Heatmap Assisted Accuracy Score Evaluation Method for Machine-Centric Explainable Deep Neural Networks

JUNHEE LEE¹, HYEONSEONG CHO¹, YUN JANG PYUN¹, SUK-JU KANG², AND HYOUNGSIK NAM¹, (Member, IEEE)

¹Department of Information Display, Kyung Hee University, Seoul 02447, Republic of Korea
²Department of Electronic Engineering, Sogang University, Seoul 04107, Republic of Korea
Corresponding author: Hyoungsik Nam (e-mail: hyoungsiknam@khu.ac.kr).

This research was supported by the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2019R1F1A1061114) and the Brain Korea 21 Four Program in 2022.

ABSTRACT

There have existed many studies about the explainable artificial intelligence (XAI) that explains the logic behind the complex deep neural network called a black box. At the same time, researchers have tried to evaluate the explainability performance of various XAIs. However, most previous evaluation methods are human-centric, that is, subjective, where they rely on how much the results of explanation are similar to what people’s decision is based on rather than what features actually affect the decision in the model. Their XAI selections are also dependent of datasets. Furthermore, they are focusing only on the output variation of a target class. On the other hand, this paper proposes a robust heatmap assisted accuracy score (HAAS) scheme over datasets that helps selecting machine-centric explanation algorithms to show what actually leads to the decision of a given classification network. The proposed method modifies the input image with the heatmap scores obtained by a given explanation algorithm and then puts the resultant heatmap assisted (HA) images into the network to estimate the accuracy change. The resultant metric (HAAS) is computed as a ratio of accuracies of the given network over HA and original images. The proposed evaluation scheme is verified in the image classification models of LeNet-5 for MNIST and VGG-16 for CIFAR-10, STL-10, and ILSVRC2012 over totally 11 XAI algorithms of saliency map, deconvolution, and 9 layer-wise relevance propagation (LRP) configurations. Consequently, for LRP1 and LRP3, MINST showed largest HAAS values of 1.0088 and 1.0079, CIFAR-10 achieved 1.1160 and 1.1254, STL-10 had 1.0906 and 1.0918, and ILSVRC2012 got 1.3207 and 1.3469. While LRP1 consists of $\epsilon$-rules for input, convolutional, and fully-connected layers, LRP3 adopts a bounded-rule for an input layer and the same $\epsilon$-rules for other layers as LRP1. The consistency of evaluation results of HAAS and AOPC has been compared by means of Kullback-Leibler divergence, ensuring that HAAS is the more robust evaluation method than AOPC independently of datasets since HAAS has much lower average divergence of 0.0251 than AOPC of 0.3048. In addition, the validity of the proposed HAAS scheme is further investigated through the inverted HA test that employs inverted HA images made up with inverted heatmap scores and estimates the accuracy degradation caused by applying them to the network. The XAI algorithms with largest HAAS results experience biggest accuracy degradation in the inverted HA test.

INDEX TERMS Heatmap, HAAS, evaluation, explainability, machine-centric, classification, deep neural network.

I. INTRODUCTION

Deep neural networks (DNNs) have gotten humongous attention thanks to a huge amount of datasets for training [1]–[3], the drastic advance in the computing hardware [4]–[9], and the advent of cloud computing [10]–[13], that have enabled the implementation of large number of layers and neurons to make much higher performance possible. DNNs have been developed in various network structures
of multi-layer perceptron (MLP) [14], convolutional neural network (CNN) [15], recurrent neural network (RNN) [16], [17], generative adversarial network (GAN) [18], [19], and graph neural network (GNN) [20]. They have been also employed in a variety of applications such as classification [21], [22], object detection [23], segmentation [24], super resolution [25], machine translation [26], natural language processing [27], image/caption/speech generation [28]–[30], algorithmic trading [31], style transfer [32], circuit design [33], recommendation system [34], and so on. Until recently, many researches have been focusing on improving the performance of the networks by means of the increased computational complexity.

In contrast to the substantial achievement with respect to performance, the increased complexity has made it difficult to understand the logic behind the decisions from the networks. Therefore, DNNs are referred to black boxes. Since DNNs cannot guarantee the perfect performance like the accuracy of 100%, their decisions need the explanation in some applications such as transportation, health care, legal, finance, and military areas [35]. Whereas transportation, health care, and military fields involve in the life-related decisions, legal and finance areas are regulated to provide reasons over decisions by law.

There have been many researches to provide the explainability over DNNs that are addressed as the explainable artificial intelligence (XAI). The importance of a feature was estimated by calculating the expectation of the output over other remaining features or by fixing them at randomly selected values [36], [37]. The sensitivity of the output computed by its derivative with respect to a specific input feature was considered as features’ importance, based on the assumption that more important features would cause larger variations of the outputs even at their small changes [38], [39]. The output of a specific target class was simply back-propagated toward the input layer through a given network by means of the deconvolution, resulting in the heatmap image that represented feature importance scores [40]. On the one hand, because simpler models were easy to interpret, the distillation scheme was proposed, where the knowledge of the high-capacity model was transferred to the small model by training the small one on the soft targets generated by the complicated model [41], [42]. Layer-wise relevance propagation (LRP) [43] used the back-propagation of the relevance score from the output layer to the input layer based on the relevance conservation rule. The propagated relevance values in the input layer were referred to the importance of the corresponding features. There were techniques to explain the models via prototype and criticism [44]. Prototype and criticism can be seen as good and bad examples, respectively. On the other hand, after the subsets of features were randomly selected by the masking data and processed by the deep network model, a simple linear model was built between masking data and corresponding outputs [45]. The weights of the linear model were interpreted as the features’ importance. The class activation map (CAM) method [46]–[48] was introduced to focus on the highest-level feature map that was the last convolutional layer. In addition, the counterfactual approach was proposed to show how the model would have to be different for a desirable output to occur [49].

Because it is impossible to realize any XAI methods that can provide the perfect explanations about decisions of DNNs, we need the evaluation on their explainability to figure out which one is the best explanation approach for a given application. To the best of our knowledge, most previous XAI evaluation approaches, which this paper classifies into human-centric methods, have focused on how similar their explanations look to what people’s decisions rely on [50]. On the other hand, this paper proposes a machine-centric evaluation method that takes into account how well the feature importance extracted by XAI reflects their actual contribution to the decision of the network. Consequently, while the results of previous methods show the dependency on datasets, the proposed evaluation scheme achieves the high consistency over datasets at the best XAI selection. Although this paper uses the network models for the image classification problem, it is ensured that the proposed idea is expandable to other decision problems.

This paper is organized as follows. Section II addresses the overview of the previous evaluation methods along with their pros and cons, and then Section III describes the proposed HAAS evaluation scheme. Section IV shows evaluation results and discussions. Section V concludes this paper.

II. PREVIOUS EVALUATION METHODS

The human-friendly explanation is one of key properties that XAIs must have [35]. It means that the explainability methods should provide representations that people can understand. However, too much emphasis on the human-friendliness in previous methods resulted in the human-centric explanation which was biased on how human beings made a decision for given problems rather than what the current model actually concluded it from. Therefore, in most previous evaluation methods, better explainability was equivalent to what the people saw the better was, that is, the subjective criterion.

