Foreground segmentation to measure deep neural networks interpretability

AABID Mohsine
Aix-Marseille Unversité
mohsine.aabid@etu.univ-amu.fr

Ronan Sicre
LIS-Laboratory
ronan.sicre@lis-lab.fr

Abstract

Neural networks are precise and efficient models used in various fields such as medicine, industry and more, but they are difficult to understand because of their many parameters (black boxes) and even less to explain their decisions. This had led several experts to build methods in order to interpret them, The best known in computer vision is the CAM method [1] for (class activation map) which can be used to understand CNN decisions in image classification tasks. This method produces a heat map or saliency map indicating the area of interest of the network, the saliency map highlights the pixels that are the most important for the decision. Which means that a good heat map corresponds to a better explanation, so we want to evaluate the quality of a heat map with several methods to know when it is meaningful. Our objective will therefore be to present some commonly used metrics and to propose new approaches to evaluate generated heat-maps.

***Keywords*—** CNN, saliency map, Class activation maps, Metrics

1 Introduction

The interpretability of neural networks is a very interesting subject, but it is also very complicated one because it is still difficult to define mathematically what interpretability is, the only thing we have is non-mathematical definitions. Some experts believe it is the degree to which a human can understand the cause of a decision made by a model [2]. Another similar definition suggests that interpretability is the ability to provide explanations in understandable terms to a human [3]. We emphasize "understanding a human being" in the definition because this topic affects people's trust in machine learning, and it is relates to ethical issues (algorithmic discrimination), and finally can be used as a powerful tool in other scientific fields.

In image classification, basic neural networks do not give very good results because they fail to capture the most important features in the image. We then use convolutional networks that apply several filters to capture the maximum number of features in the images. There are several methods to interpret a convolutional neural network [4], we can cite the intermediate activation's methods, the filters methods, the deconvolutional methods and those , CAM methods, the class activation maps look for a given layer at the features in which the network is interested. To then output a heat map encompassing the centers of interest of the model. Once the heat maps have been produced, they must be evaluated to see if the interpretation of the CNN corresponds

to our interpretation of the image. Numerous methods generating such saliency maps derive from Class Activation Maps (CAM). Grad-CAM [5] and Grad-CAM++[6] use back-propagated gradient information, while Score-CAM [7] is based on layer activation's.In order to automate the process, one needs to formalize the evaluation process by setting up metrics that evaluate specific criteria in the heat map.

Metrics can be separated into two types, classification metrics and location metrics. Classification metrics measure the effect of classification performance when we wise-multiply the image by the saliency map. While the localization metrics compares the saliency map with the bounding ground truth boxes. There are different ways to evaluate a CNN, knowing that each metric evaluates a particular criterion of a heat map for example how closely the area of interest of the network overlaps with the area given as label. We will therefore use a lot of metrics to have the maximum possible information's on the interpretation of our CNN. We will also propose new metrics based on masks and we will compare these with the methods based on boxes. We are looking to make a comparison between the mask-based metrics and the box-based metrics. Know which of the two families is more precise ? We will try to answer this question throughout the document by showing both the strengths and weaknesses of each presented metric. Our hypothesis is that the mask-based ones are more accurate but will have almost the same results as the box-based ones.

FIGURE 1 – Example of heat map

2 State of the art

It is difficult to know what exactly happens in a CNN, sometimes it predicts the right results sometimes it is wrong in its prediction, we want to know what's our model misinterprets when he goes wrong, the image below corresponds to a misinterpretation, the network is focusing on something other than the cat,

FIGURE 2 – Example of misinterpretation

We want to be able to do this operation quickly with computers so it is necessary to formalize what is a good interpretation ? And this is where the metrics come into play, we want to measure

the number of correctly interpreted images in the set of test images.

