

Explanations in terms of Hierarchically organised Middle Level Features

Andrea Apicella^{1,2,4}, Salvatore Giugliano^{1,2,4}, Francesco Isgrò^{1,2,4} and Roberto Prevete^{1,2,3,4}

¹Laboratory of Augmented Reality for Health Monitoring (ARHeMLab), Università degli Studi di Napoli Federico II, Naples, Italy

²Laboratory of Artificial Intelligence, Privacy & Applications (AIPA Lab), Università degli Studi di Napoli Federico II, Naples, Italy

³Interdepartmental Center for Research on Management and Innovation in Healthcare (CIRMIS), Università degli Studi di Napoli Federico II, Naples, Italy

⁴Department of Electrical Engineering and Information Technology, Università degli Studi di Napoli Federico II, Naples, Italy

Abstract

The rapidly growing research area of eXplainable Artificial Intelligence (XAI) focuses on making Machine Learning systems' decisions more transparent and humanly understandable. One of the most successful XAI strategies is to provide explanations in terms of visualisations and, more specifically, low-level input features such as relevance scores or heat maps of the input, like sensitivity analysis or layer-wise relevance propagation methods. The main problem with such methods is that starting from the relevance of low-level features, the human user needs to identify the overall input properties that are salient. Thus, a current line of XAI research attempts to alleviate this weakness of low-level approaches, constructing explanations in terms of input features that represent more salient and understandable input properties for a user, which we call here Middle-Level input Features (MLF). In addition, another interesting and very recent approach is that of considering hierarchically organised explanations. Thus, in this paper, we investigate the possibility to combine both MLFs and hierarchical organisations. The potential advantages of providing explanations in terms of hierarchically organised MLFs are grounded on the possibility of exhibiting explanations to a different granularity of MLFs interacting with each other. We experimentally tested our approach on *300 Birds Species* and *Cars* dataset. The results seem encouraging.

Keywords

XAI, Explainable AI, Hierarchical, middle-level, Interpretable models

1. Introduction

A large part of current successfully Machine Learning(ML) techniques can be considered as black-box systems insofar as they give responses whose relationships with the input are often challenging to understand. As ML systems are frequently being used in more and more domains and, so, by a more varied audience, there is the need of making them understandable and trusting

XAI.it 2021 - Italian Workshop on Explainable Artificial Intelligence


✉ andrea.apicella@unina.it (A. Apicella)

🆔 0000-0002-5391-168X (A. Apicella); 0000-0002-1791-6416 (S. Giugliano); 0000-0001-9342-5291 (F. Isgrò);

0000-0002-3804-1719 (R. Prevete)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

to general users [1, 2], leaving unaltered, or even improving, their performance. The rapidly growing research area of eXplainable Artificial Intelligence (XAI) is focused on this challenge. In the XAI context many approaches have been proposed to overcome the opaqueness of ML systems [3, 4, 5, 2, 6]. We note that in the literature, one of the most successful strategies is to provide explanations in terms of “visualisations” [1, 7], and, more specifically, in terms of low-level input features such as relevance scores or heat maps of the input, like sensitivity analysis [8] or Layer-wise Relevance Propagation (LRP) [3] methods. For example, LRP associates a relevance value to each input element (to each pixel in case of images) to explain the ML model answer.

The main problem with such methods is that human users are left with a significant interpretive burden. Starting from the relevance of each low-level feature, the human user needs to identify the overall input properties that are perceptually and cognitively salient to him [9, 6]. Thus, there is a current line of XAI research that attempts to alleviate this weakness of low-level approaches and overcome their limitations. The explanations are obtained in terms of input features that represent more salient and understandable input properties for a user [9, 10, 11, 6], which we call *Middle-Level input Features (MLFs)* (see, as an example, Figure 1).

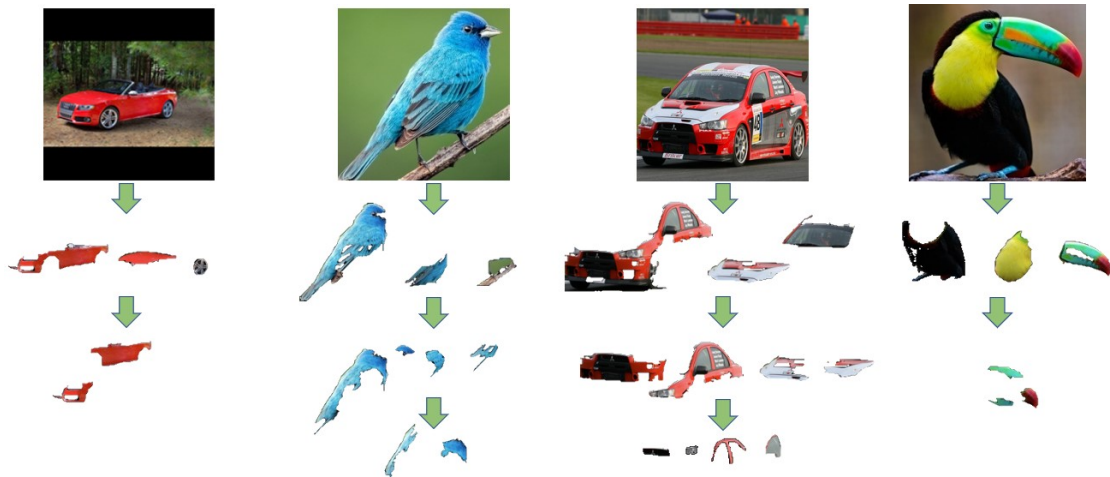


Figure 1: Examples of Middle Level input Features (MLFs) organized in a hierarchical way. Each MLF represents a part of the input which is perceptually and cognitively salient to a human being. Each MLF can be viewed as composed by finer MLF and so on. MLF are intuitively more humanly interpretable respect to low-level features (as for example raw unrelated image pixels).