SmoothGrad [39] was introduced to provide cleaner explanation by removing many scattering points at the saliency map [38]. The scattering points could be eliminated by adding some noises to the input features without consideration on whether those scattering points in the saliency map were important for the decision or not. The resultant human-centric sensitivity map showed only the group of points gathered at the target object that people’s detection would be normally based on.

Guided back-propagation [51] represented the feature importance by propagating backwards only through neurons of positive gradients in order to visualize the part that most significantly activates the decision. Because the representations of positive gradients at the first layer were emphasized, the edges of objects were highlighted. The guided back-propagation gave rise to the edge-based explanation to show
the shapes of objects that were believed to be the basis for people’s judgment. Therefore, this method could also be classified as human-centric.

Grad-CAM [47] and Grad-CAM++ [48] was proposed to localize the area of a target object in a given image rather than to figure out the feature importance. Grad-CAM++ obtained parameters of feature maps from a weighted average and localized the entire object area unlike Grad-CAM that could cover only its parts. Especially, its performance was evaluated objectively in terms of average drop%, %increase in confidence, and win%, compared to Grad-CAM. Average drop% was the drop in confidence by occluding unimportant regions, and win% was the number of cases in given images when the fall in the confidence for a map generated by Grad-CAM++ was lower than that by Grad-CAM. The performance of Grad-CAM++ was higher than Grad-CAM, however, it was natural because Grad-CAM++ highlighted bigger regions containing Grad-CAM.

LRP [43] gave rise to the fine-grained pixel-wise explanation by propagating relevance scores from the output of a target class to inputs in a fashion of layer by layer, where relevance values were computed with weights as well as activation outputs. Whereas other XAI schemes could focus on only the features of positive importance, LRP was able to provide the explanation about positive as well as negative influences. Besides a basic $\varepsilon$-rule, additional rules of relevance propagation such as $\alpha\beta$-rule, flat-rule, and bounded-rule [52], [53] that are described in more detail in Section IV, were proposed to accomplish more human-centric explanation. Previous LRP papers subjectively evaluated their explainability based on how well the target object area was highlighted.

On the other hand, there have been quantitative evaluation approaches based on the heatmap visualization for image classification problems. The representative one was the area over the MoRF perturbation curve (AOPC) [54], where MoRF stood for most relevant first. The image was divided into predefined grids ($r_1, \ldots, r_L$) which locations were ordered according to heatmap scores. The score of $r_i$ is equal to or larger than $r_j$ when $i < j$. Heatmap scores indicate how important the location is for representing the output class. The MoRF process was conducted recursively by adding perturbations to highest score regions of the image as described in (1) and Fig. 1, where $g$ was a function to eliminate the information of $r_k$ from the modified image after $(k-1)$ MoRF operations ($x^{(k)}_{MoRF}$) and $x^{(0)}_{MoRF}$ was the original image $x$.

$$x^{(k)}_{MoRF} = g(x^{(k-1)}_{MoRF}, r_k), \quad 1 \leq k \leq L \quad (1)$$

In each recursive step, the difference ($Diff(f(k))$) between model outputs for original and perturbed images of $x^{(0)}_{MoRF}$ and $x^{(k)}_{MoRF}$ was plotted, and the final AOPC value became equivalent to the area under the difference curve, as expressed in (2), leading to the conclusion that a larger AOPC is obtained from a better XAI method. $\frac{1}{L}$ is the average function over all images. Consequently, AOPC results showed that LRP algorithms were superior to saliency map [38] and deconvolution [40]. However, AOPC requires lots of iterations to achieve the evaluation results only with consideration on highest heatmap scores. Since AOPC is mainly determined by the first highest score location, it cannot accomplish the fair evaluation on other heatmap score areas. Also, AOPC makes use of the output value difference only at a target class of a given DNN model, regardless of the accuracy variation and other outputs’ changes. Therefore, their evaluation results vary depending on the dataset used. On top of AOPC, after the heatmap images were compressed, their file sizes were compared to figure out which XAI algorithms highlight the relevant regions and not more. The file sizes of LRP heatmaps were also smaller than saliency map and deconvolution, however, the assumption that better XAI schemes should have smaller file sizes was also human-centric since the

![FIGURE 1. AOPC evaluation based on the iterative MoRF process. $f(x)$ is the output of a DNN model over the input image $x$.](image-url)
explanation with more scattering points was considered to be worse like SmoothGrad.

\[
AOPC = \left( \frac{1}{L} \sum_{k=1}^{L} f(x_{\text{MoRF}}^{(0)}) - f(x_{\text{MoRF}}^{(k)}) \right)
\]  

The inside-total relevance ratio \( \mu \) \cite{55} was introduced based on the hypothesis that pixels in the object area are more important for the model’s decisions. Therefore, \( \mu \) was obtained as the ratio of inside-total positive relevance \( (R_{in}) \) in the object area and total positive relevance \( (R_{tot}) \) as presented in (3). Additionally, the dependency on the object size was avoided by multiplying \( \mu \) with the ratio of total area \( (S_{tot}) \) and object area \( (S_{in}) \) resulting in \( \mu_w \) as described in (4). Because this method used only positive relevance values as well as the assumption that the model’s decision should rely on the information in the object area, it was also human-centric. This assumption of object-centricity was verified to some extent by investigating the output changes for the occlusion of object and context areas, however, their evaluation results also showed that some features in the image context influenced the decision. The object-centricity is what the human desires for the model, but may not be what the decision actually is based on.

\[
\mu = \frac{R_{in}}{R_{tot}} 
\]

\[
\mu_w = \mu \cdot \frac{S_{tot}}{S_{in}} 
\]

III. PROPOSED HEATMAP ASSISTED ACCURACY SCORE

The proposed heatmap assisted accuracy score (HAAS) scheme has four advantages over previous quantitative evaluation methods. First, HAAS is much simpler than AOPC and inside-total relevance ratio schemes since it does not require any iterations for sequential occlusion or information removal. Second, HAAS directly provides the accuracy ratio as the quantitative evaluation metric while previous methods focus on the variation of only the target output and need additional processes such as averaging, plotting, generating bounding boxes, summing relevance, and estimating area. Because the accuracy is determined by the maximum of all the output units, HAAS takes into account all the output variations. Third, HAAS is able to evaluate both positive and negative influences of features by taking into account all the heatmap scores directly, whereas previous schemes consider only the positive influences. Lastly, HAAS is a machine-centric method because XAI algorithms is evaluated directly based on the performance of the given model, but not from people’s perspectives. Their explainability is investigated by the accuracy changes in the given model using images modified according to features’ importance. In addition, whereas APOC results vary regarding to datasets, HAAS shows the robust evaluation.

A heatmap is one of visualization methods to show the importance of each feature over a given decision in color or gray scale. As depicted in Fig. 2 that was extracted by LRP over a CNN of the handwritten number classification, red and blue colors represent features of positive and negative influences, respectively. Green colors indicate that those features have little impacts on the output. In other words, while the data of red areas contribute to the increase of a specific output, blue regions lead to its decrease.

