Multi Layered Feature Explanation Method for Convolutional Neural Networks ⋆

Luca Bourroux, Jenny Benois-Pineau, Romain Bourqui, Romain Giot

Multi Layered Feature Explanation Method for Convolutional Neural Networks*

Luca Bourroux, Jenny Benois-Pineau[0000−0003−0659−8894], Romain Bourqui[0000−0002−1847−2589], and Romain Giot[0000−0002−0638−7504]

Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR5800, F-33400 Talence, France

Abstract. The most popular methods for Artificial Intelligence such as Deep Neural Networks are, for the vast majority, considered black boxes. It is necessary to explain their decisions to understand the input data which influence most the result.
Methods presented in this paper aim at an explanation in image classification tasks: which data in the input are the most important for the result. We further extend the Feature Explanation Method (FEM) from our previous work, transforming it into a multi-layered FEM (MLFEM). The evaluation of the method is designed by comparison of explanation maps with human Gaze Fixation Density maps (GFDM). We show that proposed MLFEM outperforms FEM and popular DNN explanation methods in terms of classical comparison metrics with GFDM.

Keywords: XAI · Features attribution · ResNet · Gaze Fixation Density Maps

1 Introduction and related work

Deep learning (DL) approaches have become indispensable in data analysis and classification. Although the results of DL models have been exemplary, they lack transparency, which prevents a wide use of them in critical applications such as medical image analysis or security.

The need for explanations of Deep Neural Networks (DNN) decisions has led to an active research in eXplainable Artificial Intelligence (XAI). In the field of pattern recognition for images and videos, explanation of a DNN's decision consists in identifying the set of input pixels that have contributed the most into the decision [1]. A famous example of decisions on a wrong data is given by Ribeiro *et al.* [2]: here, a trained classifier wrongly used the presence of snow as the distinguishing feature between the *"Wolf"* and *"Husky"* classes. One of the ways to automatically verify if the DL classifier builds its decision on the adequate pattern in the input data is to compare the set of "important" pixels with human observations of visual content; we use this approach in this paper.

Ayyar *et al.* [1] analyzed the variety of explanations methods highlighting important patterns from the input image. Following their proposed taxonomy,

* Supported by LaBRI.

we can identify "black-box" and "white-box" families of explanation methods. The "black-box" methods remain model agnostic and are applicable to any classifier, as they identify important pixels in images by masking different parts of the input and tracking induced decisions (*e.g.*, LIME[2] method). "White-box" methods, on the contrary, use the internal architecture of DNNs and can be subdivided into several groups. The first one concerns the methods based on linearization of the Deep-CNN. One of the first was the so-called Deconvolution Network (DeconvNet) proposed by Zeiler *et al.* [3]. The principle here was to build the mapping of the output score to the input space using reverse filters or "deconvolution" and thus identifying the important pixels. The methods based on gradient back-propagation such as a popular GradCam [4] or its further improved versions, (*e.g.*, smoothgrad [5] or integrated gradients [6]) proceed by propagation of gradients from the last layers to the input with regard to the changed input. The important input information is located where the gradients are strong. The *Layered Relevance Propagation method* (LRP) [7] is also based on the same idea of back propagation, but without the need of gradient computations. Here, the relevance of neurons from the last decision layer through receptive fields in previous layers using the principle of conservation of the relevance at each layer allows identifying important input neurons-pixels. *Feature Explanation Method* (FEM) [8] (Section 2.1) is also based on backpropagation, but it is not linear in the sense that with the help of statistical filtering of the features of the last convolutional layer it identifies the most important ones. The various features maps are usually depicted using a heatmap overlaying the input image, explaining important regions in the input to the user.

Propagating information through subsequent layers, the Deep NNs loose high resolution information due to the cascaded convolutions and subsampling. Hence, it is logical to explore the DNN classifier a bit more, preserving important details in each conv layer which finally bring the classification decision. We assert that the fusion of explanations from several layers is a key to improve the final explanation. For this reason, we propose an extension of FEM, the Multi-Layered FEM (MLFEM), that relies on information fusion on feature importance from different layers. We study different strategies of fusion and benchmark them against Gaze Fixation Density Maps resulting from psycho-visual experiments on image databases.