Another interesting and very recent approach is that of considering hierarchically organised explanations [12, 13, 14, 13]. Thus, in this paper, we investigate the possibility to combine both MLFs and hierarchical organisations. The potential advantages of providing explanations in terms of hierarchically organised MLFs are grounded on the possibility of exhibiting explanations to a different granularity of MLFs interacting with each other. For example, natural images can be described in terms of the objects they show at various levels of granularity, and their relations [15, 16, 17]. In this paper, in particular, we take advantage of using a general framework that we recently proposed in a paper under review [18]. The paper is organised as follows: in Section 2 we discuss differences and advantages of our approach with respect to similar approaches

presented in the literature; Section 3 describes in detail the proposed approach; experiments and results are discussed in Section 4 and 5; the concluding Section summarises the main high-level features of the proposed explanation framework and outlines some future developments.

2. Background

Building explanations in terms of Middle-Level input Features (MLFs) is a growing research area in the XAI community. For example, in [9] Concept Activation Vectors (CAV) are introduced as a way to visually represent human-understandable concepts. These concepts are extracted by an external labelled dataset. The authors proposed a method to quantify the influence of each concept on the classifier output. CAV-based explanations are expressed in terms of high-level visual concepts which, unlike our approach, do not necessarily belong to the ML system input. In [10], the CAVs are automatically extracted without using external labelled data expressing human-friendly concepts, but to produce explanations related to an entire class (*global* explanation) instead of the single ML system input (*local* explanation). Thus, again, human users are left with a significant interpretive load: starting from external high-level visual concepts, the human user needs to identify the input properties perceptually and cognitively related to these concepts. On the contrary, in our approach MLFs are expressed in terms of elements belonging to the input itself. In [19] LIME is proposed. Especially in the context of images, LIME is one of the predominant XAI methods discussed in the literature [20, 21]. It provides local explanations in terms of relevant image regions of the input that the classifier receives. In [11] the authors use the concept of “fault lines”, defined as “high-level semantic aspects of reality that humans zoom in on when imagining an alternative to it”. The proposed explanations are given in terms of images representing semantic aspects which should help the user to understand why the classifier output was a given label instead of another one. The semantic aspects are constructed using feature maps of the pre-trained Convolutional Neural network. Thus, a critical point is that high-level or middle-level user-friendly concepts are computed on the basis of the neural network classifier to be explained. In this way, an unsafe short-circuit can be created in which the visual concepts used to explain the classifier are closely related to the classifier itself. This fact could lead to the creation of false human-friendly visual concepts if the classifier is not reliable. By contrast, in our approach, MLFs are extracted independently from the classifier.

A crucial aspect that distinguishes our proposal from the above-discussed research line is grounded on the fact that we propose explanations in terms of hierarchically organised MLFs.

From the point of view of the hierarchical properties, a number of research works have already tried to give hierarchical explanations in XAI domain. In [13] a method to build hierarchical saliency map exploiting the relations between the input features is proposed. In [12] the Mahé framework is described. As in LIME, Mahé perturbs the input data and construct a proxy model using the outputs of the original model on the perturbed data. Differently from LIME, Mahé uses a Neural Network as model approximator instead of a linear model. As in [13], the possible interactions between variables are captured to build hierarchical explanations. However, none of these two methods focus on input features that represent more salient and understandable input properties such as MLFs. In [14] another method for hierarchical explanation is proposed,

but limited to text classification problems.

3. Proposal

3.1. General description

Given an ML classification model M which receives an input $\mathbf{x} \in R^d$ and outputs $\mathbf{y} \in R^c$, we want to produce an explanation of the output \mathbf{y} in terms of MLFs organised in a hierarchical way. Our XAI method is based on a more general approach which we have proposed in a paper currently under review [18]. Our method can be divided into two consecutive steps.

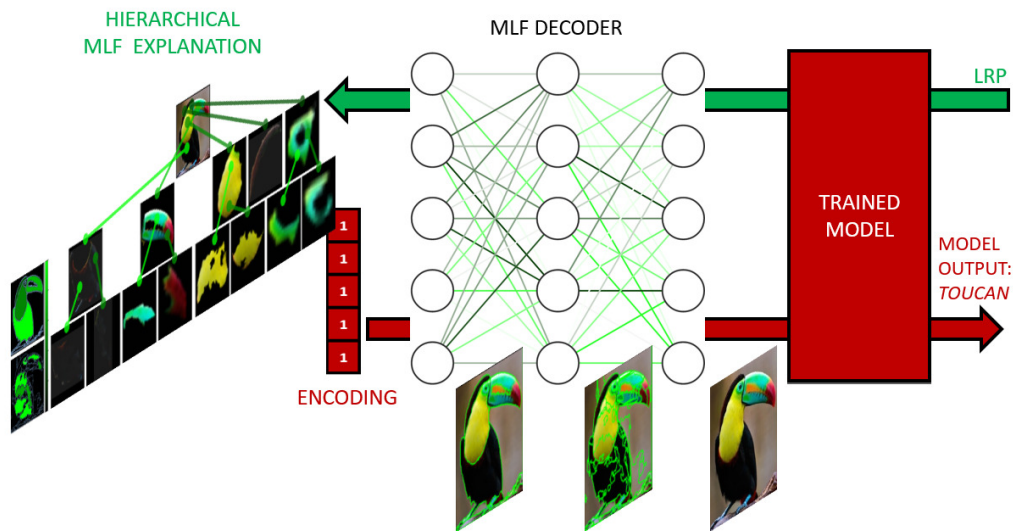


Figure 2: The proposed method. A neural networks having as weights the segments returned by a hierarchical segmentation algorithm is used to model the Middle-Level Features decoder. Each layer of the the hierarchical segmentation is represented by a MLF decoder layer, going from the coarser to the finest detail level. In the last layer, each pixel composing the input image is considered as a segment. The initial encoding fed to the MLF decoder is the 1 vector since all the segments compose the input image. For each hierarchical layer, the relevance backward algorithm returns the most relevant segments (see text for further details).