Our hypothesis for HAAS is that since heatmaps include the information about the features’ influence, the performance of the model should be improved by modifying input images according to heatmap scores. The more accurate explanation the heatmap provides with respect to a network model, the more the performance will be enhanced. Based on this hypothesis, the proposed HAAS scheme is illustrated in Fig. 3. \( f(\cdot) \) is the model, \( x \) is the input image, \( h \) is the heatmap, \( HA(x, h) \) is the HA image for \( x \) and \( h \), and \( Acc(h) \) is the accuracy over \( N \) images. After the decision of a classification model is interpreted in a shape of a heatmap by a given XAI algorithm, the input images are modified by the resultant scores assigned to pixels. The modified images are referred to HA images here. Those HA images are put into the model again and the explainability of the given XAI algorithm is evaluated by the performance improvement metric, \( HAAS \), that is computed by a ratio of accuracies over original and HA images as described in (5). When \( HAAS \) is larger than 1.0, the HA images improve the accuracy of the classification model, leading to the conclusion that heatmap scores extracted by the XAI algorithm are explaining the feature’s importance well. Conversely, less \( HAAS \) than 1.0 means that HA images have deteriorated the performance due to the incorrectly extracted explanation.

\[
HAAS = \frac{Acc(f(HA(x, h)))_N}{Acc(f(x))_N} 
\]
HA images ($HA$) are composed by clipping the product of $x_{\text{Norm}}$ and $1 + h_{\text{Norm}}$ within $-1$ and $+1$ as presented in (6). Finally, the proposed HAAS method measures the accuracy variation of the output decisions obtained after putting HA images into the classification model again.

$$HA = \max\{-1, \min\{1, x_{\text{Norm}} \cdot (1 + h_{\text{Norm}})\}\} \quad (6)$$

### IV. EVALUATION RESULTS

#### A. EVALUATION SETUP

The evaluation is conducted for two CNN models of LeNet-5 [56] and VGG-16 [57] with four datasets of MNIST [58], CIFAR-10 [59], STL-10 [60], and ILSVRC2012 [61] that are famous ones in the classification application. While LeNet-5 network for MNIST and VGG-16 networks for CIFAR-10 and STL-10 are re-trained with training datasets normalized at the range of $-1$ to $+1$, a pre-trained VGG-16 network is employed for ILSVRC2012. As presented in Fig. 4, the last Softmax functions are omitted during the XAI evaluation because the polarity information at the output should be used to express both positive and negative influences of features. However, during the training phase, all the networks have included Softmax functions for class outputs.

XAI algorithms used for the evaluation are saliency map, deconvolution, and LRPs with various rule configurations as investigated in the previous AOPC paper [54] because the performance of HAAS is compared to AOPC’s results. These XAIs bring out the explanation for each feature that is a pixel in an image, but not the object localization. Saliency maps [38] are constructed by (7), where $M_{p,c}$ is the magnitude of the sensitivity value for the input feature ($x_p$) at a pixel $p$ and the target class $c$, and $f_c(x)$ is the model output at the class $c$ for the input image $x$. Therefore, $M_{p,c}$ does not include the information about the polarity of the influence on the output. The resultant heatmap scores of the saliency map ($h_{p,c}^{SM}$) are obtained by normalizing $M_{p,c}$ with the maximum value as shown in (8).

$$M_{p,c} = \left| \frac{\partial f_c(x)}{\partial x_p} \right| \quad (7)$$

$$h_{p,c}^{SM} = \frac{M_{p,c}}{\max[M_c]} \quad (8)$$

When the model consists of convolutional layers ($Conv$) as expressed in (9), deconvolution methods [40] give rise to the explanation by deconvolutional layers ($Deconv$) without ReLU activation functions as described in (10). $a_l^{(i)}$ is the activation output of the $(l)$-th layer, $z_l^{(l+1)}$ is the output of the $(l+1)$-th convolutional layer, $\theta_l^{(l+1)}$ is the parameter of the $(l+1)$-th convolutional layer, and $D_l^{(i)}$ is the backward signal in the $(l)$-th layer for the target class $c$. The parameters of $Deconv$ are set to be equal to $\theta$ of $Conv$. The final heatmap score ($h_{p,c}^{DC}$) of the deconvolution method at the pixel $p$ are achieved by normalizing $D_l^{(i)}$ with its maximum absolute value as shown in (11).

$$z_l^{(l+1)} = Conv(a_l^{(i)}, \theta_l^{(l+1)}) \quad (9)$$

$$D_l^{(i)} = Deconv(R_c(z_l^{(l+1)}), \theta_l^{(l+1)}) \quad (10)$$

$$h_{p,c}^{DC} = \frac{D_l^{(i)}}{\max(|D_l^{(i)}|)} \quad (11)$$

LRPs [43] are also investigated for 5 propagation rules of ε-rule, $\alpha_1$/$\beta_0$-rule, $\alpha_2$/$\beta_1$-rule, flat-rule, and bounded-rule. First of all, $z_{l+1}^{(i,j)}$ defined in (12) as the product of $\alpha_l$ and the parameter $w_{ij}^{(l+1)}$ between $i$-th input and $j$-th output of the $(l+1)$-th convolutional layer. The sum of all these products for the $j$-th output with a bias $b_{j}^{(l+1)}$ is $z_j^{(l+1)}$ like (13).

$$z_{l+1}^{(i,j)} = \alpha_l \cdot w_{ij}^{(l+1)} \quad (12)$$

$$z_j^{(l+1)} = z_{l+1}^{(i,j)} + b_j^{(l+1)} \quad (13)$$
\[
\begin{align*}
\z^{(l+1)}_{ij} &= a^{(l)}_i \cdot w^{(l+1)}_{ij} \\
\z^{(l+1)}_{j} &= \sum_i \z^{(l+1)}_{ij} + b^{(l+1)}_j
\end{align*}
\]

Then, the \( \epsilon \)-rule generates the relevance \( (R^{\epsilon}_{i \leftarrow j}) \) propagated from the \( j \)-th output to the \( i \)-th input as described in (14) with a stabilizer \( (\epsilon) \) to avoid too small denominator close to zero.

\[
R^{\epsilon}_{i \leftarrow j} = \frac{z^{(l+1)}_{ij}}{z^{(l+1)}_j + \epsilon \cdot \text{sign}(z^{(l+1)}_j)} \cdot R^{(l+1)}_j
\]

\( \alpha \beta \)-rules emphasize the positive components since \( \alpha \) is set to be larger than \( \beta \) based on \( \alpha - \beta = 1 \). \( z^{+, (l+1)}_{ij} \) and \( z^{-, (l+1)}_{ij} \) are positive and negative components of \( z^{(l+1)}_{ij} \) as shown in (15) and (16). \( b^{+, (l+1)}_j \) and \( b^{-, (l+1)}_j \) are positive and negative biases as shown in (17) and (18).

