Feature Purity: A Quantitative Evaluation Metric for Feature Extraction in Convolutional Neural Networks

Wenjun Chen¹, Chao Cong¹*, Liwen Huang¹
¹College of Electrical and Electronic Engineering, Chongqing University of Technology, Chongqing, 400054, China
*Corresponding author’s E-mail: congchao@cqut.edu.cn

Abstract. The quantitative analysis of specific convolutional layer feature information in Convolutional Neural Networks (CNN) has received considerable critical attention in the field of deep learning. In this paper, we proposed an innovative measuring method based on the principle of information entropy to quantitatively analyze the performance of feature extraction in CNN. In the method, the feature purity was defined as the normalized entropy of activation histogram, which was calculated from all kernels in feature layer of CNN. Feature purities were evaluated respectively in different layers of CNN as the inspection for internal structure in specific classification models. At the same time, general degrees of purities among different models were compared with classification performances. The experiments were performed in models of AlexNet, VGG, ResNet, and SENet with different epochs trained by cifar10 and ImageNet1000 datasets. Additionally, a visualization evaluation was performed by grad-CAM method in the manners of the intra-model and inter-model purities. Experimental results showed a strong relationship between feature purity and the classification accuracy in each model. Additionally, the visualized feature map also demonstrated that salient region extracted in specified feature layer had good consistency with purity value. As a result, the proposed feature purity can serve as a quantitative metric representing the degree of feature extraction in CNN without the dependency of label or specific network structure; and has significant interpretability and versatility across models.

1. Introduction
In recent years, deep learning technology has developed rapidly in the field of computer vision, from basic image classification tasks to further applications such as object detection [1], semantic segmentation [2] and so on. Compared with traditional methods based on artificial design of visual features, convolutional neural networks (CNN) has achieved the most advanced performance in these fields. However, the internal mechanism of CNN is always regarded as a ‘black box’, and its poor algorithm interpretability becomes one of the defects that have been criticized. Therefore, a growing number of researchers have realized that the significant value of interpretability in neural networks, and the internal working mechanism of the models has been extensively concerned.

Explainable research of neural network has gone through many stages [21]. Classical interpretable methods include visual interpretation method [3,4,5,6], which interprets neural network by visualizing the feature information of corresponding classes extracted from feature layer. On the other hand, interpretable methods based on models and data are also proposed. LIME [9] explained the neural network by fitting a simple interpretable model (linear classification) locally in the classification model of complex neural networks. Influence function [10] has studied the effects of training samples (e.g.,
reducing one sample, perturbing the sample) on model prediction to explained the neural network. Wang et al [11] has analyzed the relationship between the generalization behavior of convolution neural network and the spectrum of image datasets. the representation ability of the middle layer of convolutional neural network is judged by knowledge consistency [12].

At present, there are still some problems to be solved in the research of interpretability of neural network. The existing work mainly explained the model by looking at a certain classes of corresponding features extracted by CNN, or explains the model from the local or data of the model. The evaluation of global feature information and the criterion of its measurement are helpful to further understand and evaluate neural network model.

Therefore, we proposed a feature measurement method based on the principle of maximum entropy. Our main idea is to measure the global feature information by evaluating the overlap degree of cross-class information extracted by convolution layer. That is, by counting the activation degree of kernels to all classes, we got the activation histogram of kernels, and calculated the information entropy of feature layer to obtained the feature purity to evaluated the global feature information extracted by convolutional layer. The comparison of intra- and inter-model feature purities proved that there is a positive correlation between the purity and the performance of kernels, layers, or models. At the same time, the purity of the feature layer increases gradually with the increasing of training epochs.

Our main contributions are summarized as follows:

Based on the principle of maximum entropy, we proposed a neural network feature measurement method. By experiments, we get the relationship between the feature layers within the intra-model and inter-model, meanwhile the variation trend of feature extraction information under different training degrees are obtained. And through Grad-CAM visual interpretation.

The proposed method can compare the extracted features between different feature layers of the same neural network. At the same time, the metric does not depend on the specific network structure selected and can be used across the network.

