.3 % of the image with an average occupation of 7.3%. For different generators and regularizations, we accordingly list the results of the averaged IOU for p_i between the 80th and 100th percentile value in Table VI, and total and partial AUC_{Loc} in Table VII. Firstly, Single AE clearly outperforms the Duo version for all IOU and AUC scores. The Single AE approach compelled g_a and g_s to capture the same information on the original image by sharing a common autoencoder. As shown in Table V, the proximity between x and x_a as well as between x_s and x_a is better for Single AE approaches. Then, the weights

TABLE VI
IOU SCORES AT DIFFERENT THRESHOLDS OF BINARIZATION -
COMPARISON ACROSS THE DIFFERENT GENERATORS ARCHITECTURES

Explanation method	IOU				
Percentile	80	85	90	95	98
Duo AE (TV)	0.190	0.182	0.164	0.122	0.070
Duo AE (W,TV)	0.188	0.184	0.170	0.132	0.079
Single AE_1 (TV)	0.187	0.182	0.166	0.127	0.075
Single AE_1 (W)	0.227	0.222	0.204	0.157	0.090
Single AE_1 (W,TV)	0.234	0.235	0.220	0.171	*0.099*
Single AE_2 (W)	0.240	0.245	0.229	0.172	0.095
Single AE_2 (W, TV)	*0.248*	*0.250*	*0.232*	*0.173*	0.097
With Augmentations					
Duo AE (TV)	0.243	0.232	0.206	0.156	0.085
Duo AE (W,TV)	0.263	0.253	0.227	0.166	0.093
Single AE_1 (TV)	0.262	0.249	0.218	0.156	0.086
Single AE_1 (W)	0.262	0.254	0.233	0.181	0.105
Single AE_1 (W,TV)	0.268	0.261	0.240	0.188	0.112
Single AE_2 (W)	0.288	0.288	0.268	0.204	**0.115**
Single AE_2 (W, TV)	**0.292**	**0.292**	**0.272**	**0.206**	**0.115**

regularization between similar path and adversarial path introduced in (8) improves all the localization performance e.g. from $IOU_{90} = 0.166$ to $IOU_{90} = 0.220$ for Single AE_1 (TV). This is consistent with the findings in V. Total variation regularization on the resulting explanation mask also slightly increases IOU and AUC scores for Single $AE_{1,2}$. In addition, the Single generator with two convolutional layers (AE_2) performs better than the single-layer one (AE_1).

Finally, the use of augmentations during generator's prediction improves localization scores for all cases e.g. up to 4 points for IOU_{90} (Table VI), from 7 to 11 points for total and partial AUC_{Loc} (Table VII).

TABLE VII
ESTIMATED AUC SCORES FOR PRECISION-RECALL - COMPARISON
ACROSS THE DIFFERENT GENERATORS ARCHITECTURES

Explanation method	Total AUC	Partial AUC
Duo AE (W,TV)	0.257	0.162
Single AE_1 (TV)	0.253	0.157
Single AE_1 (W)	0.310	0.220
Single AE_1 (W, TV)	0.325	0.239
Single AE_2 (W)	0.325	0.248
Single AE_2 (W, TV)	**0.339**	**0.256**
With Augmentations		
Duo AE (W,TV)	0.362	0.263
Single AE_1 (TV)	0.353	0.254
Single AE_1 (W)	0.370	0.274
Single AE_1 (W,TV)	0.381	0.287
Single AE_2 (W)	0.405	0.322
Single AE_2 (W, TV)	**0.412**	**0.328**

When compared to state of the art methods (Tables VIII, IX), Single AE_2 (W, TV) achieves comparable localization scores. Our method even slightly outperforms the best performers Mask Generator and BBMP for IOU scores for percentile thresholds from 80 to 95 %. It is also the case for both partial and total AUC compared to the best state of the art approaches: GradCAM, BBMP and Mask Generator. Only Mask Generator and Gradient outperform or compete with our method for IOU_{98}. We can also note that the ***naive***

explanation directly defined as the difference between x_a and x (Adv. AE (TV)) produces much poorer results.