\[
\begin{align*}
R^{+, (l+1)}_{i \leftarrow j} &= \begin{cases} 
z^{(l+1)}_{ij}, & \text{if } z^{(l+1)}_{ij} > 0 \\
0, & \text{otherwise.}
\end{cases} \\
R^{-, (l+1)}_{i \leftarrow j} &= \begin{cases} 
z^{(l+1)}_{ij}, & \text{if } z^{(l+1)}_{ij} < 0 \\
0, & \text{otherwise.}
\end{cases}
\end{align*}
\]

\[
\begin{align*}
b^{+, (l+1)}_j &= \begin{cases} 
b^{(l+1)}_j, & \text{if } b^{(l+1)}_j > 0 \\
0, & \text{otherwise.}
\end{cases} \\
b^{-, (l+1)}_j &= \begin{cases} 
b^{(l+1)}_j, & \text{if } b^{(l+1)}_j < 0 \\
0, & \text{otherwise.}
\end{cases}
\end{align*}
\]

With \( z^{+, (l+1)}_j \) and \( z^{-, (l+1)}_j \) obtained by the sums of positive and negative parts as expressed in (19) and (20), \( \alpha \beta \)-rules are described in (21). For \( \alpha \beta \)-rules are described in (21). For \( \alpha_1 \beta_0 \) and \( \alpha_2 \beta_1 \), the numbers following \( \alpha \) and \( \beta \) are their assigned numbers respectively.

\[
z^{+, (l+1)}_j = \sum_i z^{+, (l+1)}_{ij} + b^{+, (l+1)}_j
\]
layers are addressed only by a $\epsilon$ for internal convolutional layers. All fully-connected (FC) rules as summarized in Table 1.

The flat-rule is simply defined to uniformly distribute the relevance of a higher layer to a lower layer as presented in (22) without any consideration on activation outputs and parameters. $\sum_j 1$ is the total number of inputs connected to the $j$-th output of the $(\ell + 1)$-th layer.

$$R^B_{\alpha \beta}((l+1)) = \left( \frac{1}{\sum_i 1} \right) \cdot R^B_j((l+1))$$ (22)

Lastly, the bounded-rule is presented with $z_{ij}^B((l+1))$ defined in (25). $w_{ij}^{+,(l+1)}$ and $w_{ij}^{-,(l+1)}$ are positive and negative parameters as expressed in (23) and (24). $low_i$ is the minimum value of features connected to $w_{ij}^{+,(l+1)}$, and $high_i$ is the maximum value of features linked to $w_{ij}^{-,(l+1)}$. Therefore, it is guaranteed that $z_{ij}^B((l+1))$ is not negative. The resultant propagated relevance of the bounded-rule ($R^B_{\alpha \beta}((l+1))$) with a stabilizer of $\epsilon$ is shown in (26).

$$w_{ij}^{+,(l+1)} = \begin{cases} w_{ij}^{(l+1)}, & \text{if } w_{ij}^{(l+1)} > 0 \\ 0, & \text{otherwise.} \end{cases}$$ (23)

$$w_{ij}^{-,(l+1)} = \begin{cases} w_{ij}^{(l+1)}, & \text{if } w_{ij}^{(l+1)} < 0 \\ 0, & \text{otherwise.} \end{cases}$$ (24)

$$z_{ij}^B((l+1)) = z_{ij}^{(l+1)} - low_i w_{ij}^{+,(l+1)} - high_i w_{ij}^{-,(l+1)}$$ (25)

$$R^B_{\alpha \beta}(i,(l+1)) = \frac{z_{ij}^B((l+1))}{\sum_j z_{ij}^B((l+1))} \cdot R^B_j((l+1))$$ (26)

Consequently, the relevance ($R^B_i(l)$) at the $i$-th input is defined as the sum of all relevance values propagated into the $i$-th input like (27) and LRP heatmap scores ($h_{p,c}^{LPR}$) are obtained by the normalized relevance at the input layer of $l = 0$ as presented in (28).

$$R^B_i(l) = \sum_j R^B_{\alpha \beta}(i,j,(l+1))$$ (27)

$$h_{p,c}^{LPR} = \frac{R^{(0)}_{p,c}}{\max\{|R^{(0)}_{c}|\}}$$ (28)

While $\epsilon$-rule, flat-rule, and bounded-rule are considered for the input layer, $\epsilon$-rule, $\alpha 1\beta 0$-rule, and $\alpha 2\beta 1$-rule are applied for internal convolutional layers. All fully-connected (FC) layers are addressed only by a $\epsilon$-rule. Consequently, this evaluation is composed of totally 9 combinations of LRP rules as summarized in Table 1.

### B. HAAS EVALUATION

All the networks are investigated by both AOPC and HAAS methods for eleven XAI algorithms. AOPC is conducted for 100 iterations per an image with regions of $3 \times 3$. First, LeNet-5 is investigated with the MNIST test dataset of 10,000 images as shown in Table 2, where the mean value of the outputs ($f_c(x)$) at the label classes and accuracy for the original images are 16.06 and 99.07 %, respectively. The HAAS results show the similar trend to the AOPC, where most LRP configurations lead to the better explanations than saliency map and deconvolution. HAAS values of all LRPs are larger than 1.0, which means HA images of LRPs generate higher accuracies than that for the original images. LRP1, that employs only $\epsilon$-rules for all layers, achieves the best HAAS performance of 1.0088 and LRP3 of bounded-rule for the input layer and $\epsilon$-rule for Conv and FC layers takes the second place of 1.0079. Heatmap and HA images over 10 original test images leading to the misclassification are illustrated in Figs. 5 and 6 along with their decisions and label classes’ outputs. The misclassification cases are marked in red and the correct decision cases are presented in blue. According to this result, the increased output level at the label class is not always linked to the accuracy improvement because the further increase at the other class output may cause misclassification. For example, in the case of the test image of 8, LRP7 with the larger output level of 7.19 provides the wrong decision of 9 while LRP8 with the smaller output level of 5.76 results in the correct decision of 8. Because the accuracy is directly related to the decision, the investigation of the accuracy is more important than the estimation of the output level to understand what the decision is made with a basis on.

Second, VGG-16 is investigated with the CIFAR-10 test dataset of 10,000 images as shown in Table 3, where the mean value of $f_c(x)$ and accuracy for the original images are 16.38 and 88.05 %, respectively. Unlike the MNIST dataset of gray images, the HAAS results over CIFAR-10 of color images are different from AOPC results. Whereas HAAS of the saliency map is larger than the deconvolution, their AOPC values show the opposite result to each other. Additionally, the maximum HAAS of 1.1254 is achieved at LRP3, but the maximum AOPC of 543.73 is obtained from LRP7. For CIFAR-10, HAAS values of all LRPs are also bigger than 1.0. LRP3 of bounded-rule for the input layer

| Combination | Input Layer | Conv Layers | FC Layers |
|-------------|-------------|-------------|-----------|
| 1           | $\epsilon$  | $\epsilon$  | $\epsilon$ |
| 2           | $\epsilon$  | $\alpha 1\beta 0$ | $\epsilon$ |
| 3           | $\epsilon$  | $\alpha 2\beta 1$ | $\epsilon$ |
| 4           | flat        | $\epsilon$  | $\epsilon$ |
| 5           | flat        | $\alpha 1\beta 0$ | $\epsilon$ |
| 6           | flat        | $\alpha 2\beta 1$ | $\epsilon$ |
| 7           | bounded     | $\epsilon$  | $\epsilon$ |
| 8           | bounded     | $\alpha 1\beta 0$ | $\epsilon$ |
| 9           | bounded     | $\alpha 2\beta 1$ | $\epsilon$ |
and $\epsilon$-rule for Conv and FC layers achieves the best HAAS performance and LRP1 of only $\epsilon$-rules takes the second place of 1.1160. Heatmap and HA images over 10 original test images leading to misclassification are illustrated in Figs. 7 and 8. For convolutional layers, $\alpha\beta$-rules present the areas of objects in the heatmap image better with respect to the localization, but $\epsilon$-rules give rise to the larger increase in both output and accuracy. Therefore, it is sure that $\epsilon$-rules for all convolutional layers represent the feature importance of each pixel better.