There is a wide variety of metrics in the literature. each tries to assess a specific criterion. The commonly used metrics are classification metrics such as : "average drop" [6] which measures the loss of predictive power when using only the masked region instead of using the whole image. "average increase" or "increase confidence" [8] which calculates the percentage of images where the CNN gives a higher predictive power by using the masks instead of using the original images.

On the other hand we have location measurements where usually we try to compare the saliency maps with the bounding ground truth boxes, the ones that are most used are the location error [9] which compares the error of the overlap rate between the ground truth box and a box generated from a threshold. We also have the "standard pointing game" [10] which looks at whether the pixel with the most interest for the networks is contained in the enclosing ground truth box. There are still other metrics that we will present in more detail later in the document.

3 Class activation map (CAM)

In this part we will briefly explain how the CAM method generates salience maps for a convolutional network. We are trying to generate a heat map from a feature map generally taken from the last layer. The authors of CAM paper use a network that contains a large number of convolutional layers and before the output layer, they perform a global average pooling, they feed the obtained features to a fully connected layer that produce the desired output. The figure below summarize the procedure.

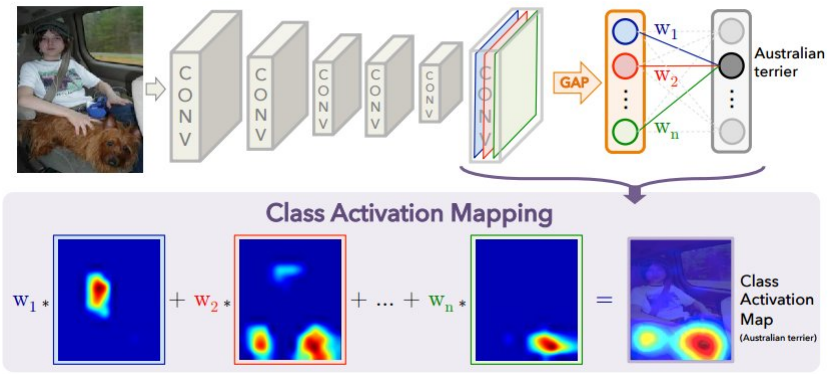

FIGURE 3 – The procedure for generating salience maps from cam paper

Let $f_k(x, y)$ represent the activation of a unit k in the last convolutional layer and w_c^k the weight for a class c of the same unit k. We define the class activation map as :

$$M_c(x, y) = \sum_k w_c^k f_k(x, y)$$

The class activation map is a weighted linear sum of the presence of visual patterns at different spatial locations.

4 Background

Before getting to the heart of the matter, it is important to explain certain concepts to ensure readers' understanding. First we will explain how we generally extract the bounding boxes from a saliency map. then we will explain what is the intersection over union (IoU) and the dice Coefficient also known as F1 score, which are two principles methods often used in the calculation of metrics.

4.1 Extract the bounding box from a binary mask

We want to extract the smallest bounding box which summarizes the areas of interest of the CNN from a heat map, a first problem we face is the disparity of the areas of interest, in some cases we can have several red areas with more interest but totally distant in this case should we take the box that contains both or just one that contains the largest of them ? The second is that the values contained in a map of heats are continuous, so we have to find a way to discretize the map to obtain a mask with 0 and 1 as pixels.

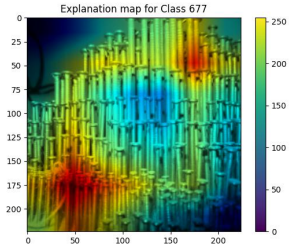

FIGURE 4 – Example of a heat map with two areas of interest.

To start we represent a saliency map as a matrix whose dimensions are the length and width of the image and whose values are between 0 for an uninteresting pixel, and 255 for a very interesting pixel. The method that comes up most often to discretize this matrix in papers is thresholding. By having a threshold we can binarize our saliency map by setting to 0 the pixels having a value lower than the threshold and 1 to the others. This is a very important step because for two thresholds we can see two completely different boxes, we will therefore risk advantaging some images and penalizing others depending on the chosen threshold. There are several ways to choose the threshold, in some articles it takes the threshold equal to 0.5, others use 20% of the global saliency in the map. We chose the method that takes the global mean, the average saliency. Note that the larger the threshold, the smaller the binary mask.