In the first step, we build an auto-encoder $AE \equiv (E, D)$ such that the input \mathbf{x} can be represented by the encoder E in a hierarchical structure. For instance, considering \mathbf{x} as an input image, we generate an encoder which returns a set of image partitions organised in a hierarchical way, i.e. each coarser detail partition can be obtained by merging partitions elements representing finer details. As discussed in [22], this target can be achieved by a segmentation algorithm which ensures the following two conditions: 1) if a contour is present at a given scale, the same contour has to be present at any finer scale (*causality principle* of the multiscale analysis [23]) and 2) even when the number of regions decreases, the contours remain stable (*location principle*). If a segmentation algorithm ensures these conditions, then it can be

considered a hierarchical segmentation algorithm. More formally, given a set of K different partitions $\{S_1, S_2, \dots, S_K\}$ of the image $\mathbf{x} \in R^d$ produced by a segmentation algorithm sorted from the coarse to the finer detail level, each region $\mathbf{v}_i^k \in S_k$ can be expressed as a linear combination of the elements of the finer level S_{k+1} , i.e. $\mathbf{v}_i^k = \sum_j \alpha_{ij}^k \mathbf{v}_j^{k+1}$, where $\alpha_{ij}^k =$

$$\begin{cases} 1 & \text{if all the pixels in } \mathbf{v}_j^{k+1} \in \mathbf{v}_i^k \\ 0 & \text{otherwise,} \end{cases}$$

for each $k \in \{1, 2, \dots, K\}$, and each \mathbf{v}_i^k is a candidate MLF to be included in the explanation. In this way, it is straightforward to build a feed-forward neural network as an autoencoder which outputs a reconstruction of the image \mathbf{x} exploiting a hierarchical segmentation of \mathbf{x} in its own internal layers. In a nutshell, given a fully-connected feed-forward neural network of $K + 1$ layers having $|S_k|$ inputs and $|S_{k+1}|$ outputs for $k \in \dots, 1, 2, \dots, K$, and S_K inputs and d outputs for the final $K + 1$ layer, we can set the identity as activation functions, biases equal to 0 and each weights $w_{ij}^k = \alpha_{ij}^k$ for $k \in \{1, 2, \dots, K\}$. For the last layer, we can consider the image \mathbf{x} as the trivial partition where each partition elements represents a single image pixel, i.e. $S_{K+1} = \{\mathbf{v}_1^{K+1}, \mathbf{v}_2^{K+1}, \dots, \mathbf{v}_d^{K+1}\}$

with $v_{ij}^{K+1} = \begin{cases} v_{ij}^{K+1} = x_j & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases}$. The weights of the $K + 1$ layer can be set equal

to $(\mathbf{v}_p^{K+1})_{p=1}^d$. The resulting network, fed with the $\mathbf{1}$ vector, outputs the original image \mathbf{x} . It can be viewed as a decoder D that decodes the sequence of all 1s as \mathbf{x} . Being a simple feed forward neural network, once stacked on the top of the model M , each relevance propagation method can be used to give a relevance score to each MLF \mathbf{v}_i^k , hierarchically organised by D . In Algorithm 1 our approach is reported in pseudo-code, while in Fig. 2 a graphical representation of the proposed method is shown.

4. Experimental setup

In this section the setup used to validate the proposed method is described. In Section 5 a set of possible explanations of the classifier outputs on image sampled from *300 Birds Species* [24] and *Cars* [25] dataset are shown and discussed. As classifier, a VGG16 [26] architecture trained on Imagenet was used. We used as MLFs the set of segmentations produced by [22]. This algorithm ensures the causality and the location principle discussed in Section 3, so the returned segmentations can be considered hierarchically related. However, any segmentation algorithm which respects the principles described in Section 3 can be used. Firstly, h sets of segments $\{S_i\}_{i=1}^h$ related between them in a hierarchical way are generated for each test image to explain, going from the coarsest ($i = 1$) to the finest ($i = h$) segmentation level. These segments are the candidate MLFs for providing an hierarchically organised explanation. Next, the weights of a h -layers fully-connected neural network are set using the obtained segmentations as described in Section 3. Finally, the network is stacked on the top of VGG16 model and the LRP algorithm is applied to the whole network fed with the "1"s vector. An LRP heatmaps for each hierarchical layer is plotted and the most relevant MLFs are highlighted.

Algorithm 1: Hierarchical segmentation-based explanation Generator

Input: data point $\mathbf{x} \in R^d$, hierarchical segmentation procedure seg and its parameters $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_K)$, a relevance propagation algorithm RP which outputs the relevances

Output: A hierarchical explanation \mathbf{exp} of the model M on the input \mathbf{x} . \mathbf{exp} will be composed of K relevance vectors indicating the importance feature scores for each layer of the hierarchy.

```
1  $\{S_1, S_2, \dots, S_K\} \leftarrow seg(\mathbf{x}, \lambda)$ ;
2 let  $S_{K+1} \leftarrow \emptyset$ ;
3 for  $x_j \in \mathbf{x}$  do
4   let  $\mathbf{v}^{K+1} \in \{0\}^d$ ;
5    $v_{jj}^{K+1} \leftarrow x_j$ ;
6    $S_{K+1} \leftarrow S_{K+1} \cup \{\mathbf{v}^{K+1}\}$ ;
7 end
8 for  $1 \leq k \leq K$  do
9   let  $W^k \in \{0\}^{|S_k| \times |S_{k+1}|}$ ;
10  let  $\mathbf{b}^k \in \{0\}^{|S_{k+1}|}$ ;
11  for  $1 \leq i \leq |S_k|$  do
12    for  $1 \leq j \leq |S_{k+1}|$  do
13      if  $\mathbf{v}_j^{k+1}$  belongs to  $\mathbf{v}_i^k$  then
14         $W_{ij}^k \leftarrow 1$ ;
15      end
16    end
17  end
18   $D \leftarrow makeFullConnectedNeuralNetwork(weights = \{W^k\}_{k=1}^{K+1},$ 
19                                     biases =  $\{\mathbf{b}^k\}_{k=1}^{K+1},$ 
20                                     activation fun =  $identity$ );
21  define  $E : \mathbf{x} \mapsto e \in \{1\}^{|S_1|}$ ;
22  stack together  $(D, M)$ ;
23   $\mathbf{exp} \leftarrow RP(\mathbf{x}, E, (D, M))$ 
24  return  $\mathbf{exp}_{1, \dots, K}$ ;
25 end
```