Third, VGG-16 is investigated with the STL-10 test dataset of 8,000 images as shown in Table 4, where the mean value of $f(c)(x)$ and accuracy for the original images are 21.38 and 91.35 %, respectively. The HAAS results over STL-10 are similar to them of CIFAR-10. The maximum HAAS of 1.0918 is accomplished at LRP3 while the maximum AOPC of 346.21 is achieved at LRP5. LRP1 of only $\epsilon$-rules also takes the second place of 1.0906. Heatmap and HA images over 10 original test images leading to misclassification are illustrated in Figs. 9 and 10. Like the HAAS evaluation results at CIFAR-10, the object areas are better marked in the heatmap images with $\alpha\beta$-rules in convolutional layers, however, accuracy and output are more enhanced with $\epsilon$-rules.

Lastly, VGG-16 is investigated with the ILSVRC2012 test dataset of 50,000 images as shown in Table 5, where the mean value of $f(c)(x)$, top-1 accuracy, and top-5 accuracy for the original images are 15.46, 63.41 %, and 85.82 %, respectively. The top-1 and top-5 HAAS results achieve the maximum HAAS values of 1.3469 and 1.1453 similarly to...
FIGURE 6. Heatmap and HA images at MNIST images of 5 to 9 digits for 11 XAI algorithms of saliency map (S. Map), deconvolution (Deconv), and LRP1 to LRP9. Red and blue colors represent wrong and correct classifications, respectively.

them of CIFAR-10 and STL-10. The maximum AOPC of 2232.26 is achieved at LRP7. LRP1 of only $\epsilon$-rules also takes the second place of 1.3207 and 1.1391 for top-1 and top-5 HAAS values. Heatmap and HA images over 10 original test images leading to misclassification are illustrated in Figs. 11 and 12. The overall HAAS evaluation results are equivalent to CIFAR-10 and STL-10.

While the best XAI algorithm selected from AOPC is strongly dependent on the dataset, the proposed HAAS shows the consistent recommendation of LRP1 and LRP3 regardless of datasets. We believe that if the evaluation method is machine-centric, the XAI selection should be independent of datasets. For the quantitative comparison for the consistencies of AOPC and HAAS, we employ Kullback–Leibler divergence ($D_{KL}$) [62] as described in (29), where $p$ and $q$ are two probability distributions and $i$ is the index of possible outcomes. The more similar two distributions are, the closer $D_{KL}$ is to 0. $D_{KL}$ is not negative. If the evaluation method is consistent over datasets, the distributions of results over XAI algorithms must be similar leading to the smaller $D_{KL}$ between different datasets. The evaluation results ($E_k(i)$) are normalized into $N_k(i)$ by (30), where $i = 1, 2, \cdots, 11$ is the index of eleven XAI algorithms from saliency map to LRP9 and $k = 1, 2, 3, 4$ is the index of four datasets from MNIST to ILSVRC2012. Because the probability of 0 must be avoided for $D_{KL}$ computation, the small stabilizer ($\gamma$) of 0.001 is added to the normalized values. After their values are modified into the probability ($P_k(i)$) by dividing with their total sum like (31), divergence values of $D_{KL}(P_1|P_2)$, $D_{KL}(P_1|P_3)$, $D_{KL}(P_1|P_4)$,
Figure 7. Heatmap and HA images at CIFAR-10 images of plane, car, bird, cat, and deer for 11 XAI algorithms of saliency map (S. Map), deconvolution (Deconv), and LRP1 to LRP9. Red and blue colors represent wrong and correct classifications, respectively.

\[ D_{KL}(P_2|P_4), D_{KL}(P_3|P_4), \text{ and } D_{KL}(P_3|P_4) \] are calculated for six combinations of (MNIST, CIFAR-10), (MNIST, STL-10), (MNIST, ILSVRC2012), (CIFAR-10, STL-10), (CIFAR-10, ILSVRC2012), and (STL-10, ILSVRC2012).

\[
D_{KL}(p|q) = \sum_i p(i) \log \frac{p(i)}{q(i)} \tag{29}
\]

\[
N_k(i) = \frac{E_k(i) - \min(E_k)}{\max(E_k) - \min(E_k)} + \gamma \tag{30}
\]

\[
P_k(i) = \frac{N_k(i)}{\sum_i N_k(i)} \tag{31}
\]

Whereas AOPC results in \(D_{KL}\) values of 0.2761, 0.5337, 0.0329, 0.3685, 0.4289, and 0.1888 with the average of 0.3048, HAAS obtains divergence values of 0.0119, 0.0335, 0.0654, 0.0071, 0.0260, and 0.0067 which average is 0.0251 that is smaller than AOPC by a factor of 10. It is ensured that the distribution of HAAS evaluation is less dependent of datasets than AOPC. Therefore, we conclude that HAAS has established more robust evaluation for XAI algorithms. Our evaluation results lead to the conclusion that \(\epsilon\)-rule and bounded-rule for an input layer and \(\epsilon\)-rule for convolutional and FC layers are most suitable to achieve the machine-centric robust evaluation of XAIs over DNNs.

C. INVERTED HA TEST

To further verify our hypothesis of HAAS that emphasizing the pixels of the image according to heatmap scores enhances the accuracy of a given classification network, the inverted
HA images \((\text{invHA})\) are generated by inverted heatmap scores as described in (32) and Fig. 13 and put into the network to estimate the change of the accuracy, compared to the HA images generated with the original heatmap scores. If the proposed HAAS scheme is reflecting the feature importance properly, the inverted HA images must lead to larger accuracy degradation at the XAI algorithms of the higher HAAS results.

\[
\text{invHA} = \max\{-1, \min\{1, x_{\text{Norm}} \cdot \{1 - h_{\text{Norm}}\}\}\} \quad (32)
\]