The reminder of the paper is organized as follows. Section 2 presents our contribution - MLFEM, Section 3 gives the evaluation methodology, while results are reported in Section 4. Conclusion and perspectives are drown in section 5.

2 Multi Layered Feature Explanation Method

Our method MLFEM is built upon FEM method [8]. The latter relies on the analysis of activations at the *last* conv layer of a CNN classifier. As each layer of a CNN embeds information at a different scale, we assume that computing FEM at several layers and fusing them would improve the quality of the feature

attribution; this is the main idea of MLFEM. In the following, we briefly review FEM and describe the adaptations for MLFEM.

2.1 Reminder of Feature Explanation Method (FEM)

Feature Explanation Method (FEM) [8] is a recent algorithm used to produce an explanation map of the decision of a CNN. In opposite to other methods of the literature, it is class agnostic and does not need to provide a class of interest. FEM makes two hypotheses. The first is that strong features at the last convolutional layer will contribute the most in the final decision of the CNN in a classification task when pushed through fully connected layers. The second hypothesis will help us to select strong features. It assumes that the features in each feature map follow a Gaussian Distribution; this is a simplification hypothesis, but in case of large feature maps it could hold. At the last layer of convolution the strongest features are the representations of the most relevant regions in the input image. This comes from the interpretation of the "convolutional" part of a deep CNN as of a multiscaled pyramid with filtering, non-linear input signal transformations and subsampling [9].

By analyzing only the last layer of the CNN part of the model, one can get input pixels which contributed the most into the final decision. As FEM uses activations of the last layer of the CNN *after* the non-linearity (most likely ReLU) only *positive* features will be picked up. As FEM is class agnostic, it is not a problem if it emphasizes on some high activations that are negatively weighted in the next layer. Indeed the features remain important for the overall classification wether they vote for or against a specific class.

The last convolutional layer produces activations of size $(W \times H)$ for D feature maps f_i. D binary maps, $b_{i,i=1\ldots D}$ corresponding to each of the D feature maps f_i are computed by selecting their strongest features: $f_i(x, y) \geq \mu_i + k\sigma_i$. Mean μ_i and standard deviation σ_i are individually estimated for each feature map f_i. In the same manner as the most important features are selected in each feature map f_i, the contribution of each map into decision is also weighted by a map-importance weight corresponding to μ_i. This will give us a saliency map $s = \sum b_i \mu_i$ of dimension $(W \times H)$ that is then upscaled to the resolution of the input image by linear interpolation. A min-max normalization is finally used to bring the domain from $\mathbb{R}^+$ to $[0, 1]$ and obtain the final normalized map of feature importance S.

2.2 Principles of Multi Layered FEM (MLFEM)

In fact, FEM as presented previously can be applied on any layer of a CNN. We can merely pretend to truncate the model at a particular layer and see that FEM would work as is. The application of FEM on a CNN consisting of L convolutional layers will yield L different feature importance maps. As all importance maps are interpolated in FEM method we finally have L maps of the input resolution. The information provided by the maps is layer-dependent and it is interesting to fuse them. Now the question is how to obtain a single

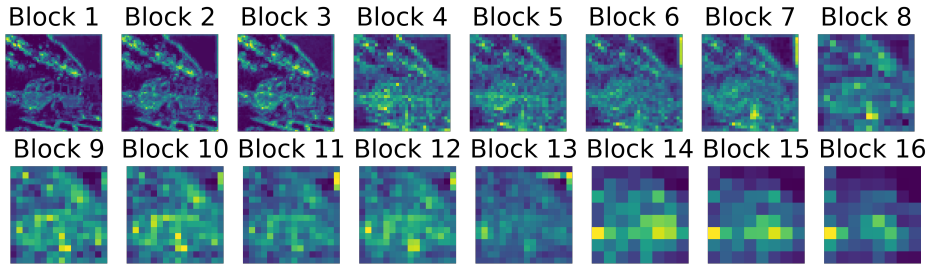


Fig. 1: FEM applied on every convolutional block of a typical ResNet50 architecture. Resolution is higher for the first layers.

heat-map for the input highlighting the pixels which have contributed into the Netwok decision the most.