5. Results

In this section, an evaluation of our approach is reported and discussed with input images taken from *300 Birds Species* and *Cars* datasets using $h \in \{2, 3\}$ hierarchy layers. The evaluation is made in both qualitative and quantitative terms. The former showing the explanations produced by the proposed method on several inputs of *300 Birds Species* and *Cars* dataset fed to VGG16 model, the latter showing the MoRF (Most Relevant First) curves [3, 27]. For all the inputs, a comparison with the explanations generated by LIME method is made.

5.1. Qualitative evaluation

5.1.1. Dataset 300 Birds Species

In Fig. 3 an example of a two-layer hierarchical explanation of the class *bald eagle*, correctly assigned to an input image, compared with LIME explanation is reported. One can observe that the first hierarchical level highlights the head as the most significant MLF for the returned class. Going deeper into the hierarchy, it is possible to see that not all the head parts contributed at the same way to the returned output. Instead, the head plumage and the beak seem to have a greater importance in the final classification. By contrast, one can note that LIME produces “flat” explanations, without any relation between different segments. In other words, given a selected input partition, LIME outputs similar results with respect our approach, but it is unable to relate segments belonging to different input partitions.

Similar results are shown in Figure 4, 5, 6 and 7. In Fig. 4 the first explanation level highlights that the swan body and the beak have been relevant for the final classification. Going deeper into the hierarchy, it is possible to see that the neck and the head result to be determinant for the obtained output.

In Fig. 5 in the first hierarchical layer is highlighted how the upper body part is particularly relevant for the classification. Next into the hierarchy, one can note that the neck and the wing result to be particularly incisive for the model output. In Fig. 6 the first hierarchical level, the most relevant middle-level-feature is the flamingo body. Interestingly, the sky has also a high weight for the classification (2nd position). In the next hierarchy level, it is highlighted which body parts contributed most to the final result, that is its distinctive S-shaped neck. Similar considerations can be done for the Fig 7 where the most relevant part of the image is almost the entire pelican. The relevance of the sky has great incidence for the classification (2nd position). Going deeper along the hierarchy, the most important parts of the whole body result to be the wing and the beak.

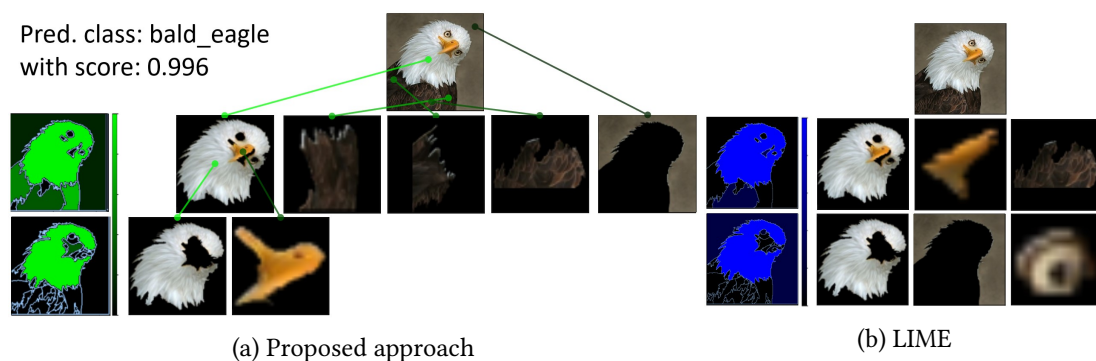


Figure 3: An example of a two-layer hierarchical explanation of the class *bald eagle* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

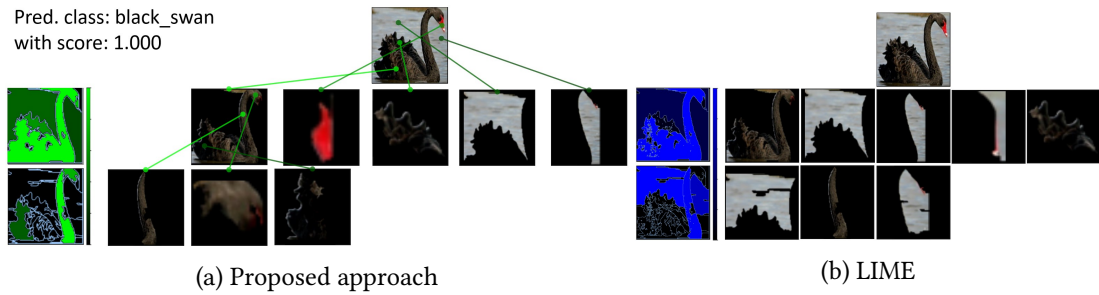


Figure 4: An example of a two-layer hierarchical explanation of the class *black_swan* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