However, when using augmentation during generator prediction phase, our method outperforms all the others. Visual illustrations are given in Figures 3 and 4 for cases where the opacities are located either at one or two different positions. When thresholding heatmaps at the 95th percentile, our method (*Single AE*) seems to generate less noisy masks than other approaches including the ***naive*** one (*Adv AE*), while capturing all discriminative structures. In addition, our method is suitable for real time situation as suggests the generation time per image of the explanation given in Table IX (on NVIDIA GPU MX130).

Fig. 4. Explanation maps generated by different methods in case of two ground truth bounding box annotations. *Top row*: the original image with the explanation map and the ground truth bounding box. *Bottom row*: Binary heatmaps for the 95th percentile

TABLE VIII
IOU SCORES AT DIFFERENT THRESHOLDS OF BINARIZATION - COMPARISON TO STATE OF THE ART METHODS

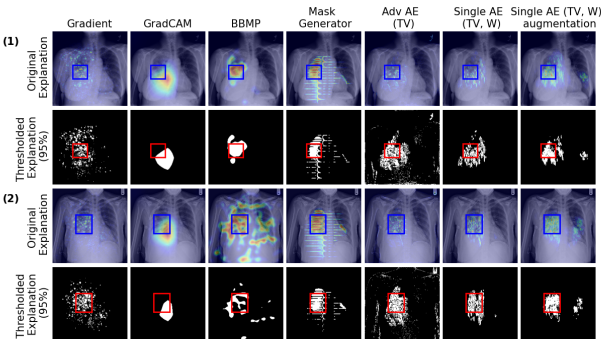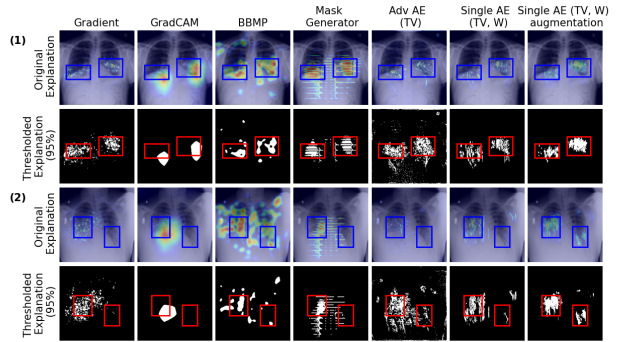| Explanation method	IOU				
Percentile	80	85	90	95	98
Gradient	0.203	0.199	0.187	0.152	0.097
Smooth Grad.	0.192	0.188	0.176	0.143	0.091
Input Grad.	0.191	0.185	0.170	0.136	0.086
Integrated Grad.	0.176	0.171	0.157	0.124	0.077
GradCAM	0.237	0.225	0.195	0.138	0.070
BBMP	0.233	0.226	0.204	0.154	0.087
Perceptual Perturbation	0.133	0.125	0.110	0.080	0.045
Mask Generator	0.222	0.219	0.208	0.169	*0.103*
Adv. AE (TV)	0.177	0.173	0.158	0.118	0.064
Adversarial vs Similar					
Single AE_2 (W, TV)	*0.248*	*0.250*	*0.232*	*0.173*	0.097
Adv. vs Sim. + Augment.					
Single AE_2 (W, TV)	**0.292**	**0.292**	**0.272**	**0.206**	**0.115**

Fig. 3. Examples of explanation maps generated by different methods in case of a single ground truth bounding box annotation. *Top row*: the original image with the explanation map and the ground truth bounding box. *Bottom row*: Binary heatmaps for the 95th percentile

As an additional experiment, we apply the augmentation technique to other state of the art methods that produce their visual explanation in one shot. Localization results are listed in Tables X and XI. All localization scores improve, while the generation time per image remains adequate (see Table XI). By using augmentations, we observe for all methods a gain similar to that observed for our method. Our best method still

TABLE IX
ESTIMATED AUC SCORES FOR PRECISION-RECALL AND COMPUTATION TIME - COMPARISON TO STATE OF THE ART METHODS

Explanation method	Total AUC	Partial AUC	Time (s)
Gradient	0.287	0.189	2.04
Integrated Grad.	0.244	0.146	1.93
GradCAM	0.324	0.235	0.78
BBMP	0.326	0.229	17.14
Perceptual Perturbation	0.180	0.084	30.74
Mask Generator	0.327	0.226	0.09
Adv. AE (TV)	0.238	0.145	0.10
Adversarial vs Similar			
Single AE_2 (W, TV)	*0.339*	*0.256*	0.05
Adv. vs Sim. + Augment.			
Single AE_2 (W, TV)	**0.412**	**0.328**	0.63

achieves better localization results for AUC metrics. For IOU, Mask Generator outperforms our method for $p_i \geq p_{95}$.

