SHAP-CAM: VISUAL EXPLANATIONS FOR CONVOLUTIONAL NEURAL NETWORKS BASED ON SHAPLEY VALUE

Anonymous authors
Paper under double-blind review

ABSTRACT

Explaining deep convolutional neural networks has been recently drawing increasing attention since it helps to understand the networks' internal operations and why they make certain decisions. Saliency maps, which emphasize salient regions largely connected to the network's decision-making, are one of the most common ways for visualizing and analyzing deep networks in the computer vision community. However, saliency maps generated by existing methods cannot represent authentic information in images due to the unproven proposals about the weights of activation maps which lack solid theoretical foundation and fail to consider the relations between each pixels. In this paper, we develop a novel post-hoc visual explanation method called Shap-CAM based on class activation mapping. Unlike previous class activation mapping based approaches, Shap-CAM gets rid of the dependence on gradients by obtaining the importance of each pixels through Shapley value. We demonstrate that Shap-CAM achieves better visual performance and fairness for interpreting the decision making process. Our approach outperforms previous methods on both recognition and localization tasks.

1 INTRODUCTION

The sensational advance of machine learning within the form of deep neural networkss has opened up modern Artificial Intelligence (AI) capabilities in real-world applications. Deep learning models achieve impressive results in tasks like object detection, speech recognition, machine translation, which offer tremendous benefits. However, the connectionist approach of deep learning is fundamentally different from earlier AI systems where the predominant reasoning methods are logical and symbolic. These early systems can generate a trace of their inference steps, which at that point serves as the basis for explanation. On the other hand, the usability of today's intelligent systems is limited by the failure to explain their decisions to human users. This issue is particularly critical for risk-sensitive applications such as security, clinical decision support or autonomous navigation.

For this gap, various methods have been proposed by researchers over the last few years to figure out what knowledge is hidden in the layers and connections when utilizing deep learning models. For example, one category of methods rationalize the decision of a model by training another deep model which comes up with explanations why the model behaved the way it did. Another approach was to examine the neural network models of the black box by trying to intelligently modify the input and analyze the model's response to it. While encouraging development has been carrying this field forward, existing efforts are restricted and the goal of explainable deep learning still has a long way to go, given the difficulty and wide range of issue scopes.

In the context of understanding Convolutional Neural Networks (CNNs), Zeiler & Fergus made one of the first efforts in understanding what a CNN learns (Zeiler & Fergus, 2014). However, their method necessitates extensive computations to generate this explanation. Using a new technique called CAM (Class Activation Mapping), Zhou et al. followed up on the same goal and demonstrated that various levels of the CNN functioned as unsupervised object detectors (Zhou et al., 2016). They were able to obtain heat maps that illustrate which regions of an input image were looked at by the CNN for assigning a label by employing a global average pooling layer and showing the weighted combination of the resulting feature maps at the penultimate (pre-softmax) layer.

However, this technique was architecture-sensitive and involved retraining a linear classifier for each class. Similar methods were examined with different pooling layers such as global max pooling and log-sum-exp pooling (Oquab et al., 2015; Pinheiro & Collobert, 2015). After that, Selvaraju et al. developed Grad-CAM, an efficient version of CAM that combines the class-conditional property of CAM with current pixel-space gradient visualization techniques like Guided Back-propagation and Deconvolution to emphasize fine-grained elements on the image (Selvaraju et al., 2016). Grad-CAM improved the transparency of CNN-based models by displaying input regions with high resolution details that are critical for prediction. Grad-CAM++, one of the variations of Grad-CAM, introduce pixel-wise weighting of the gradients of the output w.r.t. a particular spatial position in the final convolutional feature map of the CNN (Chattopadhay et al., 2018). In Grad-CAM++, more reliable expressions for the pixel-wise weights were derived. However, gradient-based CAM methods cannot represent authentic information in images due to the unproven proposals about the weights of activation maps.