Let M be our convolutional neural network, with $l = 1, ..., L$ convolutional layers. Let us denote F_l a feature tensor obtained at each convolutional layer after the positive non-linearity (ReLU).

Let us denote by $H(F_l)$ the operator which implements FEM method yielding to a normalized importance map S_l of features F_l. The multi layered FEM pixel importance map is obtained by fusing all the importance maps S_l with a fusion operator $\bigoplus$: $S = \bigoplus_{l=1}^{L} S_l$.

The combination of L different maps can be done in a recursive manner. For each intermediary map S_l we will construct S as the combination of S and the current intermediary map S_l. We will then move along for each $l < L$. An alternative is to devise a fusion operator that takes a variable number of arguments to produce a single final feature importance map.

Intuition behind Multi-layerd FEM The reasoning for applying multiple times the same explanation method at different network's layers is the following. The network passes the input image through multiple layers of convolution. Convolution layers produce results that are position invariant. They are meant to pick up on spatially-local feature. With each step deeper in the network, the convolutional layer picks up on a more and more abstract concepts (see Figure 1). The very first layers are generally performing edge detection while the later ones extract abstract concept like "face", "car" etc..

So different information is available at different points in the network. By combining the different activation maps at different points in the network, we can reconstruct a heatmap that takes advantage of all this scattered information.

2.3 Fusion operators

We can apply quite a number of usual operators in data fusion. In our present work, we have appealed to the algebraic fusion operators, and to the fusion by a convolutional neural network trained with regard to the ground truth obtained from human observers of the content.

Algebraic fusion operators applied in our work are presented below. They are applied individually to the element of the maps.

- The max operator $max(a, b)$ is the result of the max operation applied element wise to a and b. $max_{u,v}(a, b) = max(a_{u,v}, b_{u,v})$
- The weighted addition $add(a, b)$ is the result of the addition of a and b given a factor α. $add_{u,v}(a, b) = \alpha \cdot a_{u,v} + (1 - \alpha) \cdot b_{u,v}$
- The *top* operator. It is defined in relation to the *add* operator, taking only the top 50% features of b. $b'_i = b_i$ if $(b_i > \mu(b))$, 0 otherwise, top(a, b) = add(a, b')
- The fem operator. We can produce the same result as the FEM method would by using this fusion operator. $fem(a, b) = b$ it is also a special case of the *add* operator with $\alpha = 0$

The maximum is commutative: $max(a, b) = max(b, a)$, but the *add* and so the *top* operators are not: $add(a, b) \neq add(b, a)$, $top(a, b) \neq top(b, a)$. The *add* operation in fact constructs a geometric sequence. $\Sigma\alpha(1-\alpha)^{l-1}S_l$. The normalization operator can also be applied, either at the end of the fusions or interleaved between each binary operator.

By the fact that *add* and *top* are not commutative they take advantage of the structure of data, namely, of the fact that the first operand of the operator is the cumulated map in the recursive fusion approach and can be more or less taken into account regulated by the parameter α.

Fusion by a convolutional neural network The idea here consists in training a light CNN m which input is the set of feature importance maps S_l from all layers of the CNN model M to be explained, interpolated to the resolution of input images of M. The training of m is fulfilled with regard to the ground truth expressing human perception of the visual content in the classification task. This perception is measured by Gaze Fixation Density Maps (GFDM) obtained in a psycho-visual experiment when human subjects observe images to classify them. We refer the reader to [10] for a detailed explanation of such an experiment. Human gaze fixations from a number of observers are recorded for each image by an eye-tracking device.

Then on each fixation (u, v) in the image plane, a 2D-Gaussian surface $N_{(u,v,\Sigma)}$ is centered with the mean vector $\mu = (u, v)^T$ and a diagonal covariance matrix Σ with equal σ^2 values on the principal diagonal. The scale parameter σ is defined from the geometry of the experiment to represent the projection of the fovea, into the image plane. Summing up and normalizing multi-Gaussian surface from different observers for the same image, its GFDM G is obtained. An example of such maps on the dataset from [10] is illustrated in figure 2.