Like the HAAS evaluation, the inverted HA test is also conducted with two DNNs of LeNet-5 and VGG-16, and four test datasets of MNIST, CIFAR-10, STL-10, and ILSVRC2012. The output value at the label class \(f_c(\text{invHA}(x, h))\), accuracy at the inverted HA images, and the accuracy difference \((\Delta\text{Accuracy})\) between inverted HA and original HA images are measured as summarized in Tables 6, 7, 8, and 9. As expected, their results ensure that LRP1 and LRP3 with largest HAAS values experience larger accuracy losses for inverted HA images such as -13.82 % and -10.86 % in MNIST, -81.92 % and -81.33 % in CIFAR-10, -80.00 % and -73.98 % in STL-10, and \((-78.19 \%, -83.83 \%)\) and \((-78.78 \%, -82.53 \%)\) in (Top-1, Top-5) of ILSVRC2012, compared to other XAI algorithms. As a conclusion, the hypothesis of the proposed HAAS scheme is reasonable, where emphasizing features according to heatmap scores affect the accuracy of DNN-based classifier. The HAAS evaluation gives rise to the machine-centric...
V. CONCLUSION

Whereas DNNs have accomplished dramatic performance improvement in many areas, understanding the inside of the networks becomes more difficult. Consequently, various XAI algorithms have been come up with to interpret the logic behind decisions of complex neural networks, and at the same time, it has been necessary to study methods to evaluate their explainability. This paper proposes a machine-centric HAAS scheme to evaluate the explainability of XAIs while most previous evaluation methods focus on how closer their results are to what people rely on. Furthermore, unlike AOPC needing many iterations and focusing only on the output of a target class, HAAS provides the quantitative scores directly by putting HA images generated from original image and heatmap scores into the given model without any iterations. To estimate the accuracy changes, it should take into account all the outputs at the same time. In addition, HAAS uses both positive and negative influences of features. Especially, over four datasets for classification networks, HAAS achieves lower $D_{KL}$ of 0.0251 than AOPC of 0.3048 in average. Low $D_{KL}$ means that HAAS provides more robust evaluation than AOPC independently of datasets. The evaluation through the accuracy changes of the network and the high consistency of the best XAI selection lead to the conclusion that HAAS is the machine-centric evaluation. Although the proposed HAAS has been investigated only for the classification way that can explain how well the given XAI algorithm actually addresses each feature importance.
networks, we are sure that our machine-centric evaluation scheme can pave the efficient way to further improvement on performance as well as explainability in a variety of DNNs.

REFERENCES

[1] A. Levant, “ImageNet: A large-scale hierarchical image database,” in Proc. 2009 IEEE CVPR, Miami, FL, USA, 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5209648.

[2] X.-W. Chen and X. Lin, “Big Data Deep Learning: Challenges and Perspectives,” IEEE Access, vol. 2, pp. 514–525, 2014. DOI: 10.1109/ACCESS.2014.2325029.

[3] G. Nguyen, S. Dlugolinsk, M. Bobak, V. Tran, A. L. Garcia, I. Heredia, P. Malik, and L. Hluchy, “Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey,” Artif. Intell. Rev., vol. 52, pp. 77–124, 2019. DOI: 10.1007/s10462-018-9097-9.

[4] A. Cano, “A survey on graphic processing unit computing for large-scale data mining,” WIREs Data Min. Knowl. Discov., vol. 8, Art. No. e1232, 2017. DOI: 10.1002/widm.1232.

[5] N. P. Jouppi et al., “In-Datacenter Performance Analysis of a Tensor Processing Unit,” in Proc. ACM/IEEE ISCA, Toronto, ON, Canada, 2017, pp. 1–12. DOI: 10.1145/3079856.3080246.

[6] Intel AI Hardware. Intel Corporation. [Online]. Available: https://www.intel.com/content/www/us/en/artificial-intelligence/hardware.html

[7] S. Tang, AI-Chip: A list of ICs and IPs for AI, Machine Learning and Deep Learning, [Online]. Available: https://github.com/basicmi/AI-Chip

[8] J. Kobielus, Powering AI: The explosion of new AI hardware accelerators, [Online]. Available: https://www.infoworld.com/article/3290104/powering-ai-the-explosion-of-new-ai-hardware-accelerators.html

[9] A. Reuther, P. Michaleas, M. Jones, V. Gadepally, S. Samsi and J. Kepner, “Survey of Machine Learning Accelerators,” in Proc. IEEE HPEC, Waltham, MA, USA, 2020, pp. 1–12. DOI: 10.1109/HPEC43674.2020.9266449.

[10] Y. Huang, X. Ma, X. Fan, J. Liu, and W. Gong, “When deep learning meets edge computing,” in Proc. IEEE ICNP, Toronto, ON, Canada, 2017. DOI: 10.1109/ICNP.2017.8117585.
FIGURE 11. Heatmap and HA images at ILSVRC2012 images of kite, crane, boxer, beaver, and baseball for 11 XAI algorithms of saliency map (S. Map), deconvolution (Deconv), and LRP1 to LRP9. Red and blue colors represent wrong and correct classifications, respectively.

11] A. E. Eshratifar, A. Esmaili, and M. Pedram, “BottleNet: A Deep Learning Architecture for Intelligent Mobile Cloud Computing Services,” in Proc. IEEE/ACM ISLPED, Lausanne, Switzerland, 2019. DOI: 10.1109/ISLPED.2019.8824955.

12] H. Yan, P. Yu, and D. Long, “Study on Deep Unsupervised Learning Optimization Algorithm Based on Cloud Computing,” in Proc. ICTBS, Changsha, China, 2019, pp. 679–681. DOI: 10.1109/ICTBS.2019.00168.

13] F. Jauro, H. Chiroma, A. Y. Gital, M. Almutairi, S. M. Abdulhamid, and J. H. Abawajy, “Deep learning architectures in emerging cloud computing architectures: Recent development, challenges and next research trend,” Appl. Soft Comput., vol. 96, Art. No. 106582, 2020. DOI: 10.1016/j.asoc.2020.106582.

14] M. W. Gardner and S. R. Dorling, “Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences,” Atmos. Environ., vol. 32, pp. 2627–2636, 1998. DOI: 10.1016/S1352-2310(97)00447-0.

15] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” in Proc. ICET, Antalya, Turkey, 2017. DOI: 10.1109/ICEntech.2017.8308186.

16] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Comput., vol. 9, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735.

17] A. Shcherbinsky, “Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network,” Phys. D: Nonlinear Phenom., vol. 404, Art. No. 132306, 2020. DOI: 10.1016/j.physd.2019.132306.

18] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, “Generative Adversarial Networks: An Overview,” IEEE Signal Process. Mag., vol. 35, pp. 53–65, 2018. DOI: 10.1109/MSP.2017.2765202.

19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, and B. Xu, “Generative adversarial networks,” Commun. ACM, vol. 63, pp. 139–144, 2020. DOI: 10.1145/3422622.

20] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, “A Comprehensive Survey on Graph Neural Networks,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, pp. 4–24, 2020. DOI: 10.1109/TNNLS.2020.2978386.

21] Y. Li, H. Zhang, X. Xue, Y. Jiang, and Q. Shen, “Deep learning for remote sensing image classification: A survey,” WIREs Data Min. Knowl. Discov., vol. 8, Art. No. e1264, 2018. DOI: 10.1002/widm.1264.