To this end, Wang et al. proposed Score-CAM which got rid of the dependence on gradients by obtaining the weight of each activation map through its forward passing score on target class (Wang et al., 2020). Though Score-CAM discarded gradients for generating explanations, it still suffered from self designed expression of score which lacked solid theoretical foundation and failed to take the relationship between pixels into consideration. In this work, we present a new post-hoc visual explanation method, named Shap-CAM, where the importance of pixels is derived from their marginal contribution to the model output utilizing Shapley value. Our contributions are:

- We propose a novel gradient-free visual explanation method, Shap-CAM, which introduce Shapley value in the cooperative game theory to estimate the marginal contribution of pixels. Due to the superiority of Shapley value and the consideration of relationship between pixels, more rational and accurate contribution of each pixel is obtained.
- We quantitatively evaluate the generated saliency maps of Shap-CAM on recognition and localization tasks and show that Shap-CAM better discovers important features.
- We show that in a constrained teacher-student setting, it is possible to achieve an improvement in the performance of the student by using a specific loss function inspired from the explanation maps generated by Shap-CAM, which indicates that our explanations discover authentic semantic information mined in images.

The remainder of the paper is organized as follows. In Section 2, we introduce the related work about visual explanations and Shapley value. In Section 3, we develop our Shap-CAM for the generation of visual explanations based on Shapley value. In Section 4, we present some experimental results on recognition and localization tasks and show the effectiveness of our proposed method. We finish the paper with final conclusions and remarks.

2 Related Work

2.1 Visual Explanations

We give a summary of related attempts in recent years to understand CNN predictions in this part. Zeiler et al. provided one of the earliest initiatives in this field, developing a deconvolution approach to better grasp what the higher layers of a given network have learned (Zeiler & Fergus, 2014). Springenberg et al. extended this work to guided backpropagation, which allowed them to better comprehend the impact of each neuron in a deep network on the input image (Springenberg et al., 2014). From a different perspective, Ribeiro et al. introduced LIME (Local Interpretable Model-Agnostic Explanations), an approach that uses smaller interpretable classifiers like sparse linear models or shallow decision trees to make a local approximation to the complex decision surface of any deep model (Ribeiro et al., 2016). Shrikumar et al. presented DeepLift, which approximates the instantaneous gradients (of the output with respect to the inputs) with discrete gradients to determine the relevance of each input neuron for a given decision (Shrikumar et al., 2017). Al-Shedivat et al. presented Contextual Explanation Networks (CENs), a class of models that learns to anticipate and explain its decision simultaneously (Al-Shedivat et al., 2017). Unlike other posthoc model-explanation tools, CENs combine deep networks with context-specific probabilistic models to create explanations in the form of locally-correct hypotheses.

Class Activation Mapping (CAM) (Zhou et al., 2016) is a technique for discovering discriminative regions and giving understandable explanations of deep models across domains. In CAM, the authors demonstrate that a CNN with a Global Average Pooling (GAP) layer after the last convolutional layer shows localization capabilities despite not being explicitly trained to do so. The CAM explanation regards the importance of each channel as the weight of fully connected layer connecting the global average pooling and the output probability distribution. However, an obvious limitation of CAM is the requirements of a GAP penultimate layer and retraining of an additional fully connected layer. To resolve this problem, Grad-CAM (Selvaraju et al., 2016) extends the CAM explanation and regards the importance of each channel as the gradient of class confidence w.r.t. the activation map. In Grad-CAM, the authors naturally regard gradients as the importance of each channel towards the class probability, which avoids any retraining or model modification. Variations of Grad-CAM, like Grad-CAM++ (Chattopadhay et al., 2018), use different combinations of gradients and revise the weights for adapting the explanations to different conditions.