We call this CNN-based fusion operator NET. As a light fusion CNN m we use a simple architecture. The input tensor consisting of intermediate importance maps S_l is pushed through three successive convolution layers with pooling. They have the total effect of pooling the input tensor by a factor of 4 and multiplying the depth by a factor of 8. Lastly, a weighted sum is computed to output a final

Fig. 2: Example of GFDMs on Mexculture database[10]

predicted 2D map. The loss function is the Euclidean loss, which is the mean square error between G and the application of NET operator to the input set of of importance maps S_l. The fusion network m is trained on GFDMs of a training set of a given dataset. We present our datasets in section 4.

2.4 Implementation of MLFEM on CNN classifiers

The MLFEM is a white-box method and can be applied to any architecture of a CNN in visual classification tasks. Here we present the implementation of MLFEM for ResNet50 [11] architecture.

ResNet50 contains 16 residual blocks; as a block is the natural unit of choice to apply FEM, we apply it to the output of each of them, after the activation function. This gives us 16 different applications of FEM to fuse. The goal is to maximize the different semantic meaning that one can extract: for ResNet50 we take advantage of the structure of the network.

ResNet50 used in our experiments is trained with the Adam optimizer [12], with a binary cross entropy for binary classification tasks and a categorical cross-entropy loss for multi-label classification tasks.

3 Evaluation of MLFEM explanations

Methodology. In [1] and [13] the methodology of evaluation of explanation methods was proposed. It consists in the comparison of the pixel importance maps obtained by the network sensing with the maps expressing human perception of the same visual content. This perception is expressed by gaze fixation density maps we have presented in section 2.3. Today, this methodology is possible thanks to public databases with the recorded gaze fixation of observers, like [10,14,15] and we will apply it to our MLFEM method with different fusion operators and compare with different explanation methods which generate pixel importance maps as MLFEM does.

Evaluation Metrics Evaluating the relevance of pixel importance explanation maps is an open problem. There is no widely agreed upon metrics for assessing their quality. We propose to employ metrics widely used in psychovisual community for comparison of saliency maps[16]: the Pearson Correlation Coefficient

and the Similarity metric. Pearson Correlation Coefficient(PCC) is defined as:

$$corr(x,y) = \frac{\sum_u^W \sum_v^H (x_{u,v} - \overline{x})(y_{u,v} - \overline{y})}{\sqrt{\sum_u^W \sum_v^H (x_{u,v} - \overline{x})^2} \sqrt{\sum_u^W \sum_v^H (y_{u,v} - \overline{y})^2}}$$

with $x_{u,v}$ being the value of the pixel at position (u, v) in one saliency map and $y_{u,v}$ in another.

The similarity metric is defined as such:

$$sim(x,y) = \sum min(x_{u,v}, y_{u,v})$$

Design of experiments. To evaluate the proposed MLFEM method, we will perform three kinds of experiments: i) overall method comparison, ii) dependence on correct or wrong classification results, iii) sensitivity to clutter in the image.

Overall method comparison. We will compare pixel importance maps generated by MLFEM with different fusion methods and the reference methods such as FEM and GradCam [4] which remains the most popular in explanation methods generating pixel importance maps.

Dependence on correct/wrong prediction. We divide the test dataset into images that were correctly categorized by the neural network M and those that were not. We will analyze if a drop in correlation with the ground trouth GFDMs is observed.

In the case of a drop, we can deduce that the convolutional part of the network did not pick up on the relevant features of the image. The fully connected part of the network will then not have the correct information in feature space to classify the image.

If the metrics do not change between correctly and wrongly classified images, we can presume that the fully connected part of M, given supposedly correct feature space information, was not able to classify the images. We can then add more fully connected layers to help the network categorize the inputs.

Sensitivity to clutter. In case the image is cluttered, the GFDMs are dispersed as human attention is attracted by multiple singularities/objects in the image. It is reasonable to expect that the CNN allocates more importance to a strongest relevant region in the image. In this case, the similarity metrics between our explanation map and GFDMs will be lower than for images with low clutter effects.