22] J. Zhou, J. X. Huang, Q. Chen, Q. V. Hu, T. Wang, and L. He, “Deep Learning for Aspect-Level Sentiment Classification: Survey, Vision, and Challenges,” IEEE Access, vol. 7, pp. 78454–78483, 2019. DOI: 10.1109/ACCESS.2019.2920075.
FIGURE 12. Heatmap and HA images at ILSVRC2012 images of cradle, mask, scoreboard, teapot, and cup for 11 XAI algorithms of saliency map (S. Map), deconvolution (Deconv), and LRP1 to LRP9. Red and blue colors represent wrong and correct classifications, respectively.

[23] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikaninen, “Deep Learning for Generic Object Detection: A Survey,” Int. J. Comput. Vis., vol. 128, pp. 261–318, 2020. DOI: 10.1007/s11263-019-01247-4.

[24] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, “A survey on deep learning techniques for image and video semantic segmentation,” Appl. Soft Comput., vol. 70, pp. 41–65, 2018. DOI: 10.1016/j.asoc.2018.05.018.

[25] Z. Wang, J. Chen, and S. C. H. Hoi, “Deep Learning for Image Super-Resolution: A Survey,” IEEE Trans. Patterns Anal. Mach. Intell., vol. 43, pp. 3365–3387, 2021. DOI: 10.1109/TPAMI.2020.2982166.

[26] P. Basmatkar, H. Holani, and S. Kaushal, “Survey on Neural Machine Translation for multilingual translation system,” in Proc. ICCMC, Erode, India, 2019. DOI:10.1109/ICC2019.9819788.

[27] D. W. Otter, J. R. Medina, and J. K. Kalita, “A Survey of the Usages of Deep Learning for Natural Language Processing,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, pp. 604–624, 2021. DOI:10.1109/TNNLS.2020.2979670.

[28] L. Wang, W. Chen, W. Yang, F. Bi, and F. R. Yu, “A State-of-the-Art Review on Image Synthesis With Generative Adversarial Networks,” IEEE Access, vol. 8, pp. 63514–63537, 2020. DOI: 10.1109/ACCESS.2020.2982224.

[29] S. Amirian, K. Rasheed, T. R. Taha, and H. R. Arabnia, “Automatic Image and Video Caption Generation With Deep Learning: A Concise Review and Algorithmic Overlap,” IEEE Access, vol. 8, pp. 218386–218400, 2020. DOI: 10.1109/ACCESS.2020.3042484.

[30] A. Wali, Z. Alamgir, S. Karim, A. Fawaz, M. B. Ali, M. Adan, and M. Mujtaba, “Generative adversarial networks for speech processing: A review,” IEEE Access, vol. 72, Art. No. 101308, 2022. DOI: 10.1016/j.csl.2021.101308.

[31] A. M. Ozbayoglu, M. U. Gudelek, and O. B. Sezer, “Deep learning for financial applications: A survey,” Appl. Soft Comput., vol. 93, Art. No. 106384, 2020. DOI: 10.1016/j.asoc.2020.106384.

[32] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, “Neural Style Transfer: A Review,” IEEE Trans. Vis. Comput. Graph., vol. 11, pp. 3365–3385, 2020. DOI: 10.1109/TVCG.2019.2921336.

[33] H. Nam, Y. I. Kim, J. Bae, and J. Lee, “GateRL: Automated Circuit Design Framework of CMOS Logic Gates Using Reinforcement Learning,” Elec-
TABLE 2. HAAS Evaluation Results of Saliency Map (S. Map), Deconvolution (Deconv), and 9 LRP Configurations for MNIST. The biggest values are marked in bold and the second results are with underlines.

| XAs       | Mean $f_c(h(x, A))$ | HAAS | AOPC |
|-----------|---------------------|------|------|
| S. Map    | 16.61               | 0.9998 | 642.08 |
| Deconv    | 14.31               | 0.9958 | 543.24 |
| LRP1      | **18.86**           | 1.0008 | **719.06** |
| LRP2      | 17.25               | 1.0071 | 646.26 |
| LRP3      | 17.90               | 1.0079 | 703.76 |
| LRP4      | 17.10               | 1.0084 | 685.73 |
| LRP5      | 16.76               | 1.0042 | 680.83 |
| LRP6      | 16.80               | 1.0042 | 659.16 |
| LRP7      | 18.20               | 1.0077 | 700.45 |
| LRP8      | 16.97               | 1.0058 | 627.45 |
| LRP9      | 17.4                | 1.0068 | 676.32 |
| Original  | Mean $f_c(x)$ = 16.06, Accuracy = 99.07% |

TABLE 3. HAAS Evaluation Results of Saliency Map (S. Map), Deconvolution (Deconv), and 9 LRP Configurations for CIFAR-10. The biggest values are marked in bold and the second results are with underlines.

| XAs       | Mean $f_c(h(x, A))$ | HAAS | AOPC |
|-----------|---------------------|------|------|
| S. Map    | 15.40               | 1.0079 | 509.52 |
| Deconv    | 14.18               | 0.8736 | 529.45 |
| LRP1      | 30.24               | 1.1160 | 514.62 |
| LRP2      | 24.56               | 1.1026 | 445.03 |
| LRP3      | **32.52**           | **1.1254** | 474.92 |
| LRP4      | 17.63               | 1.0020 | 534.80 |
| LRP5      | 18.07               | 1.0002 | 481.37 |
| LRP6      | 18.05               | 1.0048 | 495.31 |
| LRP7      | 18.80               | 1.0311 | **543.73** |
| LRP8      | 18.26               | 1.0154 | 496.50 |
| LRP9      | 18.37               | 1.0241 | 511.23 |
| Original  | Mean $f_c(x)$ = 16.38, Accuracy = 88.05% |

TABLE 4. HAAS Evaluation Results of Saliency Map (S. Map), Deconvolution (Deconv), and 9 LRP Configurations for STL-10. The biggest values are marked in bold and the second results are with underlines.

| XAs       | Mean $f_c(h(x, A))$ | HAAS | AOPC |
|-----------|---------------------|------|------|
| S. Map    | 21.02               | 0.9910 | 281.47 |
| Deconv    | 17.91               | 0.9388 | 289.68 |
| LRP1      | 35.07               | 1.0906 | 268.90 |
| LRP2      | 28.89               | 1.0793 | 275.74 |
| LRP3      | **36.08**           | **1.0918** | 277.97 |
| LRP4      | 22.20               | 0.9974 | 339.99 |
| LRP5      | 22.65               | 0.9968 | **346.21** |
| LRP6      | 22.65               | 1.0001 | 344.77 |
| LRP7      | 22.82               | 0.9985 | 345.78 |
| LRP8      | 22.49               | 0.9985 | 346.17 |
| LRP9      | 22.58               | 1.0018 | 343.87 |
| Original  | Mean $f_c(x)$ = 21.38, Accuracy = 91.35% |

TABLE 5. HAAS Evaluation Results of Saliency Map (S. Map), Deconvolution (Deconv), and 9 LRP Configurations for ILSVRC2012. The biggest values are marked in bold and the second results are with underlines.