However, gradient-based CAM methods do not have solid theoretical foundation and receive poor performances when the gradients are not reliable. Adversarial model manipulation methods fool the explanations by manipulating the gradients without noticeable modifications to the original images (Heo et al., 2019), proving that the gradient-based CAM methods are not robust and reliable enough. Score-CAM gets rid of the dependence of gradients and introduces channel-wise increase of confidence as the importance of each channel (Wang et al., 2020). Score-CAM obtains the weight of each activation map through its forward passing score on target class, the final result is obtained by a linear combination of weights and activation maps. This approach however suffers from self designed expression of score which lacked solid theoretical foundation. Besides, it fails to consider the relationship between different pixels.

2.2 SHAPLEY VALUE

One of the most important solution concepts in cooperative games was defined by Shapley (Shapley, 1953). This solution concept is now known as the Shapley value. The Shapley value is useful when there exists a need to allocate the worth that a set of players can achieve if they agree to cooperate. Although the Shapley value has been widely studied from a theoretical point of view, the problem of its calculation still exists. In fact, it can be proved that the problem of computing the Shapley value is an NP-complete problem (Deng & Papadimitriou, 1994).

Several authors have been trying to find algorithms to calculate the Shapley value precisely for particular classes of games. In Bilbao et al. for example, where a special class of voting game is examined, theoretical antimatriod concepts are used to polynomially compute the Shapley value (Fernández et al., 2002). In Granot et al. a polynomial algorithm is developed for a special case of an operation research game (Granot et al., 2002). In Castro et al., it is proved that the Shapley value for an airport game can be computed in polynomial time by taking into account that this value is obtained using the serial cost sharing rule (Castro et al., 2008).

Considering the wide application of game theory to real world problems, where exact solutions are often not possible, a need exists to develop algorithms that facilitate this approximation. Although the multilinear extension defined by Owen is an exact method for simple games (Owen, 1972), the calculation of the corresponding integral is not a trivial task. So, when this integral is approximated (using the central limit theorem) this methodology could be considered as an approximation method. In Fatima et al. , a randomized polynomial method for determining the approximate Shapley value is presented for voting games (Fatima et al., 2006). Castro et al. develop an efficient algorithm that can estimate the Shapley value for a large class of games (Castro et al., 2009). They use sampling to estimate the Shapley value and any semivalues. These estimations are efficient if the worth of any coalition can be calculated in polynomial time.

In this work, we propose a new post-hoc visual explanation method, named Shap-CAM, where the importance of pixels is derived from their marginal contribution to the model output utilizing Shapley value. Due to the superiority of Shapley value and the consideration of relationship between pixels, more rational and accurate explanations are obtained.

3 APPROACH

In this section, we first present the prelimilaries of visual explanations and the background the CAM methods. Then we introduce the Shapley Value (Shapley, 1953), a way to quantify the marginal contribution of each player in the cooperative game theory. We apply this theory to our problem and propose the definition of Shap-CAM. Finally, we clarify the estimation of Shapley value in our method.

3.1 PRELIMILARIES

Class Activation Mapping (CAM) (Zhou et al., 2016) is a technique for discovering discriminative regions and giving understandable explanations of deep CNN across domains. Let function $\boldsymbol{Y} = f(\boldsymbol{X})$ be a CNN which takes $\boldsymbol{X}$ as an input data point and outputs a probability distribution $\boldsymbol{Y}$. We denote Y^c as the probability of class c. For the last convolutional layer, $\boldsymbol{A}^k$ denotes the feature map of the k-th channel.

In CAM, the authors demonstrate that a CNN with a Global Average Pooling (GAP) layer after the last convolutional layer shows localization capabilities despite not being explicitly trained to do so.However, an obvious limitation of CAM is the requirements of a GAP penultimate layer and retraining of an additional fully connected layer. To resolve this problem, Grad-CAM (Selvaraju et al., 2016) extends the CAM explanation and regards the importance of each channel as the gradient of class confidence $\boldsymbol{Y}$ w.r.t. the activation map $\boldsymbol{A}$, which is defined as:

$$L^c_{ij,\ Grad-CAM} = ReLU\Big(\sum_k w^c_k A^k_{ij}\Big) \tag{1}$$

where

$$w^c_k = \frac{1}{Z}\sum_i\sum_j \frac{\partial Y^c}{\partial A^k_{ij}} \tag{2}$$