2. Related work
Visual interpretation of neural networks included Deconvolution [3] and Guided-backpropagation [4], they has elaborated the concept of features in the shallow and deep layers of CNN. Further, by replacing the full connection layer with the global average pooling and retraining the weight, the CAM [5] can understand the importance of each part of the input image to the category, and then visualize the classification results of the neural network. Selvaraju et al. proposed Grad-CAM [6], that without to replace the full connection layer, the global average of the gradient is used to calculate the weight, and the importance of the category is obtained. They investigated the visual interpretation of a specific class of features of the image, but they cannot understand the global features extracted by the feature layer well. In addition, TCAV [24] used the directional derivatives to measure the sensitivity of model predictions with respect to concepts at any model layer.

Entropy is a commonly used metric to measure the disorder or uncertainty in information theory. At the same time, information entropy is often used as a quantitative index of information content of a system. A larger entropy value means the system contains more information. Moreover, a KSE (kernel sparsity and entropy) [25] method is proposed to quantitate the feature map importance in a feature-agnostic manner to guide model compression. Li et al [26] proposed an entropy-based filter pruning (EFP) method to prunes unimportant filters based on the amount of information carried by their corresponding feature maps.

Different from them, we used the Entropy and proposed a concept of feature purity to quantitative analysis CNN feature extraction performance. In CNN, for a specific convolutional layer, we apply the weights of each channel in Grad-CAM to multiplied the feature map, and obtained the weighted feature map. Combined with it, we calculated the activation of each kernel under threshold for every classes by information entropy, and obtained the feature purity to analyzed the global feature information extracted by convolution layer. At the same time, the proposed feature purity can serve as a quantitative metric
representing the degree of feature extraction in CNN without the dependency of label or specific network structure; and has significant interpretability and versatility across models.

3. Method

3.1 Principle of Feature Purity
In our method, the feature purity was defined as the normalized entropy of activation histogram, which was calculated from all kernels in feature layer of CNN. In CNN, we want that a filter can only learn the features belonging to a specific category as pure as possible, that is, it is activated by a specific category. Corresponding, for a convolutional layer, we hope that the filters activated relatively uniform, with less overlapping of cross-class information expressed by different kernels. Then the feature information extracted by this layer is ‘purier’ and more meaningful, that means the feature purity is higher, as shown in Figure 1(a). The feature purity is relatively small if the filter activated insufficient for classes, as shown in Figure 1(b). In the deeper layers with more filters, we can think that some kernels learn different features of a certain class. That is, the features extracted by convolutional layer are more abundant, which can help the model make better decision. Corresponding, its feature purity is higher, as shown in Figure 1(c).

![Figure 1](image)

Figure 1. Sample of different activation cases in the feature layer. The dashed circles represent that kernels are not activated, and other circles of different colors represent that neurons are fully activated by different classes.

3.2 Algorithm Flow
The steps of feature purity calculation are shown in Figure 2. Same as Grad-CAM, for a given convolutional layer $A \in R^{c\times k\times w}$, we defined the weight $\beta_m^n$ of the $c$ feature map in the convolution layer to the category $m$ as follow:

$$\beta_m^n = \frac{1}{Z} \sum_i \sum_j \frac{\partial y_m}{\partial A_{ij}}$$  

(1)
Where $Z$ is the number of pixels of the feature, $y^m$ is the score of the corresponding category $m$, $A^c_{ij}$ representing the pixel value at the $(i,j)$ position in the $c$ feature map.

After obtaining the weight of the corresponding feature map of the category $m$, the weight is multiplied by the corresponding feature map one by one, and the weighted feature map $L \in \mathbb{R}^{c \times w \times h}$ is obtained:

$$L = P^m_c A^c$$  \hfill (2)

Then we applied global average pooling for it, and obtained feature activation matrix $M \in \mathbb{R}^{1 \times c}$. 

$$M = \text{ReLU} \left( \frac{1}{Z} \sum_i \sum_j L^c_{ij} \right)$$  \hfill (3)

Where $L^c_{ij}$ representing the pixel value at the $(i,j)$ position in the $c$ feature map.