| XAs       | Mean $f_c(h(x, A))$ | HAAS (Top-1) | HAAS (Top-5) | AOPC |
|-----------|---------------------|---------------|---------------|------|
| S. Map    | 15.51               | 0.9904 | 0.9913 | 1802.85 |
| Deconv    | 13.55               | 0.8843 | 0.9357 | 1505.22 |
| LRP1      | 27.32               | 1.3207 | 1.1391 | 1907.97 |
| LRP2      | 21.93               | 1.2355 | 1.1088 | 2190.78 |
| LRP3      | **28.69**           | **1.3469** | **1.1453** | 2154.92 |
| LRP4      | 15.04               | 0.9986 | 0.9960 | 2186.83 |
| LRP5      | 15.32               | 1.0035 | 1.0015 | 2139.14 |
| LRP6      | 15.40               | 1.0113 | 1.0056 | 2158.05 |
| LRP7      | 15.33               | 1.0055 | 1.0025 | **2232.26** |
| LRP8      | 15.27               | 1.0008 | 1.0012 | 2210.08 |
| LRP9      | 15.40               | 1.0082 | 1.0042 | 2225.37 |
| Original  | Mean $f_c(x)$ = 15.46, Accuracy = 63.41 % (Top-1), 85.82 % (Top-5) |
FIGURE 13. Heatmap and HA images at ILSVRC2012 for 11 XAI algorithms of saliency map (S. Map), deconvolution (Deconv), and LRP1 to LRP9.

TABLE 6. Inverted HA Test Results for MNIST. The values of LRP1 are marked in bold and the ones of LRP3 are with underlines.

| XAIs      | Mean $f_t$ (invHA(x, h)) | Accuracy (%) | $\Delta$Accuracy (%) |
|-----------|--------------------------|--------------|----------------------|
| S. Map    | 9.89                     | 97.59        | -1.46                |
| Deconv    | 12.65                    | 98.66        | 0.01                 |
| LRP1      | 7.6                      | 86.12        | -13.82               |
| LRP2      | 7.54                     | 88.84        | -10.93               |
| LRP3      | 7.98                     | 88.99        | -10.86               |
| LRP4      | 8.72                     | 93.57        | -6.13                |
| LRP5      | 7.52                     | 92.97        | -6.52                |
| LRP6      | 8.78                     | 94.97        | -4.52                |
| LRP7      | 8.50                     | 89.93        | -9.9                 |
| LRP8      | 8.27                     | 93.07        | -6.57                |
| LRP9      | 8.87                     | 93.11        | -6.63                |

TABLE 7. Inverted HA Test Results for CIFAR-10. The values of LRP1 are marked in bold and the ones of LRP3 are with underlines.

| XAIs      | Mean $f_t$ (invHA(x, h)) | Accuracy (%) | $\Delta$Accuracy (%) |
|-----------|--------------------------|--------------|----------------------|
| S. Map    | 12.88                    | 80.37        | -5.86                |
| Deconv    | 13.94                    | 77.47        | 0.55                 |
| LRP1      | 1.23                     | 16.34        | -81.92               |
| LRP2      | 5.81                     | 44.98        | -52.10               |
| LRP3      | 1.40                     | 17.76        | -81.33               |
| LRP4      | 11.71                    | 79.43        | -8.80                |
| LRP5      | 9.52                     | 72.67        | -15.40               |
| LRP6      | 10.46                    | 75.88        | -12.59               |
| LRP7      | 11.05                    | 75.26        | -15.53               |
| LRP8      | 10.94                    | 76.69        | -12.72               |
| LRP9      | 11.34                    | 77.53        | -12.64               |

TABLE 8. Inverted HA Test Results for STL-10. The values of LRP1 are marked in bold and the ones of LRP3 are with underlines.

| XAIs      | Mean $f_t$ (invHA(x, h)) | Accuracy (%) | $\Delta$Accuracy (%) |
|-----------|--------------------------|--------------|----------------------|
| S. Map    | 17.33                    | 87.15        | -3.38                |
| Deconv    | 17.99                    | 87.59        | 1.83                 |
| LRP1      | 2.84                     | 19.63        | -80.00               |
| LRP2      | 11.54                    | 57.46        | -4.13                |
| LRP3      | 4.26                     | 25.76        | -73.98               |
| LRP4      | 17.40                    | 84.16        | -6.90                |
| LRP5      | 16.00                    | 85.08        | -6.28                |
| LRP6      | 17.11                    | 85.08        | -5.26                |
| LRP7      | 17.64                    | 86.85        | -4.36                |
| LRP8      | 17.83                    | 87.26        | -4.25                |

TABLE 9. Inverted HA Test Results for ILSVRC2012. The values of LRP1 are marked in bold and the ones of LRP3 are with underlines.

| XAIs      | Mean $f_t$ (invHA(x, h)) | Accuracy (%) | $\Delta$Accuracy (%) |
|-----------|--------------------------|--------------|----------------------|
| S. Map    | 14.13                    | 57.48        | -81.10               |
| Deconv    | 13.68                    | 56.22        | 79.98                |
| LRP1      | 2.86                     | 5.69         | 13.93                |
| LRP2      | 7.99                     | 5.69         | -78.19               |
| LRP3      | 3.02                     | 6.76         | -78.57               |
| LRP4      | 13.99                    | 57.29        | -8.10                |
| LRP5      | 13.69                    | 56.09        | 80.04                |
| LRP6      | 13.82                    | 56.56        | 80.41                |
| LRP7      | 14.07                    | 58.55        | 81.99                |
| LRP8      | 14.40                    | 59.71        | 82.85                |
| LRP9      | 14.42                    | 59.64        | 82.83                |

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
JUNHEE LEE received the B.S. degree from the Department of Information Display, Kyung Hee University, Seoul, South Korea, in 2020, where he is currently pursuing the M.S. degree. His current research is focused on display circuit and deep learning.

HYEONSEONG CHO received the B.S. degree from the Department of Information Display, Kyung Hee University, Seoul, South Korea, in 2021, where he is currently pursuing the M.S. degree. His current research is focused on generative networks based on deep learning.

YUN JANG PYUN received the B.S. degree from the Department of Information Display, Kyung Hee University, Seoul, South Korea, in 2021, where he is currently pursuing the M.S. degree. His current research is focused on image processing networks based on deep learning.

SUK-JU KANG received the B.S. degree in electronic engineering from Sogang University, Republic of Korea, in 2006, and the Ph.D. degree in electrical and computer engineering from the Pohang University of Science and Technology in 2011. From 2011 to 2012, he was a Senior Researcher with LG Display, where he was a project leader for resolution enhancement and multi-view 3D system projects. From 2012 to 2015, he was an Assistant Professor of Electrical Engineering with Dong-A University, Busan. He is currently an Associate Professor of Electronic Engineering with Sogang University. His current research interests include image analysis and enhancement, video processing, and multimedia signals.

HYOUNGSIK NAM (M’10) received his B.S., M.S. and Ph.D. degrees in EECS from Korea Advance Institute of Science and Technology (KAIST), Daejon Korea, in 1996, 1998, and 2004. He joined Samsung Electronics as a senior engineer in 2005, where he had worked on Active-Matrix Liquid-Crystal Displays. He is currently an associate professor in the department of Information Display at Kyung Hee University, Seoul Korea. His current research interests are low power technologies, integrated circuits, signal/user interfaces for flat panel displays, and machine learning application.