Constant Z stands for the number of pixels in the activation map. In Grad-CAM, the authors naturally regard gradients as the importance of each channel towards the class probability, which avoids any retraining or model modification. Variations of Grad-CAM, like Grad-CAM++ (Chattopadhay et al., 2018), use different combinations of gradients and revise w^c_k in Equation 1 for adapting the explanations to different conditions.

However, gradient-based CAM methods do not have solid theoretical foundation and receive poor performances when the gradients are not reliable. Adversarial model manipulation methods (Heo et al., 2019) fool the explanations by manipulating the gradients without noticeable modifications to the original images, proving that the gradient-based CAM methods are not robust and reliable enough. Score-CAM (Wang et al., 2020) gets rid of the dependence of gradients and introduces channel-wise increase of confidence as the importance of each channel. This approach however suffers from self designed expression of score which lacked solid theoretical foundation. Besides, it fails to consider the relationship between different pixels.

3.2 DEFINITION OF SHAPCAM

In order to obtain more accurate and rational estimation of the marginal contribution of each pixel to the model output, we turn to the cooperative game theory. One of the most important solution concepts in cooperative games was defined by Shapley (Shapley, 1953). This solution concept is now known as the Shapley value. The Shapley value is useful when there exists a need to allocate the worth that a set of players can achieve if they agree to cooperate. Consider a set of n players $\mathbb{P}$ and a function $f(\mathbb{S})$ which represents the worth of the subset s of players $\mathbb{S} \subseteq \mathbb{P}$. The function $f : \mathbb{P} \to \mathbb{R}$ maps each subset to a real number. Shapley Value is one way to quantify the marginal contribution of each player to the result $f(\mathbb{P})$ of the game when all players participate. For a given player i, its Shapley value can be computed as:

$$Sh_i(f) = \sum_{\mathbb{S}\subseteq\mathbb{P}, i\notin\mathbb{S}} \frac{(n-s-1)!s!}{n!}\big[f(\mathbb{S}\cup\{i\}) - f(\mathbb{S})\big], \quad i = 1,\ldots,n \tag{3}$$

The Shapley value for player i defined above can be interpreted as the average marginal contribution of player i to all possible coalitions $\mathbb{S}$ that can be formed without it.

Notably, it can be proved that Shapley value is the only way of assigning attributions to players that satisfies the following four properties:

- **Null player.** If the class probability does not depend on any pixels, then its attribution should always be zero.
- **Symmetry.** If the class probability depends on two pixels but not on their order (i.e. the values of the two pixels could be swapped, never affecting the probability), then the two pixels receive the same attribution. This property, also called anonymity , is arguably a desirable property for any attribution method: if two players play the exact same role in the game, they should receive the same attribution.
- **Linearity.** If the function f can be seen as a linear combination of the functions of two sub-networks (i.e. $f = af_1 + bf_2$), then any attribution should also be a linear combination, with the same weights, of the attributions computed on the sub-networks, i.e. $Sh_i(\boldsymbol{x}|f) = a{\cdot}Sh_i(\boldsymbol{x}|f_1){+}b{\cdot}Sh_i(\boldsymbol{x}|f_2)$. Intuitively, this is justified by the need for preserving linearities within the network.
- **Efficiency.** An attribution method satisfies efficiency when attributions sum up to the difference between the value of the function evaluated at the input, and the value of the function evaluated at the baseline, i.e. $\sum_{i=1}^{n} Sh_i = \Delta f = f(\boldsymbol{x}) - f(\mathbf{0})$. In our problem, this property indicates that all the attributions of the pixels sum up to the difference between the output probability of the original feature map and the output of the feature map where no original pixels remain. This property, also called completeness or conservation, has been recognized by previous works as desirable to ensure the attribution method is comprehensive in its accounting. If the difference $\Delta f > 0$, there must exist some pixels assigned a non-zero attribution, which is not necessarily true for gradient-based methods.