FIGURE 5 – Example of thresholding

After thresholding we often have several components (a connected pixel area with value 1

forming a connected component) in our image, the second step will therefor be to look for the largest components in the binary image, this will be used to define the outline of the box that will contain it. Again it's not a rule, but it's often what we do in practice. The figure below gives an overview of what we have just said.

FIGURE 6 – Largest component

The last step is to find the smallest mask bounding box. To do this, we just need to find the extreme pixels of the mask to define these boxes, here is what we find for the mask presented above.

FIGURE 7 – bounding box

4.2 Intersection over Union (IoU)

We present this method because it is used in many metrics. This is how it works, given two geometric shapes A and B, we want to know how much these two shapes overlap, for this we calculate the IoU which consists of taking the area of the intersection of the two shapes divided by the area of their union.

$$IoU = \frac{Area(A \cap B)}{Area(A \cup B)}$$

The figure below illustrates a little what we have just explained :

4.3 Dice Coefficient (F1)

Like the IoU it's a coefficient of good classification. here is how it is calculated :

$$F1 = \frac{2Area(A \cap B)}{Area(A) + Area(B)}$$

The metrics are positively correlated, in other words if the first one finds a good overlap the other does too, but they are not entirely equivalent [11], even though they represent different things, when we use them as an evaluation metric for different models, they are practically the same [12]. If we consider the binary image containing the bounding box of the ground truth

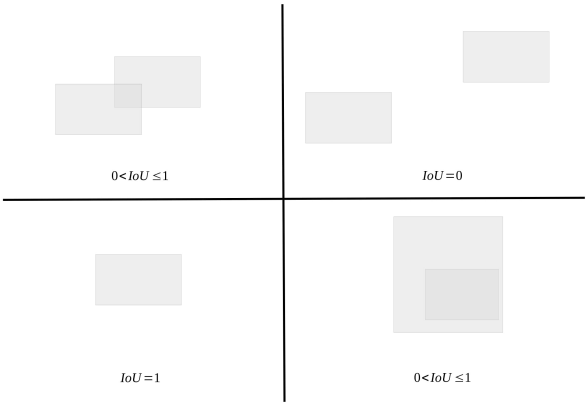

FIGURE 8 – Some examples of intersection over union for rectangles

as a reference, and the one with the box predicted from a threshold as a hypothesis, we can reformulate the IoU and the dice coefficient differently.

$$IoU = \frac{TP}{TP + TN + FN}$$

$$F1 = \frac{2TP}{2TP + TN + FN}$$

With TP the pixels contained both in the ground truth box and the box predicted by the threshold (P the class of pixels with value 1), FN the pixels contained only in the ground truth box (N the class of pixels with value 0), and F the pixels contained in the predicted box but not in the ground truth box. If we put $A = TP$ and $B = TP + FP + FN$ We therefore obtain :

$$IoU = \frac{A}{B}$$

$$F1 = \frac{2A}{A + B} = \frac{\frac{2A}{B}}{\frac{A+B}{B}} = \frac{2IoU}{IoU + 1}$$

If we consider the function :

$$f : [0, 1] \to \mathbb{R}$$

$$x \mapsto \frac{2x}{x + 1}$$

The function is strictly increasing on this interval. f is differentiable and

$$f' : x \mapsto \frac{2}{(x + 1)^2}$$

Note that : $\forall x \in [0, 1] : \ f'(x) \geq 0$

We deduce that when the dice coefficient increases the IoU increases and it is the same in the other direction, so the two metrics are similar from a numerical point of view, which leads us to

take only one for the predictions.