Given $n$ images, we can get its activation matrix $M_{n \times c} \in \mathbb{R}^{n \times c}$. Then, the activation matrix $\hat{M}_{n \times c}$ of $c$ kernels is obtained by setting threshold statistics. Then we can obtain the activation probability of each channel $p_c$.

$$\hat{M}_{n \times c} = f(M_{n \times c})$$  \hfill (4)

$$f(x) = \begin{cases} 1 & x \geq \text{threshold} \\ 0 & x < \text{threshold} \end{cases}$$  \hfill (5)

$$p_c = \frac{\sum_i \sum_j \hat{M}_{ic}}{\sum_i \sum_j \hat{M}_{ij}}$$  \hfill (6)

Then the information entropy of the layer $H$ can be calculated. Ideally, the maximum information entropy activated by the $c$ channels is: $H_{\text{max}} = -\log \frac{1}{c}$. The normalized feature purity score of the corresponding feature layer $F_S$ is obtained:

$$H = -\sum_c p_c \cdot \log \frac{p_c}{Z}$$  \hfill (7)

$$F_S = H / H_{\text{max}}$$  \hfill (8)

Where $F_S$ value is $0 \sim 1$. A large value of $F_S$ means that the activation of all kernels are more uniform, and the feature extracted by kernels are more ‘pure’. Ideally, when the activation probability of each channel $p_c = 1/c$, $F_S$ reached maximum 1. If its value is small, it is suggesting that channels are not fully activated, and the features extracted by kernels are less.
4. Experiment

We used CIFAR10 [13] and ImageNet (ILSVRC-12 [14]) to experiment. On CIFAR10 dataset, we use Tiny VGG [22] and ResNet18 [15] as the baseline model. The simple data augmentation is implemented for training: 4 pixels are padded on each side, and a 32 × 32 crop is randomly sampled from the padded image or its horizontal flip. The learning rate set to be 0.001, and batch size set to be 64. On the ImageNet (ILSVRC-12) datasets, we chose AlexNet [16], VGG [17], DenseNet121 [18], ResNet [16] and SENet154 [19] as experimental models. For the corresponding weight we selected pytorch officially published pre-training weights.

First, we compared the effectiveness of the feature measure under different thresholds to choose the best threshold. Then choose three methods to compared: (1) Intra-model method, which means the comparison of feature purities in different feature layers of the same model. (2) Inter-model method, which means the comparison of feature purities extracted from different models. (3) Comparison of the purities from models with different training epochs. The selection of threshold is carried out on ResNet18 and VGG19 in CIFAR10. Intra-model Comparison chose ResNet18 and VGG19 models on dataset CIFAR10 for experiment. For Inter-model Comparison, we used AlexNet, VGG, DenseNet121, ResNet and SENet154 models on dataset ImageNet (ILSVRC-12) to experiment. For purity comparison at different training epochs, Tiny VGG and ResNet18 were selected on the CIFAR10 for experiment. The corresponding feature layers of a given category are visualized by Grad-CAM and the relationship between feature purity and feature layer training effect is further compared.

4.1 Thresholds in Purity Scores

We chose different thresholds (0.1-0.9) on ResNet18 and VGG19 and compared the purity scores of each feature layer at different thresholds, as shown in Figure 3a and b. Then, we evaluated the effectiveness of their feature measures and selected the best threshold. It can be demonstrated that when the threshold is 0.4-0.5, the purity of each feature layer has significant differentiation, which can distinguish the features extracted by different feature layers. At the same time, we find that with the increase of threshold, the purity of shallow layer decreases more, and the purity of deep layer decreases a little. When the threshold is 0.4-0.5, the shallow purity decreases a lot, which indicates that the activation of shallow layer is weakly, the activation of deep kernels is strong. When the threshold is greater than 0.5, the feature purity begins to decrease significantly, which leading the effectiveness of feature metrics become bad, especially in the shallow feature layer. Therefore, under the condition of...
ensuring the validity of feature measurement and the loss of feature information as much as possible, the threshold value of 0.5 is selected for the experiment.

4.2 Intra-model Comparison

Correspondingly on the ResNet18 model, we chose the images of each classes and observed the features extracted by different feature layers of the same model by Grad-CAM visualization, and compared the feature purity in different conv-layer.

We choose some obviously layers to compared. As shown in Figure. 4, by observing the Grad-CAM of the corresponding layer, it is found that the feature layer with great purity score extracted more feature information, and has a better response to all kinds of activation. For example, the feature purity score of layer4.1.conv2 is 0.971, the Grad-CAM of the corresponding layer shows that the activation response region of the its class is better, the localization ability of each classes is better, and the semantic information extracted by the layer is great. While for layer1.1.conv1, layer1.1.conv2 with a purity score of 0.147,0.394, the corresponding activation regions are scattered, and the localization of the classes is relatively poor, indicating that kernels are not fully activated or activated weak. Among them layer4.1.conv1 there is not activation to birds, but in general, the activation to other classes is relatively good. Generally, in line with the mentioned in the [3], shallow nonlinear ability is weak, learning ability is insufficient, only can extracted the object edge, texture, contour, and other low-level features, so the activation of features is relatively weak, the purity of features is small. With the increasing of layers, nonlinear ability is improved, and the extracted features are high-level semantic information about categories, can learned more information.