Back to our problem on class activation mapping, we consider each pixel (i, j) in the feature map of the last convolutional layer $\boldsymbol{A}$ as a player in the cooperative game. Let $\mathbb{P} = \{(i, j)|i = 1, \ldots, h; j = 1, \ldots, w\}$ be the set of pixels in the feature map $\boldsymbol{A}$, where h, w stand for the height and width of the feature map. Let $n = h \cdot w$ be the number of pixels in the activation map. We then define the worth function f in Equation 3 as the class confidence Y^c, where c is the class of interest. For each subset $\mathbb{S} \subseteq \mathbb{P}$, $Y^c(\mathbb{S})$ represents the output probability of class c when only the pixels in the set $\mathbb{S}$ remain. By the symbolization above, the original problem turns to an n-player game $(\mathbb{P}, Y^c)$. Naturally, the Shapley Value of the pixel (i, j) represents its marginal contribution to the class confidence. Thus, we define the Shapley Value as the saliency map of our Shap-CAM:

$$\begin{aligned} L^c_{ij,\ Shap-CAM} &= Sh_{(i,j)}(Y^c) \\ &= \sum_{\mathbb{S}\subseteq\mathbb{P},(i,j)\notin\mathbb{S}} \frac{(n-s-1)!s!}{n!}\big[Y^c(\mathbb{S}\cup\{(i,j)\}) - Y^c(\mathbb{S})\big] \end{aligned} \tag{4}$$

The contribution formula that uniquely satisfies all these properties is that a component's contribution is its marginal contribution to the performance of every subnetwork of the original model (normalized by the number of subnetworks with the same cardinality). Most importantly, this formula takes into account the interactions between neurons. As a simple example, suppose there are two neurons that improve performance only if they are both present or absent and harm performance if only one is present. The equation considers all these possible settings. This, to our knowledge, is one of the few methods that take such interactions into account and is inspired by similar approaches in Game Theory.

Shapley value was introduced as an equitable way of sharing the group reward among the players where equitable means satisfying the aforementioned properties. It's possible to make a direct mapping between our setting and a cooperative game; therefore, proving the uniqueness of Shap-CAM.

3.3 Estimation of Shapley Value

Exactly computing Equation 3 would require $\mathcal{O}(2^n)$ evaluations. Intuitively, this is required to evaluate the contribution of each activation with respect to all possible subsets that can be enumerated

with the other ones. Clearly, the exact computation of Shapley values is computationally unfeasible for real problems. Sampling is a process or method of drawing a representative group of individuals or cases from a particular population. Sampling and statistical inference are used in circumstances in which it is impractical to obtain information from every member of the population. Taking this into account, we use sampling in this paper to estimate the Shapley value and any semivalues. These estimations are efficient if the worth of any coalition can be calculated in polynomial time. Here we use a sampling algorithm to estimate Shapley value which reduces the complexity to $\mathcal{O}(Kn)$, where K is the number of samples taken (Castro et al., 2009).

Following the definition of Shapley value in Equation 3, an alternative definition of the Shapley value can be expressed in terms of all possible orders of the players. Let $O : \{1, \ldots, n\} \to \{1, \ldots, n\}$ be a permutation that assigns to each position k the player $O(k)$. Let us denote by $\pi(\mathbb{P})$ the set of all possible permutations with player set $\mathbb{P}$. Given a permutation O, we denote by $Pre^i(O)$ the set of predecessors of the player i in the order O, i.e. $Pre^i(O) = \{O(1), \ldots, O(k-1)\}$, if $i = O(k)$.