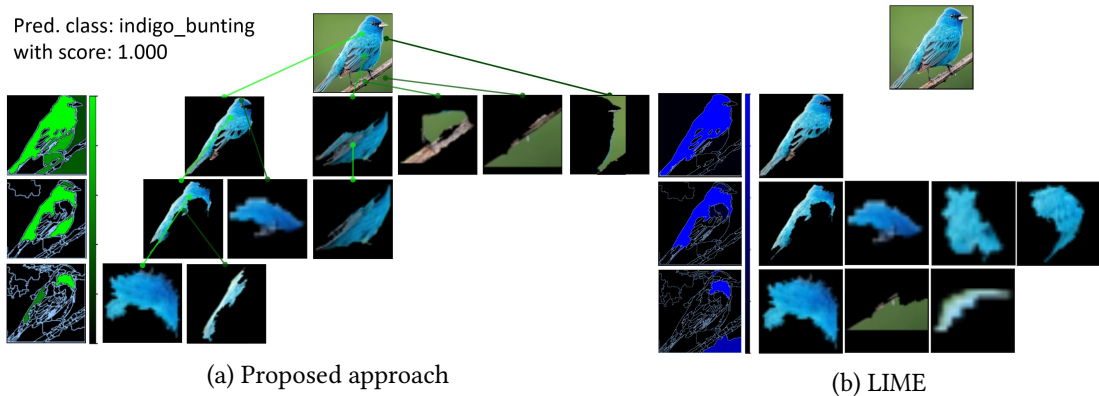


Figure 5: An example of a three-layer hierarchical explanation of the class *indigo_bunting* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) to the finest (last row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

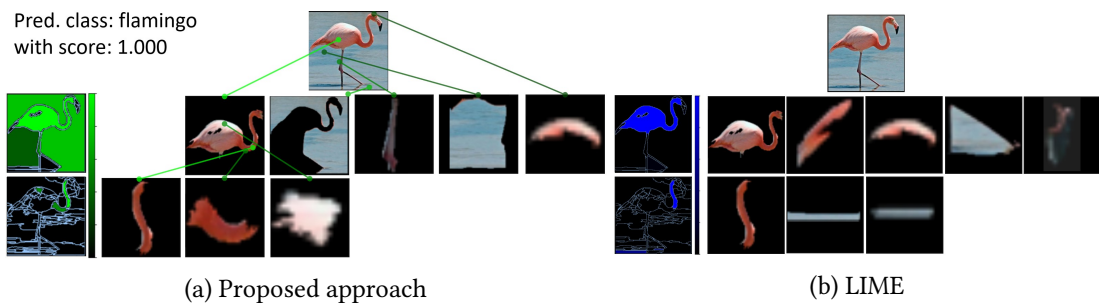


Figure 6: An example of a two-layer hierarchical explanation of the class *flamingo* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

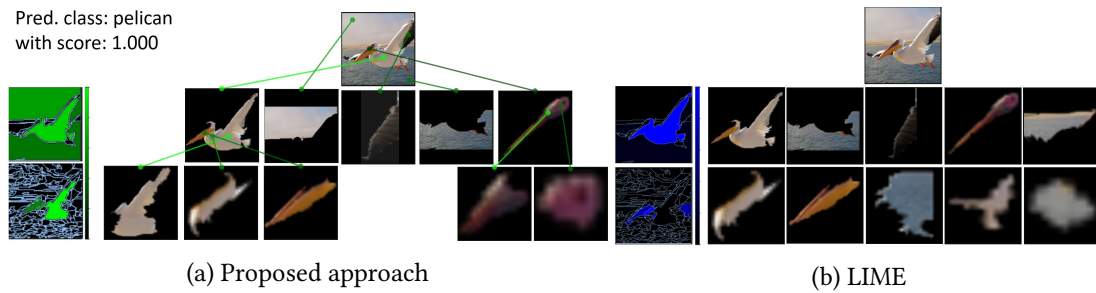


Figure 7: An example of a two-layer hierarchical explanation of the class *pelican* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

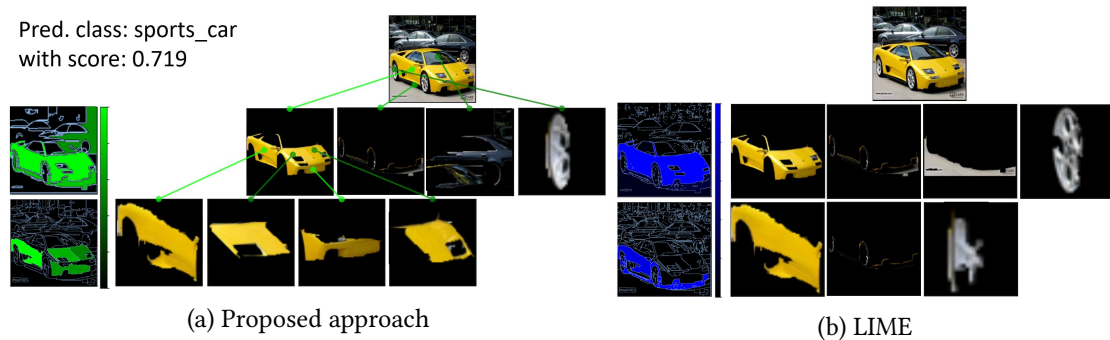


Figure 8: An example of a two-layer hierarchical explanation of the class *sports_car* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

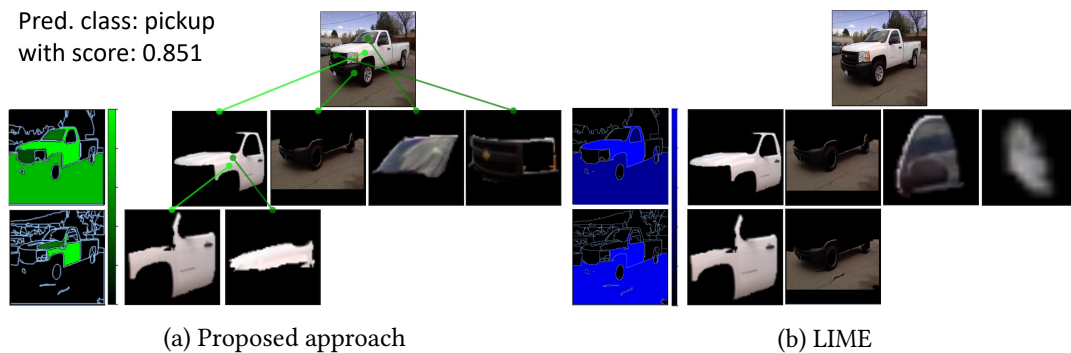


Figure 9: An example of a two-layer hierarchical explanation of the class *pickup* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