5 Classification metrics

In this part we will discuss classification metrics in more detail, giving the formulas for each metric and the purpose behind it.

5.1 Average Drop [6]

As said previously in the state of the art, this metric measures the loss of predictive power when only the masked region is used, and by the predictive power is meant the measured class probability. More explicitly let p_i^c be the predicted probability for the class c of an image i and o_c^i the same thing but for the masked version of the image, this metric is calculated as follows :

$$AD = \frac{1}{n} \sum_{i=1}^{n} \frac{|p_i^c - o_c^i|}{p_c^i}$$

the intuition behind this metric is, if our model interprets well, then giving the masked version of the image that contains the solution as input, we should have results close to those found by the image original as input. So the smaller the value calculated by this metric, the more the model uses the right features for its interpretation, lower is better.

5.2 Average Increase [8]

Similarly, the average increase consists of calculating the number of images in the set of test images that have a probability p_i^c less than o_i^c, in other words we seek the percentage of images where the network has correctly interpreted .

$$AI = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{p_i^c \leq o_i^c}$$

The greater the value of this metric, the more the network uses the right features to make its decision, higher is better.

5.3 Insertion

The insertion metric [13] measures the increase in probability for a class c as more and more pixels are introduced, There are several ways of adding pixels from an image, it can be introduced to a constant canvas or by starting with a highly blurred image and gradually unblurring regions.

5.4 Deletion

The deletion metric [7] measure the decrease in the probability of class c when removing pixels, again there different ways of removing pixels, we can set the pixel values to zero or any other constant gray value, blurring the pixels or even cropping out a tight bounding box.

6 Localization metrics

As for classification metrics, we present localization metrics that we found interesting. The advantage of localization metrics is that they are very fast to calculate because we only need the heat map produced to evaluate the interpretation, while classification metrics need to make transformations on the images input and watch the effect it produces.

6.1 Localization error (LE)[9]

We can see the localization error as the overlap error between the bounding box generated from the heat map and the ground truth box. For a set of images I we will build from this set and a threshold τ, several bounding boxes. then we will calculate the average of the errors for each image.

$$LE = \frac{1}{n}\sum_{i=1}^{n}(1 - IoU(B_{t,i}, B_{p,i}))$$

$B_{p,i}$ and B_{t_i} correspond respectively to the box generated using the threshold τ and to the ground truth box for an image i in the image set I. n is the size of the set I.

This method seems natural and simple to implement the problem is that it gives us not much information about the saliency map. In addition, it does not seek an optimal τ which will guarantee the best results every time, which makes it less stable because it can favor images and penalize others according to the threshold.

There are several variants of the localization error, such as the localization-Recall-Precision (LRP) Error [14] or the official metric in [9] which do same thing as the localization metric but verify if the predicted class whether correct or not.

6.2 Pixel wise F1 score [15]

Another way to evaluate consists in considering each pixel of the binarized image as a data with a label of 0 or 1. This metric gives a rate of good prediction for each pixel, of a given image. But before giving the formula of the F-measure, let's start by explaining how precision and recall are calculated.

Precision is the number of well-classified pixels compared to the predicted mask pixel. This how we calculate the precision :

$$P = \frac{\sum_p \mathbb{1}_{p \in B_t \cap B(\tau)}}{|B(\tau)|}$$

With $B(\tau)$ the box generated with the threshold τ, B_t the ground truth box, p is a pixel in the image and $\mathbb{1}$ the indicator function.

Recall is the number of pixels well classified according to the references or the number of pixels whose ground truth box.

$$R = \frac{\sum_p \mathbb{1}_{p \in B_t \cap B(\tau)}}{|B_t|}$$

The F measure is a combination of precision and recall. It allows us to have an overall idea of the number of well-classified pixels.

$$F1 = \frac{2PR}{P+R}$$

As for the localization error this metric strongly depends on the chosen τ threshold, depending on the mask the values of the metric can change drastically.