Thus, the Shapley value can be expressed equivalently in the following way:

$$Sh_i(f) = \frac{1}{n!} \sum_{O \in \pi(\mathbb{P})} \big[f(Pre^i(O) \cup \{i\}) - f(Pre^i(O)) \big], \quad i = 1, \ldots, n \tag{5}$$

In estimation, we randomly take K samples of player order O from $\pi(\mathbb{P})$, calculate the marginal contribution of the players in the order O, which is defined in the summation of Equation 5, and finally average the marginal contributions as the approximation.

Then we will obtain, in polynomial time, an estimation of the Shapley value with some desirable properties. To estimate the Shapley value, we will use a unique sampling process for all players. The sampling process is defined as follows:

(1) The population of the sampling process P will be the set of all possible orders of N players.

(2) The vector parameter under study is $Sh = (Sh_1, \ldots, Sh_n)$.

(3) The characteristics observed in each sampling unit are the marginal contributions of the players in the order O, i.e.

$$\begin{aligned} \chi(O) &= (\chi(O)_1, \ldots, \chi(O)_n) \\ where \quad \chi(O)_i &= f(Pre^i(O) \cup \{i\}) - f(Pre^i(O)) \end{aligned} \tag{6}$$

(4) The estimate of the parameter will be the mean of the marginal contributions over the sample M.

4 Experiments

In this section, we conduct experiments to evaluate the effectiveness of the proposed explanation method. First, we assess the fairness of the explanation (the significance of the highlighted region for the model's decision) on image recognition in Section 4.1. In Section 4.2 we show the effectiveness for class-conditional localization of objects in a given image. The knowledge distillation experiment is followed in Section 4.3.

In the following experiments, unless stated otherwise, we use pre-trained VGG16 network (Simonyan & Zisserman, 2014) from the Pytorch model zoo as a base model. Publicly available object classification dataset, namely, ILSVRC2012 val (Russakovsky et al., 2015) is used in our experiment. For the input images, we resize them to (224 × 224 × 3), transform them to the range [0, 1], and then normalize them using mean vector [0.485, 0.456, 0.406] and standard deviation vector [0.229, 0.224, 0.225]. For simplicity, baseline image X_b is set to 0.

4.1 Faithfulness Evaluation via Image Recognition

The faithfulness evaluations are carried out as depicted in Grad-CAM++ (Chattopadhay et al., 2018) for the purpose of object recognition. The original input is masked by point-wise multiplication with the saliency maps to observe the score change on the target class. In this experiment, rather than do point-wise multiplication with the original generated saliency map, we slightly modify by limiting

Table 1: Recognition evaluation results on the ImageNet (ILSVRC2012) validation set (lower is better in Average Drop, higher is better in Average Increase).

Method	Mask	RISE	GradCAM	GradCAM++	ScoreCAM	ShapCAM
Average Drop(%)	63.5	47.0	47.8	45.5	31.5	**28.0**
Average Increase(%)	5.29	14.0	19.6	18.9	30.6	**31.8**

Table 2: Recognition evaluation results on the PASCAL VOC 2007 validation set (lower is better in Average Drop, higher is better in Average Increase).

Method	GradCAM	GradCAM++	ScoreCAM	ShapCAM
Average Drop(%)	28.5	19.5	15.6	**13.2**
Average Increase(%)	21.4	19.0	28.9	**32.7**

the number of positive pixels in the saliency map. Two metrics called Average Drop, Average Increase In Confidence are introduced:

- Average Drop %: The Average Drop refers to the maximum positive difference in the predictions made by the prediction using the input image and the prediction using the saliency map. It is given as: $\sum_{i=1}^{N} \frac{\max(0, Y_i^c - O_i^c)}{Y_i^c} \times 100$. Here, Y_i^c refers to the prediction score on class c using the input image i and O_i^c refers to the prediction score on class c using the saliency map produced over the input image i.
- Increase in Confidence %: In this metric, we measure the number of times in the entire dataset, the model's confidence increased when providing only the explanation map regions as input. The Average Increase in Confidence is denoted as:$\sum_{i=1}^{N} \frac{sign(Y_i^c < O_i^c)}{N} \times 100$. $sign$ presents an indicator function that returns 1 if input is True.