![Figure 3](image_url)

Figure 3. ResNet18 and VGG19 characteristic purity of each characteristic layer at different thresholds. L represent layer, C represent the conv-layer, F represent features.
Figure 4. Comparison of Grad-CAM and purity of each class of CIFAR10 datasets in ResNet18 feature layer. The right-side numbers are the feature purity score, and the number of red and green represents the bad value and good value, respectively.

4.3 Inter-model Comparison

For the measurement of different model features, we mainly considered two sides to compared: First, we selected each feature layer that contains low, medium, and high semantic features before each model downsampling, and calculated the average of its feature purity to measure the feature information extracted by the model. Secondly, considering that the deep feature layer contains more high-level semantic information and has a great influence on the result of final classification, the last feature layer of the model is considered to be compared.

On the imageNet (ILSVRC-12) dataset, we used VGG and ResNet to compute the purity of each feature layer before downsampling, and calculated their average value to observed the score of the feature purity of each characteristic layer, As shown in Table 1. It can be demonstrated that the feature purity of the last layer (C5) is higher than other feature layers. With the increase of model depth and performance, the purity was improved. The average value shows different trends with the different models and had a positive correlation with the performance. In contrast, there is a strong correlation between the purity and performance of the final convolutional layer of the model, which has more closely related to the final classification results. Therefore, the C5 layer was chosen to compare the purity of different models.

Table 1 Feature purity of different feature layers in VGG and ResNet, C1-C5 represents the feature layer before each downsampling of the model, the mean represents the average of the five feature layers.

| layer | VGG 13 bn | VGG 16 bn | VGG 19 bn | ResNet34 | ResNet50 | ResNet101 |
|-------|-----------|-----------|-----------|----------|----------|-----------|
| C1    | 0.407     | 0.559     | 0.55      | 0.629    | 0.692    | 0.644     |
| C2    | 0.404     | 0.308     | 0.135     | 0.506    | 0.641    | 0.739     |
| C3    | 0.409     | 0.283     | 0.407     | 0.226    | 0.491    | 0.575     |
| C4    | 0.763     | 0.82      | 0.777     | 0.443    | 0.337    | 0.515     |
| C5    | 0.93      | 0.934     | 0.939     | 0.931    | 0.948    | 0.952     |
| mean  | 0.583     | 0.581     | 0.562     | 0.547    | 0.622    | 0.639     |
Table 2 Comparison of the feature purity and accuracy in different models on imagenet1000 datasets

| Model     | Accuracy | Purity |
|-----------|----------|--------|
| AlexNet   | 56.43%   | 0.512  |
| VGG16     | 71.64%   | 0.793  |
| DenseNet121 | 74.67% | 0.87   |
| ResNet50  | 76.00%   | 0.948  |
| SENet154  | 81.30%   | 0.963  |

Figure 5. Comparison of activation patterns and feature purity of last convolutional layers on imageNet1000 of different models. Red box and green box represent the feature layer activated poor and activated well in Grad-CAM, respectively.

Subsequently, on the imageNet (ILSVRC-12) datasets, we used AlexNet, VGG16, DenseNet121, ResNet50 and SENet154 with different performance to experiments. As shown in Table 2, the purity of the last feature layer of the model with better performance is relatively large. Figure 5 shows the purity and Grad-CAM comparison of each model in imageNet. A red box and a green dotted box are used to mark the classes activated poor and activated well in Grad-CAM, respectively. We observed that there have more green box in model of better performance, at the same time the model feature layer got a high score of feature purity and included less background information in Grad-CAM. On the other hand, the localization of filters to classes is better.

4.4 Purities with different training epochs
Moreover, in order to studied the vary in the purity of models with different training epochs during training, we use Tiny VGG and ResNet18 to compared the purity of feature layers under different training degrees of the same model. The models of training 10, 20, 50 and 100 epochs were selected to compared the purity of the last convolutional layer. Table 3 shows the feature purity scores between ResNet18 model and Tiny VGG model in different training epochs. Corresponding as shown in Figure 6, 7, we use the Grad-CAM to visualized the last feature layer of Tiny VGG and ResNet18 training different epochs and observed the change of the purity of the corresponding feature layers. We use red box and a green box to marked the classes activated poor and well in Grad-CAM, respectively.