6.3 Standard pointing game (SP)[10]

The standard pointing game checks for each heat map if the pixel with maximum saliency is in the ground truth box.

$$SP = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{p^* \in B_t}$$

Like the localization error, this metric is easy to implement but it is not very precise, especially if the pixel with the largest component does not contain the pixel with the maximum saliency which brings us to the next metric.

6.4 Energy based pointing game (EP)[7]

Instead of looking only if the pixel whose saliency is maximum is in the ground truth box, we will calculate the proportion of saliency energy in the ground truth box. We want to know how much energy of the saliency map falls into the target object bounding box.

$$EP = \frac{\sum_{p \in B_t} S_p^c}{\sum_{p \in B_t} S_p^c + \sum_{p \notin B_t} S_p^c}$$

S_p^c is the value of the pixel p in the saliency map S. After that we need to calculate the average of this metric on all the test set of images. It's better compared to the "standard pointing game", we have a more telling proportion but this metric does not work very well every time, once again this metric does not take into account the size of the box. We will disadvantage the images in the box of the ground truth is small and give the advantage to the image whose box is large because it will have more chance to contain more pixels.

6.5 Box accuracy (BA) [15]

This metric uses two thresholds, one for the binary mask and another for the IoU. The goal is to know if there is a mask t such that the IoU of the box generated by the mask and that of the ground truth have an overlap rate greater than σ. In a way it's a grid search to find the mask that maximizes the IoU.

$$BA(\tau, \sigma) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{IoU(B(\tau), B_t \geq \sigma)}$$

Although this metric seeks the best results to guarantee fairness between the images evaluated, it takes a long time to implement because you have to search for several thresholds until

you find one that suits.

6.6 Saliency metric [16]

Piotr Dabkowski and Yarin Gal have proposed a new saliency metric which tries to combine salience and location of areas of interest, in general the salience map of a chat will not contain the whole chat, whereas the ground truth box will contain the whole chat and it will contain no-salient details. We therefore want a metric that takes this remark into account and chooses the box that contains only the most important features in the image.

So we want the classifier to always be able to recognize the object from the salience map produced and that the predicted box is the smallest possible. In order to have a fixed box, instead of masking, we will crop the image instead of thresholding. We are going to look for the tightest box that contains all the salient region, then we are going to give this rectangular region to the classifier to directly check if it is able to recognize the requested class. This is how they defined their metrics :

$$SM = log(max(0.05, \frac{|B_p|}{hw}) - log(p^c)$$

Where $|B_p|$ is the size of the predicted box, h and w the dimensions of the image and p^c the probability of the class of interest. We take the max between 0.05 and the size of the box relative to the image to avoid having very small values. Finally, we emphasize that the more the value calculated by the metric is not bounded, the smaller the values are, the better the results will be.

6.7 Mean Absolute Error (MAE) [17]

Finally we present a last metric, the MAE seeks to calculate the average distance between the solution and the prediction. It is defined as below :

$$MAE = \frac{1}{hw} \sum_{i=1}^{h} \sum_{j=1}^{w} |S(i,j) - G(i,j)|$$

h, w correspond to the dimension of the image, S(i,j) the salience of the pixel at position (i,j) of the image, we have $0 \geq S(i,j) \leq 1$ and G(i ,j) the value of the pixel value (i,j) in the ground truth, we have $G(i,j) \in 0, 1$.

7 Masked based localization metrics

In this part we will introduce you to the mask-based metrics we talked about in the introduction, we will explain each metric, then we will use these metrics to evaluate the results of a Grad-CAM model. These metrics will look a bit like the metrics we presented in the state of art the only difference is that we will use a ground truth mask instead of a ground truth box.

We will use a data set that contains labels in the form of a mask instead of a box, we will follow the same protocol that we presented in the "state of art" part until the part where we find the largest component in the binary mask, this component will represent the predicted mask.