Our comparison extends with state-of-the-art methods, namely gradient-based, perturbation-based and CAM-based methods, including Mask (Fong & Vedaldi, 2017), RISE (Petsiuk et al., 2018), Grad-CAM (Selvaraju et al., 2016) and Grad-CAM++ (Chattopadhay et al., 2018). Experiment conducts on the ImageNet (ILSVRC2012) validation set, 2000 images are randomly selected. Results are reported in Table 1. Results on the PASCAL VOC 2007 validation set are reported in Table 2.

As shown in Table 1 and Table 2, Shap-CAM outperforms other perturbation-based and CAM-based methods. Shap-CAM can successfully locate the most distinguishable part of the target item, rather than only determining what humans think is important, based on its performance on the recognition challenge. Results on the recognition task show that Shap-CAM can more accurately reveal the decision-making process of the original CNN model than earlier techniques.

4.2 Localization Evaluation

Bounding box evaluations are accomplished in this section. We employ the similar metric, as specified in Score-CAM, called the Energy-based pointing game. Here, the amount of energy of the saliency map is calculated by finding out how much of the saliency map falls inside the bounding box. Specifically, the input is binarized with the interior of the bounding box marked as 1 and the region outside the bounding box as 0. Then, this input is multiplied with the generated saliency map and summed over to calculate the proportion ratio, which given as $Proportion = \frac{\sum L^c_{(i,j)\in bbox}}{\sum L^c_{(i,j)\in bbox} + \sum L^c_{(i,j)\notin bbox}}$. Two pre-trained models, namely VGG-16 (Simonyan & Zisserman, 2014), ResNet18 (He et al., 2016), are used to conduct the energy-based pointing game on the 2000 randomly chosen images from the ILSVRC 2012 Validation set.

As we observe, the object comprises the majority of the image region in the ILSVRC validation set, making these images unsuitable for assessing the localization capabilities of the generated saliency

Table 3: Localization Evaluations of Proportion (%) using Energy-based Pointing Game (Higher the better).

Method	GradCAM	GradCAM++	ScoreCAM	ShapCAM
VGG-16	39.95	40.16	40.10	**40.45**
ResNet18	40.90	40.85	40.76	**41.28**

Table 4: Results for knowledge distillation to train a student from a deeper teacher network.

Loss function used	Test error rate	
	w/o L_{KD}	w/ L_{KD}
L_{cross_ent}	6.78	5.68
$L_{exp\_student}$(Grad-CAM)	6.86	5.80
$L_{exp\_student}$(Grad-CAM++)	6.74	5.56
$L_{exp\_student}$(Score-CAM)	6.75	5.42
$L_{exp\_student}$(Shap-CAM)	**6.69**	**5.37**

maps. Therefore, we randomly select images from the validation set by removing images where object occupies more than 50% of the whole image. For convenience, we only consider these images with only one bounding box for target class. We experiment on 500 random selected images from the ILSVRC 2012 validation set. Evaluation result is reported in Table 3, which shows that our method outperforms previous works. This also confirms that the Shap-CAM-generated saliency map has fewer noises.

4.3 Learning from Explanations: Knowledge Distillation

Following the knowledge distillation experiment settings in Grad-CAM++ (Chattopadhay et al., 2018), we show that in a constrained teacher-student learning setting, knowledge transfer to a shallow student (commonly called knowledge distillation) is possible from the explanation of CNN decisions generated by CAM methods. Wide ResNets (Zagoruyko & Komodakis, 2016) for both the student and teacher networks. We train a WRN-40-2 teacher network (2.2 M parameters) on the CIFAR-10 dataset. In order to train a student WRN- 16-2 network (0.7 M parameters), we introduce a modified loss $L_{exp\_student}$, which is a weighted combination of the standard cross entropy loss L_{cross_ent} and an interpretability loss $L_{interpret}$.