The result suggesting that with the increasing of training, the features extracted from Tiny VGG and ResNet18 are gradually optimized, and the extracted semantic feature information is better. The interference of background information is gradually eliminated, and the ability to located classes is better. At the same time, we find that with the increasing of training epochs, model existing the overfitting. Although the purity of the feature layer is improved, the features of some classes extracted by the feature layer gradually become worse and are located at the edge or background of the class. For example, in Figure 7, the activation of car and deer in epoch 50 is better than epoch 100. In epoch 100, due to the
overfitting of the model, the activation of the car and deer are located in the edge of the car and the background of the deer.

Table 3 feature purity scores between ResNet18 model and Tiny VGG model in different training epochs, where Tiny VGG_10 and resnet18_10 represent models after training 10 epochs; Tiny VGG_20, resnet18_20, Tiny VGG_50, resnet18_50, Tiny VGG_100, resnet18_100 and so on.

| Model   | Accuracy | Purity | Model   | Accuracy | Purity |
|---------|----------|--------|---------|----------|--------|
| Tiny VGG_10 | 81.25     | 0.713  | ResNet18_10 | 87.5     | 0.81   |
| Tiny VGG_20 | 81.25     | 0.756  | ResNet18_20 | 87.5     | 0.826  |
| Tiny VGG_50 | 85.5      | 0.823  | ResNet18_50 | 93.75    | 0.887  |
| Tiny VGG_100 | 87.5      | 0.835  | ResNet18_100 | 95.5     | 0.946  |

Figure 6. Compared purity and Grad-CAM of the last feature layer under different epochs of Tiny VGG. Red box and green box represent the feature layer activated poorly and activated well in Grad-CAM, respectively.

Figure 7. Compared purity and Grad-CAM of the last feature layer under different epochs of the ResNet18 model. Red box and green box represent the feature layer activated poorly and activated well in Grad-CAM, respectively.

5. Discussion
In CNN, the accuracy for a single output class is highly dependent on a small number of important kernels, applicable to all categories [23]. And their activation should be relatively large. In our method,
we filtered out the unimportant neurons with low activation values. Then, calculated the distribution of the activated neurons for each class to evaluate the extraction of feature information by the convolutional layer.

We hypothesized that different channels activate different classes or feature patterns in specific layer during training procedure, with less overlapping of cross-class information expressed by different kernels. The purer the feature extracted, the more 'professional' the kernel is.

To verify the reliability of our principle, a group of experiments were carried out. The comparison of intra- and inter-model has already proved that there is a positive correlation between the purity and the performance of kernels, layers, or models. At the same time, the purity of the characteristic layer increases gradually with the increasing of training epochs, which provides another evidence to our hypothesis.

The theory of informatics can also provide the interpretation of purity from another aspect. As we described above, purity is defined by following the conception of information entropy in specific reception field in CNN. The low value of purity reflects an average and uniform activation from inputting information, which represents ambiguous, unclear, and even chaos in informatics. On the contrary, the high level of purity reflects specific and unique activation, which means significant and salient from inference. And the significant and salient feature extraction are the purpose we want to achieve in designing and training a CNN model. In traditional method, this goal of significant and salient feature extraction can only be evaluated by its final performance, such as classification accuracy. However, in our method, we have provided the purity as another viewpoint to measure the performance.

By comparing the value of purity with the performance of grad-CAM in the style of feature map, another phenomenon can be observed that: in most of our experimental cases, high purities reflected higher level of salient and significant regions in the feature map of grad-CAM, and more semantic information can be activated from specified class. For different models, the purity was positively correlated with the performance of the last feature map visualized by grad-CAM. At the same time, the purity of the characteristic layer and feature map increased gradually with the deepening of the training degree.

One of the deficiencies is that the proposed method cannot solve the problem of bias sampling of the dataset. The calculation of purity needed relatively large scale and category-balanced dataset as the input. If there was an inputting dataset with a lack of categories, the calculated purity may not fully reflect the performance of a well-trained model. It means that the purity was only responsible and meaningful to give inputting datasets/categories. Additionally, our algorithm workflow filtered out the weaker neurons and lost a small part of the information. However, from another point of view, this is helpful to solve the problem of filter-class entanglement[20]. In general, the characteristic measurement result is a more specific result of neuronal activation.