7.1 Masked localization error

Let's start with the most natural idea and the easiest to implement. The mask-based localization error is used to calculate the rate of overlap between two masks.

$$MLE = 1 - IoU(M_t, M_p)$$

M_t represents the ground truth mask, while M_p represents the mask predicted by the threshold τ.

This metric remains less precise, we do not have the energy contained in the overlapping of the masks as information. Which brings us to our next metric.

7.2 Energy masked pointing game (EMPG)

As for "energy pointing game", we will calculate the general interest that the CNN grants to the ground truth mask. For this we multiply the binary mask of the ground truth by the saliency map and sum the value of the pixels and we divide the whole by the sum of the values of the pixels in the saliency map.

$$EMPG = \frac{\sum_{p \in M_t} S_p^c}{\sum_{p \in M_t} S_p^c + \sum_{p \notin M_t} S_p^c}$$

This metric does not involve the threshold when it evaluates, which is good in itself because we do not need to look for the optimal threshold.

As for the masked localization error and the masked energy pointing game we can do the same thing for the other metrics presented above, it is enough just to replace the box with the mask, We will therefore present other ways of evaluating which have not been seen in the papers.

7.3 Warping error (WE) [18]

Another approach consists in counting the number of transformations that must be done to transform the predicted mask into the ground truth mask. It penalize the topological error between two masks, for our case it will compare the topological differences between the ground truth mask and the predicted mask. But we need to find first the best deformation of the predicted mask that's minimize the Manhattan distance between the two masks.

$$WE = min(\sum_{i=1}^{h} \sum_{j=1}^{w} |M_t(i,j) - M_p^*(i,j)|)$$

Where h and w are the dimension of the image, M_t and M_p are the ground truth mask and the predicted mask for a threshold τ, modified to be the closest deformation to the ground truth mask. generally to obtain such deformation it is necessary to make a decent of the gradient because it is difficult to have a deformation which best approximates the ground truth mask (global minimum), so we are satisfied with a reasonable approximation (local minimum).

To fully understand what is the topological error, let's look at the picture below.

The image (A) has a topological error because it contains 4 components instead of 2, while (B) retains the topology of the image, there are indeed two components. A topological error

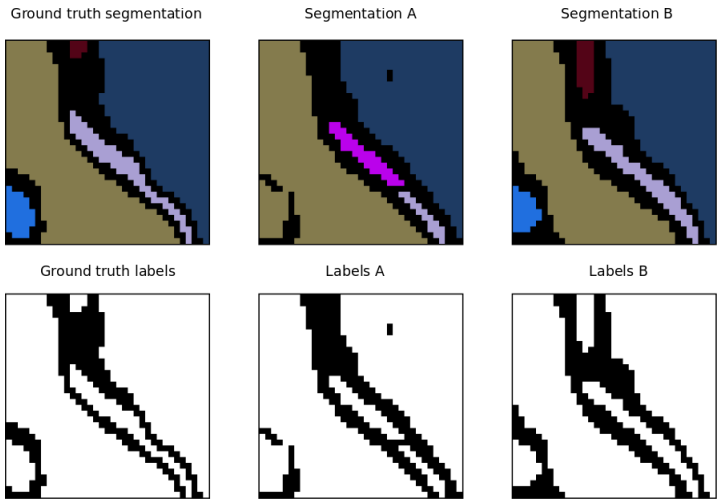

FIGURE 9 – https ://imagej.net/media/plugins/tws/warping-error-comparison.png

exists between two images if the first cannot be modified to obtain the second without tearing or merging the components.

