Evaluating CNN interpretability on sketch classification

Theodorus, Abraham, Nauta, Meike, Seifert, Christin

Abraham Theodorus, Meike Nauta, Christin Seifert, "Evaluating CNN interpretability on sketch classification," Proc. SPIE 11433, Twelfth International Conference on Machine Vision (ICMV 2019), 114331Q (31 January 2020); doi: 10.1117/12.2559536

Event: Twelfth International Conference on Machine Vision, 2019, Amsterdam, Netherlands

Evaluating CNN Interpretabilty on Sketch Classification

Abraham Theodorus, Meike Nauta, and Christin Seifert
University of Twente, Enschede, The Netherlands

ABSTRACT

While deep neural networks (DNNs) have been shown to outperform humans on many vision tasks, their intransparent decision making process inhibits wide-spread uptake, especially in high-risk scenarios. The BagNet architecture was designed to learn visual features that are easier to explain than the feature representation of other convolutional neural networks (CNNs). Previous experiments with BagNet were focused on natural images providing rich texture and color information. In this paper, we investigate the performance and interpretability of BagNet on a data set of human sketches, i.e., a data set with limited color and no texture information. We also introduce a heatmap interpretability score (HI score) to quantify model interpretability and present a user study to examine BagNet interpretability from user perspective. Our results show that BagNet is by far the most interpretable CNN architecture in our experiment setup based on the HI score.

Keywords: Interpretable CNN, sketch classification, explainable AI, quantifying model interpretabilty

1. INTRODUCTION

Recent developments in Deep Learning, especially Convolution Neural Networks (CNNs), have significantly improved performance in computer vision related tasks, e.g. object recognition and scene understanding [1]. However, superior performance comes with a trade-off. Interpreting deep CNNs is challenging [2] and the trained model is often used as a black-box system. Even though the model produces the correct output, sound reasoning needs to be present, to prevent misleading outcomes and distrust. Therefore, apart from performance, machine learning *explainability* is crucial, especially in high-risk scenarios.

There are several approaches in explaining deep CNNs to be more understandable for humans. First, the explanation can be a visualization of the network layers [3] which gives insights into what the model has learned. Explaining one specific decision is usually more favorable towards non-technical people. Lastly, one can embed interpretability directly in the model by constructing a specialized interpretable CNN architecture [3,4,5,6]. This is done by Brendel et al. [4] who tries to mimic the learning approach of humans. Humans have a natural capability to perceive how an object differs from others. Taking the example of distinguishing cat and dog images, humans consider certain object parts, such as whiskers and pointy ears, when deciding that the animal in a certain image is a cat. Other object parts, such as longer jaw and darker eyes, are considered when deciding that it is an image of a dog. Brendel et al.[4] incorporates this concept by using bag-of-local features as the features for the network, called BagNet. The features are comparable to small image patches and they could be learned through the network by applying a specific CNN design.

Contributions Existing methods for visualizing CNNs, including the BagNet approach, [4] are usually applied on natural image data. This means that the designed approach has been proven to be capable in generating interpretable models. However, this could be caused by the richer features provided by the dataset, i.e, colors and textures. In this paper, we apply BagNet to a dataset of sketches of simple concepts drawn by humans [7]. The goal is to identify whether the interpretable CNN architecture proposed by Brendel et al. [4] could perform well both in terms of accuracy and explainability on a less rich dataset, namely with limited color and no texture information. Furthermore, we also formulate a metric to quantify model interpretability based on generated class activation maps.

The human sketch dataset consists of 251 distinct classes. Because the data was crowd-sourced, the drawing varieties contained in one class is reasonably high. Moreover, the drawings only consist of grayscale line strokes and do not contain rich features, such as texture and shades. We compared the classification performance of the interpretable BagNet [4] with several pre-trained non-interpretable CNN models, i.e., VGG [8], ResNet [9], and DenseNet [10]. Besides evaluating their classification performance, we *explained* each model by extracting and comparing class activation maps from several test images. The class activation maps [11] are visualizations representing how the model sees parts of the objects before inferring prediction. Several metrics, i.e., heatmap intersection and model fidelity, are

Twelfth International Conference on Machine Vision (ICMV 2019), edited by Wolfgang Osten,
Dmitry Nikolaev, Jianhong Zhou, Proc. of SPIE Vol. 11433, 114331Q · © 2020 SPIE
CCC code: 0277-786X/20/$21 · doi: 10.1117/12.2559536

used to quantify each model's interpretability based on these class activation maps. Furthermore, in order to validate the meaningfulness of the explanation, humans were asked to label the highlighted object parts via a questionnaire. The intuition behind this is that the explanation is only interpretable when humans name the highlighted object parts. In conclusion, the main goal of this paper is to identify whether the BagNet architecture is well-performing and interpretable when trained on *less-rich-features* dataset, by evaluating the model performance and interpretability trade-off, as well as a user study on model interpretability.

In the remaining of this paper, related work is discussed in more detail in Section 2. The approach of this research is detailed in Section 3, where the used dataset, pre-processing, model training, and evaluation are examined further. The results of this research, and how these should be interpreted, are shown and interpreted in Section 4. This section will also contain respondents' opinion on the generated model explanation. In Section 5 the observed insights and challenges will be discussed. Finally, we state our conclusion in Section 6.