4 Experiments and Results

4.1 Datasets

We have chosen three different datasets to work on. The first one is MexCulture [10]. It is composed of 284 images from four classes supplied with GFDMs. It is a subset of 12000 images of the Mexican architectural style: Modern, Pre-Hispanic and Colonial, there is also a rejection class. The dataset contains 2000

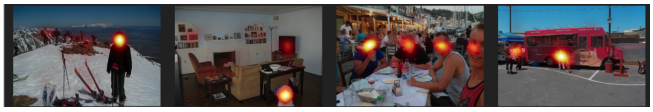


Fig. 3: Sample of the data in Salicon database (with GFDM overlaid)

images for each category for training and 2000 other images for validation. The GFDM was constructed by instructing the subject to categorize the presented images. We have illustrated the GFDMs in figure 2.

The second one is Salicon [14]. It is composed of 15000 images of different (up to 80) categories, 10000 of them are supplied with GFDMs It is common to have multiple categories present for each image, hence these images are more cluttered than those from MexCulture dataset. To construct the GFDMs, the subjects were free to "look around" the image and expressed their attention by mouse pointing. We illustrate the GFMDs from this dataset in figure 3.

Finally, we use Cat2000 [15] as a third dataset. It is composed of 4000 images, 2000 with GFDM, each divided into 20 equally populated categories. The GFDMs were obtained from gaze fixations and the images are less cluttered.

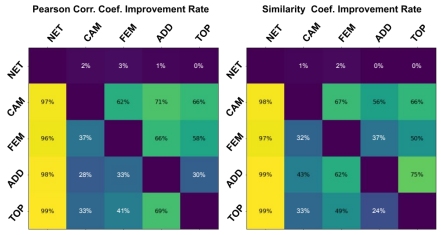
4.2 Results

Overall method comparison. We compute two similarity metrics PCC and *sim* for each classified image of each dataset for every method (FEM, GradCam noted CAM and MLFEM with NET, ADD and TOP variantes) with regard to the GFDMs. To compare the methods between them, we compute a 2×2 matrix, comparing the number of times that a method m_a was better at explaining classification than a method m_b in terms of higher value of each metric. This will give us 6 matrices, 2 for each dataset, one for the comparison using PCC and another for the *sim* metric, see figures 4a, 4b, 4c.

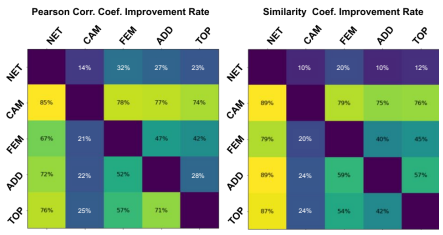
In figures 5a, 5b distribution of the metrics in Salicon and Cat2000 datasets is plotted for every method compared to the GFDMs.

We can see here in figures 4a and 5a that the proposed MLFEM method is better suited for the explanation of ResNet 50 on the Salicon dataset. We achieve a mean correlation coefficient of, 0.70 whereas the GradCam method only achieves 0.37. We can also see that without resorting to learned fusion operator, FEM and ADD/TOP are better with, respectively, $0.38, 0.43, 0.41$ values of PCC. The *sim* metric behavior is the same. The trained NET fusion operator is the best in terms of both metrics. In Mexculture dataset, the NET operator is at least as good as other operators, see figure 4c. For Cat2000 dataset the conclusion is the same as for Salicon, as illustrated in 4b and 5b.

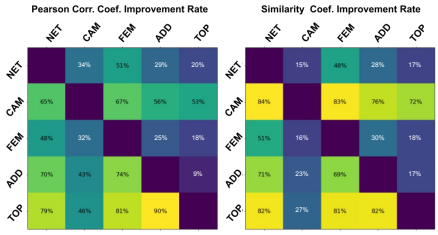
Dependence on correct/wrong prediction. For Salicon, we are in a multi-label classification task with objects of several classes in the same image. We count the number of times when every class present in an image is detected by the network: *i.e.*, the corresponding output neuron has an activation value larger than 0.5.