8 Experiments

We will now present the result of our experiments. For this we use the Pet-Data set from the Oxford-III web site which contains images of cats and dogs, masks and annotations for each image, and we use a pre-trained "imagenet" convolutional neural network. The experiments consist in calculating the localization error, the energy pointing game and the F1 score for both based boxes and based on masks methods, The table below contain the average of each presented metric :

metrics	LE	MLE	EPG	MEPG	F1	MF1
averages	0.45	0.25	0.93	0.76	0.43	0.51

TABLE 1 – Average of each metric

We notice that for the localization error the box-based error is larger than the mask-based one, we explain this because the box-based metrics take more pixels even if they do not participate decisions (the geometric figures considered are always rectangles when this is not the case). Similarly for "energy pointing game" the boxes will always contain more salience than a smaller geometric figure). We notice that unlike the other two metrics, the F1 score of the mask-based method is close to that of the box-based one. Unfortunately we can't say anything with these results so we don't know if the mask-based metrics are more accurate and will they adopt the same behavior as the mask-based methods ? More experiments are needed to empirically answer these questions.

9 Conclusion

As a conclusion, CAM methods generate salience maps which indicate the zones of interest of a classifier. We use these heatmaps to explain the decision of our model, and there are several

methods to evaluate the quality of a heatmap (is the interpretation right or wrong), among these methods there are box-based methods which are used in a lot of literature and mask-based methods, these metrics have the same behavior but the mask-based one is stricter than the box-based one. If we had had more time, we could have done more experiments on a larger data-set with various metrics to have more representative results to answer the problem of our internship.

Références

[1] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[2] Tim Miller. Explanation in artificial intelligence : Insights from the social sciences. *Artificial intelligence*, 267 :1–38, 2019.

[3] Yu Zhang, Peter Tiňo, Aleš Leonardis, and Ke Tang. A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2021.

[4] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[5] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam : Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[6] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++ : Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018.

[7] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam : Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 24–25, 2020.

[8] Hyungsik Jung and Youngrock Oh. Towards better explanations of class activation mapping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1336–1344, 2021.

[9] Jason Phang, Jungkyu Park, and Krzysztof J Geras. Investigating and simplifying masking-based saliency methods for model interpretability. *arXiv preprint arXiv :2010.09750*, 2020.

[10] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10) :1084–1102, 2018.

[11] willem (https ://stats.stackexchange.com/users/159052/willem). F1/dice-score vs iou. Cross Validated. URL `https://stats.stackexchange.com/q/276144`. URL :https ://stats.stackexchange.com/q/276144 (version : 2017-11-13).

[12] Nico (https ://stats.stackexchange.com/users/124257/nico). F1/dice-score vs iou. Cross Validated. URL `https://stats.stackexchange.com/q/488098`. URL :https ://stats.stackexchange.com/q/488098 (version : 2020-09-18).

[13] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise : Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv :1806.07421*, 2018.

[14] Kemal Oksuz, Baris Can Cam, Emre Akbas, and Sinan Kalkan. Localization recall precision (lrp) : A new performance metric for object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 504–519, 2018.

[15] Junsuk Choe, Seong Joon Oh, Seungho Lee, Sanghyuk Chun, Zeynep Akata, and Hyunjung Shim. Evaluating weakly supervised object localization methods right. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3133–3142, 2020.

[16] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. *Advances in neural information processing systems*, 30, 2017.

[17] Shuhan Chen, Xiuli Tan, Ben Wang, and Xuelong Hu. Reverse attention for salient object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 234–250, 2018.

[18] Viren Jain, Benjamin Bollmann, Mark Richardson, Daniel R. Berger, Moritz N. Helmstaedter, Kevin L. Briggman, Winfried Denk, Jared B. Bowden, John M. Mendenhall, Wickliffe C. Abraham, Kristen M. Harris, Narayanan Kasthuri, Ken J. Hayworth, Richard Schalek, Juan Carlos Tapia, Jeff W. Lichtman, and H. Sebastian Seung. Boundary learning by optimization with topological constraints. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2488–2495, 2010. doi : 10.1109/CVPR.2010.5539950.

Annexe A. Titre de l'annexe

The code used to do the experiments : `https://github.com/AMoh22/Class_activation_map_metrics.git`