6. Conclusion
In this paper, we proposed an innovative explainable metric, namely feature purity, to interpret the performance of feature extraction in CNN. Feature purity can quantitatively measure the degree of feature extraction in CNN without the dependency of label or specific network structure; and has shown significant applicability and generality across models. Future work is devoted to considering retraining the poor purity feature layer to see if it can further improve the performance of the network. At the same time, for the model with sufficient performance, the model pruning of the kernel with poor activation is considered to further improve the performance of the model by exploring the characteristic layer with poor contribution to the model.

Acknowledgments
This work was in part supported by the Science and Technology Research Program of Chongqing Municipal Education Commission(Grant No.KJQN202001131)
References

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, pages 580 – 587, 2014.

[2] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, pages 3431–3440, 2015.

[3] Matthew D., and Rob Fergus. Visualizing and understanding convolutional networks. In ECCV, pages 818-833, Springer, 2014.

[4] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806, 2014.

[5] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In CVPR, pages 2921-2929, 2016.

[6] Ramrasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In ICCV, pages 618-626, 2017.

[7] Sturmfels, Pascal, Scott Lundberg, and Su-In Lee. "Visualizing the impact of feature attribution baselines." Distill e22, 2020.

[8] Haotian Ma, Yinqing Zhang, Fan Zhou, and Quanshi Zhang. Quantifying layerwise information discarding of neural networks. arXiv preprint arXiv:1906.04109, 2019.

[9] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?" Explaining the predictions of any classifier. In SIGKDD, Pages 1135-1144. ACM, 2016.

[10] Koh, Pang Wei, and Percy Liang. Understanding black-box predictions via influence functions. arXiv preprint arXiv:1703.04730, 2017.

[11] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P. Xing. High-frequency Component Helps Explain the Generalization of Convolutional Neural Networks. In CVPR, pages 8684-8694, 2020.

[12] Ruofan Liang, Tianlin Li, Longfei Li, Jing Wang, and Quanshi Zhang. Knowledge consistency between neural networks and beyond. In ICLR, 2019.

[13] Krizhevsky, Alex, and Geoffrey Hinton. Learning multiple layers of features from tiny images. In Technical report, 2009

[14] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and F.-F. Li. ImageNet large scale visual recognition challenge. International Journal of Computer Vision (IJCV), 115(3):211–252, 2015.

[15] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks. Communications of the ACM, 2017, 60(6): 84-90.

[16] Karen Simonyan, and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[17] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In CVPR, pages 4700-4708, 2017.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, pages 770-778, 2016.

[19] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In CVPR, pages 7132-7141, 2018.

[20] Haoyu Liang, Zihao Ouyang, Yuyuan Zeng, Hang Su, Zihao He, Shu-Tao Xia, Jun Zhu, and Bo Zhang. Training Interpretable Convolutional Neural Networks by Differentiating Class-specific Filters. arXiv preprint arXiv:2007.08194, 2020.

[21] Quan-shi Zhang, and Song-chun Zhu. Visual Interpretability for Deep Learning: a Survey. Frontiers of Information Technology & Electronic Engineering, 2018, 19(01):27-39.

[22] Zijie J. Wang, Robert Turko, Omar Shaikh, Haekyu Park, Nilakash Das, Fred Hohman, Minsuk Kahng, and Duen Horng Chau. CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization. In TVCG, 2020.

[23] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba.
Understanding the role of individual units in a deep neural network. Proceedings of the National Academy of Sciences, 2020.

[24] Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., & Sayres, R. (2018). Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). Proceedings of the 35th International Conference on Machine Learning, in PMLR 80:2668-2677.

[25] Li, Y., Lin, S., Zhang, B., Liu, J., Doermann, D., Wu, Y., ... & Ji, R. (2019). Exploiting kernel sparsity and entropy for interpretable CNN compression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 2800-2809).

[26] Li, Y., Wang, L., Peng, S., Kumar, A., & Yin, B. (2019, September). Using feature entropy to guide filter pruning for efficient convolutional networks. In International Conference on Artificial Neural Networks (pp. 263-274). Springer, Cham.