(a) Salicon

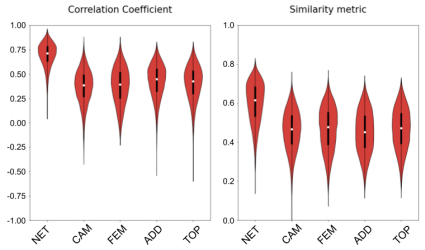


(b) Cat2000

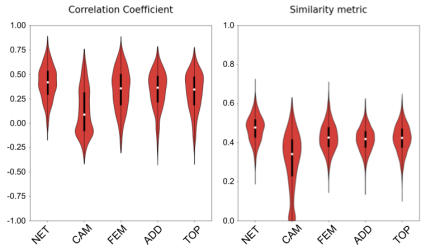


(c) MexCulture

Fig. 4: Number of images with a better explanation by method m_a - column than with method m_b - line -on the Salicon, Cat2000 and MexCulture datasets.



(a) Salicon

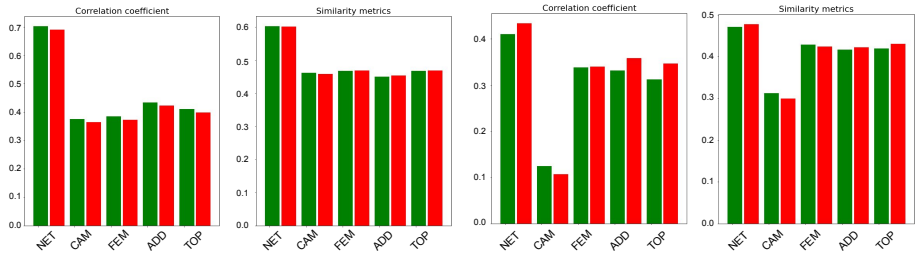


(b) Cat2000

Fig. 5: Distribution of the different metrics for each explanation method for the Salicon and Cat2000 dataset.

as a correct prediction. When at least one class is not correctly predicted, this image is considered wrongly classified.

We do not see any significant effect of the correct/wrong categorization of the image on the quality of the explanation map (Figures 6a and 6b diplay the average PCC and Similarity metrics for both correctly and wrongly classifed samples). For Salicon dataset (Figure 6a), we have a small drop in the quality of explanation for miss-classified images (in red), a 3% drop in the PCC and no change in the *sim* metric. For Cat2000, figure 6b, a small increase of the quality



(a) Salicon (b) Cat2000

Fig. 6: Metrics for correctly classified (green) and wrongly classified images (red) for the Salicon and Cat2000 dataset with NET, CAM, FEM, ADD, TOP.

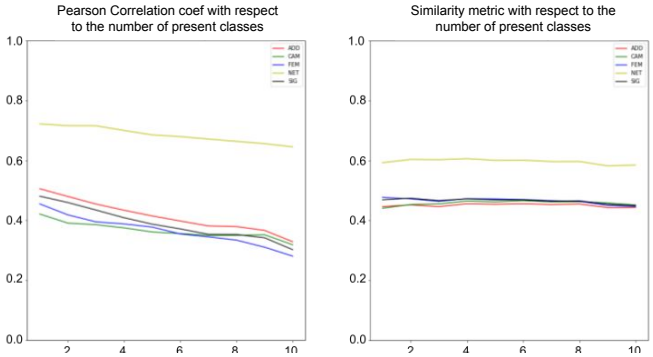


Fig. 7: The evolution of similarity metrics with regard to the number of classes present in the image to explain on the SALICON dataset.

of explanation for miss-classified images is observed. The PCC is 4% better, but there is practically no change for the similarity metric.

Sensitivity to clutter. This experiment is conducted on Salicon dataset with different classes of objects present in the same image. We divide images into ten categories, with ith category containing images that have i different classes present in it. Then we can plot the mean of the different metrics as a function of the number of classes appearing in an image.