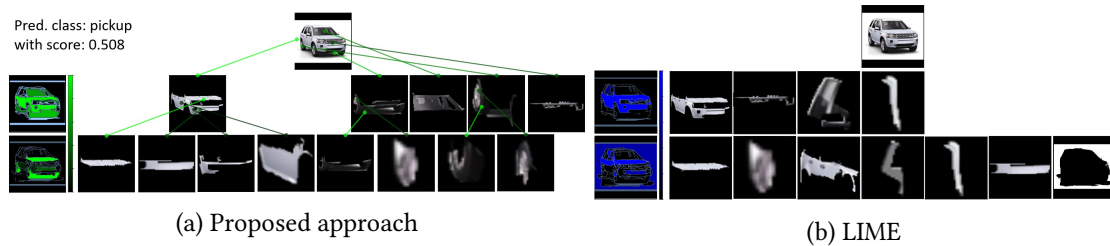


Figure 10: An example of a two-layer hierarchical explanation of the class *pickup* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

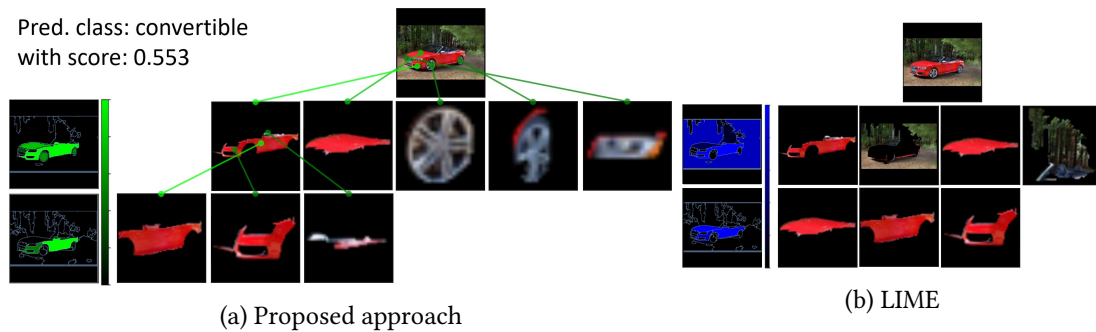


Figure 11: An example of a two-layer hierarchical explanation of the class *convertible* correctly assigned to an input image (first row) by VGG16. (a) First column: segment heat map. Left to right: segments sorted in descending relevance order. Top-down: the coarsest (second row) and the finest (third row) hierarchical level. (b) LIME explanation: same input, same segmentation used in (a).

5.1.2. Dataset Cars

In Fig. 8, an image containing a sports car is fed to VGG16 classifier. The proposed explanation method returns the car bodywork as the most relevant one for the classification output (left side, first row). Exploiting the second layer of the hierarchy (left side, second row), the different parts of the bodywork sorted by relevance can be highlighted helping the user to better understand the output. Interestingly, LIME returns the same most relevant segment (right side, first row). However, with a finer segmentation (right side, second row), no relation with the coarser one is taken into account. Similar consideration can be done for Fig. 9, 10, and 11.

5.2. Quantitative evaluation

The MoRF (Most Relevant First) curve analysis is a method to evaluate the explanation produced by an XAI method proposed and discussed in [3, 27]. In a nutshell, features of the input are iteratively replaced by random noise and fed to the classifier, in the descending order with respect to the relevance values given by the explanation to evaluate. Finally, all the probability values of the original class can be plotted generating a curve. We expect that the better the explanation method is, the steeper the curve the better the explanation. Since the proposed