Table 4 shows the results for this experiment. L_{cross_ent} is the normal cross entropy loss function, i.e. the student network is trained independently on the dataset without any intervention from the expert teacher. We further also included L_{KD}, the knowledge distillation loss introduced by Hinton et al. with temperature parameter set to 4 (Hinton et al., 2015). These results show that Shap-CAM provides better explanation-based knowledge distillation than existing CAM-based methods.

5 Conclusion

We propose Shap-CAM, a novel CAM variant, for visual explanations of deep convolutional networks. We introduce Shapley value to represent the marginal contribution of each pixel to the model output. Due to the superiority of Shapley value and the consideration of relationship between pixels, more rational and accurate explanations are obtained. We present evaluations of the generated saliency maps on recognition and localization tasks and show that Shap-CAM better discovers important features. In a constrained teacher-student setting, our Shap-CAM provides better explanation-based knowledge distillation than the state-of-the-art explanation approaches.

References

M. Al-Shedivat, A. Dubey, and E. P. Xing. Contextual explanation networks. In *arXiv preprint arXiv:1705.10301*, 2017.

J. Castro, D. Gomez, and J. Tejada. A polynomial rule for the problem of sharing delay costs in pert networks. In *Computers and Operation Research*, 2008.

Javier Castro, Daniel Gomez, and Juan Tejada. Polynomial calculation of the shapley value based on sampling. In *Computers & Operations Research*, 2009.

A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *IEEE Winter Conference on Applications of Computer Vision*, pp. 839–847, 2018.

X Deng and CH Papadimitriou. On the complexity of cooperative solution concepts. In *Mathematics of Operations Research*, 1994.

SS. Fatima, M. Wooldridge, and NR. Jenniggs. An analysis of the shapley value and its uncertainty for the voting game. In *Lectures notes in artificial intelligence*, 2006.

J. Fernández, E. Algaba, and J. Bilbao et al. Generating functions for computing the myerson value. In *Annals of Operations Research*, 2002.

R. C. Fong and A. Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *ICCV*, 2017.

D. Granot, J. Kuipers, and S. Chopra. Cost allocation for a tree network with heterogeneous customers. In *Mathematics of Operations Research*, 2002.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Juyeon Heo, Sunghwan Joo, and Taesup Moon. Fooling neural network interpretations via adversarial model manipulation. In *NeurIPS*, 2019.

G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *arXiv preprint arXiv:1503.02531*, 2015.

M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free? weakly-supervised learning with convolutional neural networks. In *CVPR*, pp. 685–694, 2015.

G. Owen. Multilinear extensions of games. In *Management Science Series B–Application*, 1972.

V. Petsiuk, A. Das, and K. Saenko. Rise: Randomized input sampling for explanation of black-box models. In *arXiv preprint arXiv:1806.07421*, 2018.

P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, pp. 1713–1721, 2015.

M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, 2016.

O. Russakovsky, J. Deng, H. Su, J. Krause, and S. Satheesh et al. Imagenet large scale visual recognition challenge. In *International journal of computer vision*, 2015.

R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batrat. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. In *arXiv preprint arXiv:1610.02391*, 2016.

L. S. Shapley. *A value for n-person games*. Contributions to the Theory of Games, 2(28): 307–317, 1953.

A. Shrikumar, P. Greenside, and A. Kundaje. Learning important features through propagating activation differences. In *arXiv preprint arXiv:1704.02685*, 2017.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *arXiv preprint arXiv:1409.1556*, 2014.

J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. In *arXiv preprint arXiv:1412.6806*, 2014.

Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *CVPR*, 2020.

S. Zagoruyko and N. Komodakis. Wide residual networks. In *arXiv preprint arXiv:1605.07146*, 2016.

M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pp. 818–833, 2014.

B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *CVPR*, pp. 2921–2929, 2016.