TABLE X
IOU SCORES AT DIFFERENT THRESHOLDS OF BINARIZATION - COMPARISON TO STATE OF THE ART METHODS WITHOUT (**TOP**) AND WITH (**BOTTOM**) AUGMENTATIONS

Explanation method	IOU				
Percentile	80	85	90	95	98
Gradient [1]	0.203	0.199	0.187	0.152	0.097
	0.256	*0.252*	*0.236*	*0.190*	*0.117*
GradCAM [2]	0.237	0.225	0.195	0.138	0.070
	0.271	*0.263*	*0.244*	*0.190*	*0.105*
BBMP [3]	0.233	0.226	0.204	0.154	0.087
Mask Generator [4]	0.222	0.219	0.208	0.169	0.103
	0.259	*0.264*	*0.259*	*0.221*	*0.137*
"Naive"	0.177	0.173	0.158	0.118	0.064
	0.239	*0.230*	*0.208*	*0.156*	*0.087*
Ours	0.248	0.250	0.232	0.173	0.097
	0.292	**0.292**	**0.272**	*0.206*	*0.115*

VI. CONCLUSION

In this work, we introduce a new method to produce a visual explanation of the classifier's decision that leverages adversarial generation learning. We propose to train simultaneously

Explanation method	Total AUC	Partial AUC	Time (s)
Gradient [1]	0.287	0.189	2.04
	0.374	*0.274*	2.83
GradCAM [2]	0.326	0.235	0.78
	0.397	*0.302*	5.09
BBMP [3]	0.326	0.229	17.14
Mask Generator [4]	0.327	0.226	0.09
	0.404	*0.308*	0.68
"Naive"	0.238	0.145	0.10
	0.325	*0.232*	0.75
Ours	0.339	0.256	0.05
	0.412	**0.328**	0.63

a couple of generators to produce an adversarial image that goes against the classifier's decision, and a similar image that is classified as the original one. We show that the differences between the two images as well as the learning procedure helps to better capture discriminative features. We have tested our method on a binary classification problem in the medical domain. We have shown that our method outperforms state of the art techniques in terms of weak localization, especially when we introduced geometric augmentations during the generation phase. Unlike some state of the art methods, our proposed method is both model-agnostic and sufficient for real time situation such as medical image analysis. Finally, we show that random geometric augmentations applied to the original image improves all the tested state of the art approaches.

In future works, we shall generalize our method to multi-classification problems and apply it to 3D medical image problems.

REFERENCES

[1] A. Esteva, B. Kuprel, R. Novoa, J. Ko, S. Swetter, H. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," in *Nature*, vol. 542, 2017, pp. 115—118.

[2] A. Holzinger, C. Biemann, C. S. Pattichis, and D. B. Kell, "What do we need to build explainable ai systems for the medical domain?" *ArXiv*, vol. abs/1712.09923, 2017.

[3] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," in *ICLR*, 2014.

[4] D. Smilkov, N. Thorat, B. Kim, F. B. Viégas, and M. Wattenberg, "Smoothgrad: removing noise by adding noise," *ArXiv*, vol. abs/1706.03825, 2017.

[5] Y. Hechtlinger, "Interpretation of prediction models using the input gradient," *ArXiv*, vol. abs/1611.07634, 2016.

[6] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *ICML*, 2017.

[7] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *CVPR*, 2016, pp. 2921–2929.

[8] 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 *ICCV*, 2017, pp. 618–626.

[9] R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," in *ICCV*, 2017, pp. 3449–3457.