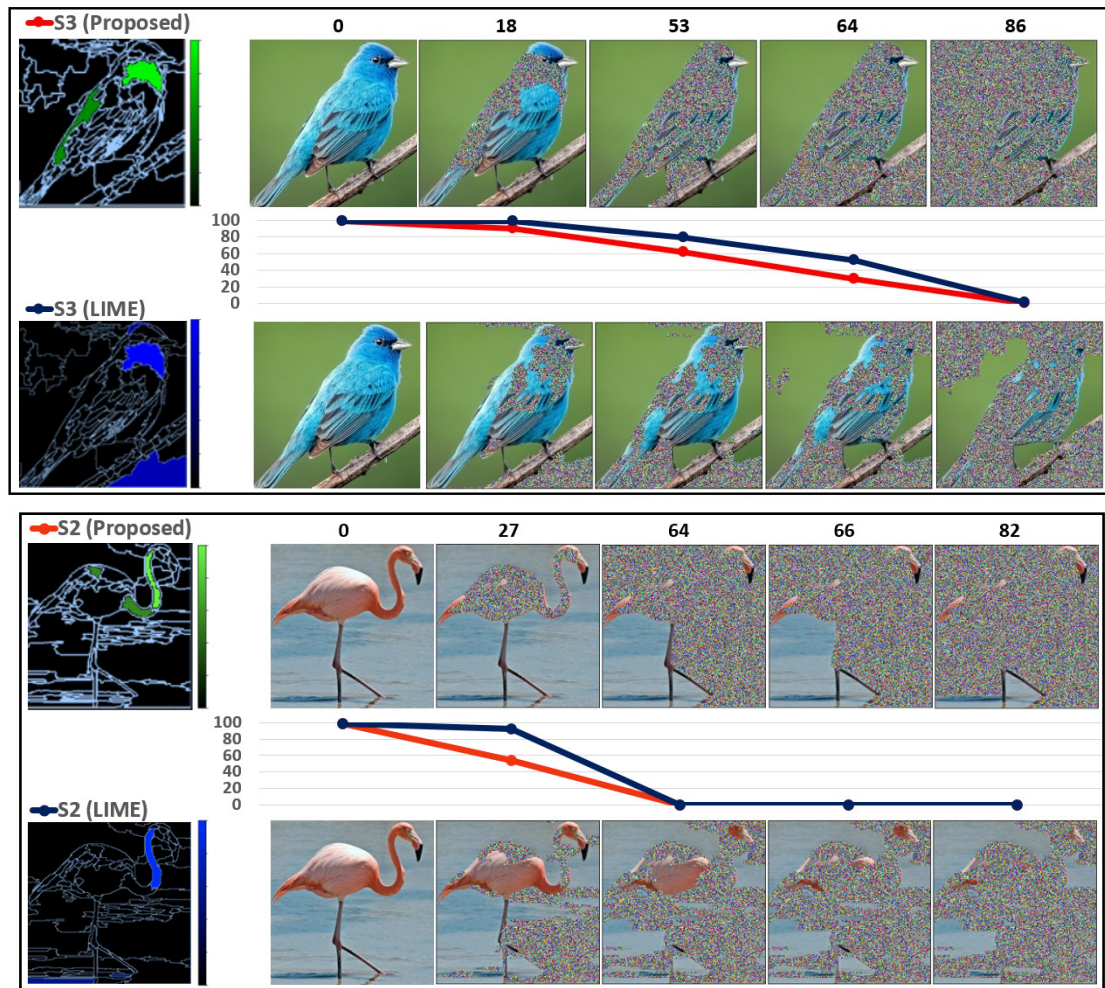


Figure 12: MoRF curves computed with the proposed approach (red) and LIME (blue) using the last layer MLF as segmentation for both methods. At each iteration step, a perturbed input based on the returned explanation is fed to the classifier. On the y axis of the plot, the classification probability of the original class for each perturbed input. On the x axis, the perturbation step. The figures in the first and the second row show the perturbed inputs fed to the classifier at each iteration for the proposed explanation and the LIME explanation, respectively.

approach relies on Middle Level Features, the MoRF curves are computed following the *region flipping* as perturbation schema, a generalisation of the *pixel-flipping* measure proposed in [3]. Furthermore, MLFs were removed from the inputs, exploiting the hierarchy in a topological sort depth-first search based on the descending order's relevances. Therefore, the MLFs of the finest hierarchical layer were considered.

As baseline, MoRF obtained with LIME are produced. To make the results comparable, the same segmentation algorithm was used both with the proposed work and LIME, instead of the segmentation algorithm used in the original paper. Comparing the MoRF curves related to

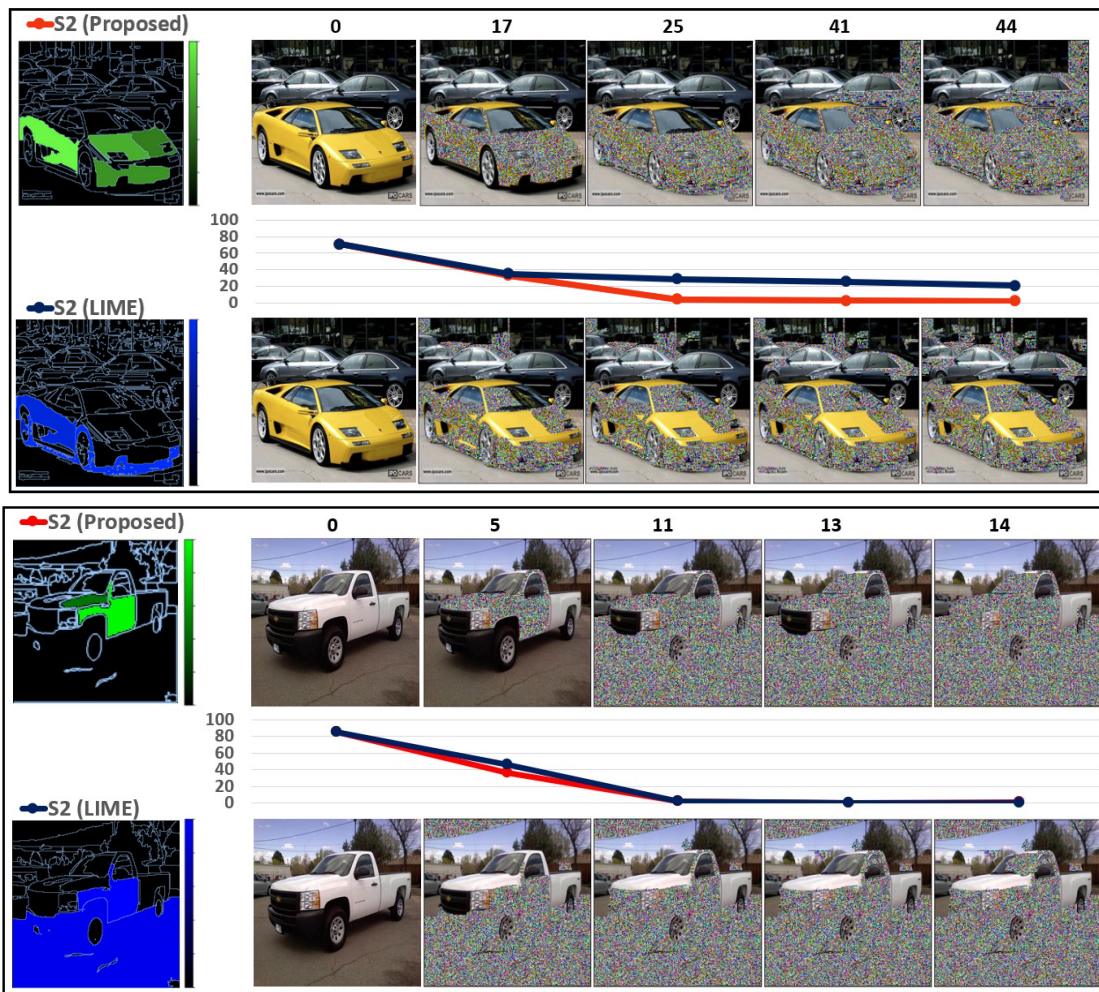


Figure 13: MoRF curves computed with the proposed approach (red) and LIME (blue) using the last layer MLF as segmentation for both methods. At each iteration step, a perturbed input based on the returned explanation is fed to the classifier. On the y axis of the plot, the classification probability of the original class for each perturbed input. On the x axis, the perturbation step. The figures in the first and the second row show the perturbed inputs fed to the classifier at each iteration for the proposed explanation and the LIME explanation, respectively.

the explanation obtained by the proposed method and LIME, it results that in most cases the proposed approach returns comparable explanations or better, as shown in Fig. 12 and 13.