We can see in Figure 7 that our hypothesis holds: the quality of our explanation drops when the number of classes present in an image increases. However, even in high clutter situation, our proposed method with NET fusion gives the best scores. The NET method only loses 11% of its performance measured by PCC, where the FEM method loses about 33%.

The similarity metric, however, shows a lower dependence in regard to the number of classes present in an input image. The implication of this merits further research in future works.

5 Conclusion

In this work, we have extended the method FEM for the explanation of decisions of CNN classifiers by introducing a Multi-Layered strategy: MLFEM. We showed its performance on the ResNet as the latter is nowadays the most efficient CNN classifier. Nevertheless, the method remains generic and applicable to any CNN whether it is residual or not.

We studied different fusion strategies of individual importance maps from each CNN layer. The evaluation has been performed accordingly to the evaluation method designed by us similarly with [13] which consists in the comparison of explanation maps with gaze fixation density maps (GFDM) of human observers. In terms of comparison metrics of explanation maps and GFDM, MLFEM achieves better performance than the similar state-of-the-art method GradCam and the original FEM. Over GradCam, we got an improvement of 89% for PCC on Salicon and 241% on Cat2000; the improvement of similarity metric is of 30% on Salicon and 51% on Cat2000. We got an improvement over FEM of 84% for PCC on Salicon and 20% on Cat2000; for similarity metric, the improvement is of 27% on Salicon and 9% on Cat2000.

We note that amongst the proposed fusion strategies, the "learnt" fusion operator achieves the best results according to the considered comparison metrics. Hence, the proposed MLFEM method better explains the decisions of the CNN classifier ResNet with regard to the human perception of the content in the classification tasks.

In the future works, it may be interesting to apply and adapt MLFEM to newly appeared transformer networks and to use it with another kind of data than images. In such a case, we will face the problem of definition of the ground truth for the evaluation methodology proposed. Hence, the proposed method opens multiple research questions which have to be addressed in the future.

References

1. M.P. Ayyar, J. Benois-Pineau, and A. Zemmari. White box methods for explanations of convolutional neural networks in image classification tasks, 2021.
2. 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*, pages 1135–1144, 2016.
3. M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Proceedings of European Conference on Computer Vision*, pages 818–833. Springer, 2014.
4. 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. *International Journal of Computer Vision*, 128(2):336–359, 10 2019.
5. D. Smilkov, N. Thorat, B. Kim, F. B. Viégas, and M. Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825:1–10, 2017.

6. M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. In *Proceedings of International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.
7. S. Bach, A. Binder, G. Montavon, F. Klauschen, KR. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10, 2015.
8. K. Ahmed Asif Fuad, PE. Martin, R. Giot, R. Bourqui, J. Benois-Pineau, and A. Zemmari. Features Understanding in 3D CNNs for Actions Recognition in Video. In *Tenth International Conference on Image Processing Theory, Tools and Applications, IPTA 2020*, Paris, France, 10 2020.
9. A. Zemmari J. Benois-Pineau. *Deep Learning in Mining of Visual Content*. Springer, Cham, 2020.
10. A. M. Obeso, J. Benois-Pineau, M. S. García-Vázquez, and A. A. Ramírez-Acosta. Visual vs internal attention mechanisms in deep neural networks for image classification and object detection. *Pattern Recognit.*, 123:108411, 2022.
11. F. Rousseau, L. Drumetz, and R. Fablet. Residual networks as flows of diffeomorphisms. *Journal of Mathematical Imaging and Vision*, 62, 04 2020.
12. Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
13. G. Jouis, H. Mouchère, F. Picarougne, and A. Hardouin. Anchors vs attention: Comparing XAI on a real-life use case. In *ICPR Workshops (3)*, volume 12663 of *Lecture Notes in Computer Science*, pages 219–227. Springer, 2020.
14. Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6 2015.
15. Ali Borji and Laurent Itti. CAT2000: A large scale fixation dataset for boosting saliency research. *CoRR*, abs/1505.03581, 2015.
16. O. Le Meur and T. Baccino. Methods for comparing scanpaths and saliency maps: Strengths and weaknesses. *Behavior research methods*, 07 2012.