[10] P. Dabkowski and Y. Gal, "Real time image saliency for black box classifiers," in *NIPS*, 2017, pp. 6967–6976.

[11] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," in *ICLR*, 2015.

[12] A. Elliott, S. Law, and C. Russell, "Adversarial perturbations on the perceptual ball," *ArXiv*, vol. abs/1912.09405, 2019.

[13] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. X. Song, "Generating adversarial examples with adversarial networks," in *IJCAI*, 2018.

[14] W. Zhang, "Generating adversarial examples in one shot with image-to-image translation gan," in *IEEE Access*, vol. 7, 2019, pp. 151 103–151 119.

[15] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *ECCV*, 2014.

[16] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. A. Riedmiller, "Striving for simplicity: The all convolutional net," in *ICLR*, vol. abs/1412.6806, 2015.

[17] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Y. Ding, A. Bagul, C. P. Langlotz, K. S. Shpanskaya, M. P. Lungren, and A. Y. Ng, "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning," *ArXiv*, vol. abs/1711.05225, 2017.

[18] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity Checks for Saliency Maps," *arXiv:1810.03292 [cs, stat]*, Oct. 2018, arXiv: 1810.03292. [Online]. Available: http://arxiv.org/abs/1810.03292

[19] S.-A. Rebuffi, R. Fong, X. Ji, and A. Vedaldi, "There and back again: Revisiting backpropagation saliency methods," in *CVPR*, 2020.

[20] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?": Explaining the predictions of any classifier," in *ACM SIGKDD*, 2016.

[21] C.-H. Chang, E. Creager, A. Goldenberg, and D. K. Duvenaud, "Explaining image classifiers by counterfactual generation," in *ICLR*, 2019.

[22] H. Uzunova, J. Ehrhardt, T. Kepp, and H. Handels, "Interpretable explanations of black box classifiers applied on medical images by meaningful perturbations using variational autoencoders," in *Medical Imaging: Image Processing*, 2019.

[23] D. Major, D. Lenis, M. Wimmer, G. Sluiter, A. Berg, and K. Bühler, "Interpreting medical image classifiers by optimization based counterfactual impact analysis," in *ISBI*, 2020, pp. 1096–1100.

[24] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," in *ICCV*, 2019, pp. 2950–2958.

[25] J. Wagner, J. M. Köhler, T. Gindele, L. Hetzel, J. T. Wiedemer, and S. Behnke, "Interpretable and fine-grained visual explanations for convolutional neural networks," in *CVPR*, 2019, pp. 9089–9099.

[26] A. Dhurandhar, P.-Y. Chen, R. Luss, C.-C. Tu, P.-S. Ting, K. Shanmugam, and P. Das, "Explanations based on the missing: Towards contrastive explanations with pertinent negatives," in *NeurIPS*, 2018.

[27] C.-Y. Hsieh, C.-K. Yeh, X. Liu, P. Ravikumar, S. Kim, S. Kumar, and C.-J. Hsieh, "Evaluations and methods for explanation through robustness analysis," *ArXiv*, vol. abs/2006.00442, 2020.

[28] A. Khakzar, S. Baselizadeh, S. Khanduja, S. T. Kim, and N. Navab, "Explaining neural networks via perturbing important learned features," *ArXiv*, vol. abs/1911.11081, 2019.

[29] W. Woods, J. Chen, and C. Teuscher, "Adversarial explanations for understanding image classification decisions and improved neural network robustness," in *Nature Machine Intelligence*, vol. 1, 2019, pp. 508–516.

[30] G. Maicas, G. Snaauw, A. P. Bradley, I. D. Reid, and G. Carneiro, "Model agnostic saliency for weakly supervised lesion detection from breast dce-mri," in *ISBI*, 2019, pp. 1057–1060.

[31] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *ICML*, 2017.

[32] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *CVPR*, 2017, pp. 3462–3471.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *ECCV*, 2016.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*, 2009.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[36] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *CVPR*, 2017.

[37] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015.

[38] W. Fu, M. Wang, M. Du, N. Liu, S. Hao, and X. Hu, "Distribution-guided local explanation for black-box classifiers," 2019.