6. Conclusion

We proposed in this paper to combine two recent approaches in the XAI research literature: explanations in terms of input features that represent more salient and understandable input properties for a user, Middle-Level input Feature (MLF), and their hierarchical organisation. To

this aim, a new XAI method has been proposed, built on a more general approach proposed in a paper [18] freely available on Arxiv. We have experimentally evaluated the advantages of providing explanations in terms of hierarchically organised MLFs. In particular, we have highlighted the possibility of exhibiting explanations to a different granularity of MLFs interacting with each other. The experiments were conducted using *300 Birds Species* and *Cars* as dataset, and *VGG16* as classifier. Our approach was evaluated in both qualitatively and quantitatively manner and compared with LIME. The preliminary results seem encouraging. When it is possible to find MLF relationships at different degrees of input granularity, we can obtain explanations easier to understand.

Acknowledgments

This work was carried out under the initiative “Departments of Excellence” (Italian Budget Law no. 232/2016), through an excellence grant awarded to the Department of Information Technology and Electrical Engineering of the University of Naples Federico II, Naples, Italy.

References

- [1] M. Ribera, A. Lapedriza, Can we do better explanations? a proposal of user-centered explainable ai., in: *IUI Workshops*, 2019.
- [2] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai, *Information Fusion* 58 (2020) 82–115.
- [3] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, W. Samek, On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation, *PloS one* 10 (2015) e0130140.
- [4] A. Nguyen, J. Yosinski, J. Clune, Multifaceted Feature Visualization: Uncovering the Different Types of Features Learned By Each Neuron in Deep Neural Networks, *ArXiv e-prints* (2016).
- [5] D. Doran, S. Schulz, T. R. Besold, What does explainable AI really mean? A new conceptualization of perspectives, *CoRR abs/1710.00794* (2017).
- [6] A. Apicella, F. Isgro, R. Prevete, G. Tamburrini, Middle-level features for the explanation of classification systems by sparse dictionary methods, *International Journal of Neural Systems* 30 (2020) 2050040.
- [7] Q. Zhang, S. Zhu, Visual interpretability for deep learning: a survey, *Frontiers of Information Technology & Electronic Engineering* 19 (2018) 27–39.
- [8] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: Visualising image classification models and saliency maps, in: *2nd International Conference on Learning Representations, Workshop Track Proceedings*, Banff, Canada, 2014.
- [9] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, et al., Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav), in: *International conference on machine learning*, PMLR, 2018, pp. 2668–2677.

- [10] A. Ghorbani, J. Wexler, J. Zou, B. Kim, Towards automatic concept-based explanations, arXiv preprint arXiv:1902.03129 (2019).
- [11] A. Akula, S. Wang, S.-C. Zhu, Cocox: Generating conceptual and counterfactual explanations via fault-lines, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 2020, pp. 2594–2601.
- [12] M. Tsang, Y. Sun, D. Ren, Y. Liu, Can i trust you more? model-agnostic hierarchical explanations, arXiv preprint arXiv:1812.04801 (2018).
- [13] C. Singh, W. J. Murdoch, B. Yu, Hierarchical interpretations for neural network predictions, in: International Conference on Learning Representations, 2019.
- [14] H. Chen, G. Zheng, Y. Ji, Generating hierarchical explanations on text classification via feature interaction detection, arXiv preprint arXiv:2004.02015 (2020).
- [15] M. Tschannen, O. Bachem, M. Lucic, Recent advances in autoencoder-based representation learning, arXiv preprint arXiv:1812.05069 (2018).
- [16] C. K. Sønderby, T. Raiko, L. Maaløe, S. K. Sønderby, O. Winther, Ladder variational autoencoders, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016, pp. 3745–3753.
- [17] S. Zhao, J. Song, S. Ermon, Learning hierarchical features from deep generative models, in: International Conference on Machine Learning, PMLR, 2017, pp. 4091–4099.
- [18] A. Apicella, F. Isgrò, R. Prevede, A general approach for explanations in terms of middle level features, arXiv preprint arXiv:2106.05037 (2021).
- [19] M. T. Ribeiro, S. Singh, 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, KDD '16, ACM, New York, NY, USA, 2016, pp. 1135–1144.
- [20] J. Dieber, S. Kirrane, Why model why? assessing the strengths and limitations of lime, arXiv preprint arXiv:2012.00093 (2020).
- [21] X. Zhao, X. Huang, V. Robu, D. Flynn, Baylime: Bayesian local interpretable model-agnostic explanations, arXiv preprint arXiv:2012.03058 (2020).
- [22] S. J. F. Guimarães, J. Cousty, Y. Kenmochi, L. Najman, A hierarchical image segmentation algorithm based on an observation scale, in: Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), Springer, 2012, pp. 116–125.
- [23] L. Guigues, J. P. Cocquerez, H. Le Men, Scale-sets image analysis, International Journal of Computer Vision 68 (2006) 289–317.
- [24] G. Piosenka, 300 bird species dataset, 2014. Data retrieved from 300 Bird Species Dataset, <https://www.kaggle.com/gpiosenka/100-bird-species>.
- [25] J. Krause, M. Stark, J. Deng, L. Fei-Fei, 3d object representations for fine-grained categorization, in: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13), Sydney, Australia, 2013.
- [26] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2015.
- [27] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, K.-R. Müller, Evaluating the visualization of what a deep neural network has learned, IEEE transactions on neural networks and learning systems 28 (2016) 2660–2673.