2. RELATED WORK

Sketch classification has been investigated in several research papers. The authors introducing the sketch data set, Eitz et al., produced a sketch classifier using a Support Vector Machine with 56% accuracy.[7] In their research, humans still outperformed this classifier with 73% accuracy. Yu et al. [12] used the same *TU-Berlin* sketch dataset and constructed a CNN architecture called "*Sketch-A-Net*" which could finally beat the human benchmark with 74.9% accuracy. Nevertheless, none of these networks are interpretable to humans.

In a visual interpretability survey [13] conducted by Zhang et al., applying interpretability in the network design has become one of the research directions in interpretable AI and this paper has a similar direction. There are existing interpretable CNN architectures. Zeiler et al. [3] visualizes the learned filters of a CNN to get better understanding on what the model sees by mapping the filters back to the input pixel space through a deconvolutional network. The produced visualization is helpful for understanding what filters are being learned by the model and it is also useful for debugging the model internals. However, the learned filters could only be visualized per activation unit and not per layer, thus the visualization is not intuitively grasped.

Brendel et al. [4] introduces a CNN architecture which is able to classify images based on bag-of-visual-features. This approach tries to reinforce image features representations to use parts of images as the features. The method is similar to feature extraction in text classification, namely bag-of-words, where distinct vocabularies in a corpus are counted to form a feature vector. In the case of images, the term *feature vector* is defined by the number of occurrences of each *visual word* in the vocabulary. The authors constructed a deep neural network architecture called BagNet which uses *bag-of-visual-features* retrieved from ResNet inferences and connected them to a linear classifier. The linearity of the model allows them to measure the visual features evidence of the predicted outcome. This approach uses a solid feature representation that is understandable by human. The architecture of BagNet is similar to ResNet with the only difference of 1x1 kernels that replace ResNet's 3x3 kernels.

BagNet is related to the approach of Chen et al. [5] which mimics the human reasoning process into the CNN architecture. This architecture is able to learn sets of prototypical image parts for each class. Even so, the produced representation is limited by a pre-allocated number of learned protoyipical features per class. Similarly, Zhang et al. [6] constructed an interpretable CNN architecture which could produce semantic representations of the object parts in an image. This approach is able to push the representations of each filter towards corresponding object parts, then the semantic representation can be used to reason why the particular images belong to certain classes. A more extensive approach was introduced by Konam et al. [14] in which both the most important neurons and patches of image can be identified by using two separately trained classifiers. This method looks promising, however training a secondary classifier to retrieve the representative patches of images add another layer of complexity.

3. EXPERIMENT

Dataset

The *TU-Berlin* sketch dataset [7] consists of 251 sketch categories with approximately 100 images per category/class. For our work, we manually selected a subset of 10 classes that we expect are very hard to classify. We chose similar looking classes, e.g. *revolver* along with *rifle* and *knife* along with *sword*.

BagNet requires a 224x224 input dimension. Therefore, we resized the original 1111x1111 images to 224x224, i.e., the dimension that BagNet expects. Furthermore, the RGB sketch images are converted into grayscale images. The dataset, which comprises of 1000 images from 10 classes, is small and we apply data augmentation to prevent model overfitting. In the dataset we observed various angles of how people drew the sketch images and therefore duplicated each image 11 times, then rotated by 11 different angles from 30 to 330 degrees. This results in a data set of approximately 10,000 images in total.

Pre-trained Model

Transfer learning is a popular technique to save training time by taking advantage of existing high performing models' learned weights. It involves freezing most of the layers and unfreezing several high-level layers, thus less parameters are required for training. The configurations mentioned in Table 1 are applied to the pre-trained models in order to train the sketch dataset.

Table 1. Pre-trained Model Settings

Model	Unfrozen Layers
VGG-16	last layer
ResNet-50	4th and last layer
DenseNet-169	denseblock4 and last layer
BagNet-33	4th and last layer

Brendel et al. [4] provided a pre-trained BagNet which was trained on ImageNet datasets. According to their experiments, BagNet-33 interpretability visualization is much clearer, thus we used a pre-trained BagNet-33 while unfreezing the 4th layer and replacing the last layer with linear layer. Unlike ResNet or VGG, the number suffix (33) in BagNet corresponds to image patches size used in the model. For comparison purposes, some noninterpretable pre-trained models are also used. Since each of them performs differently on the sketch dataset, we determined optimal parameters empirically.

Training and Evaluation

After performing data augmentation, each class comprises of 960 images. The dataset is randomly split into 90% training and 10% test data. The training data is then split further using 6-fold cross validation. The training uses early stopping with permission size of 5 epochs. The models generated by each fold are then evaluated on the test set and the resulting accuracy values are averaged to get the model's final accuracy. Then, the best folds of each pre-trained model are chosen.

The heatmaps extraction approach used in this paper follows the method [4] presented by Brendel et al. For each correctly predicted test image by each model, the patches of size 33x33 are extracted, then fed into all of the chosen models to obtain the class activation heatmaps. The images are padded (using 16 pixels zero padding surrounding the images) beforehand, so the resulting activation maps have identical size with the input image. BagNet-33 architecture learns 33x33 local image patches representation, hence this particular size is chosen.

The 10-chosen classes out of 251 classes are determined based on observed similarities: For instance, the *axe* class is similar to the *sword* class, and a *cup* class is similar to *teapot* and *wineglass*. Therefore, the generated heatmaps of these similar classes are inspected. Furthermore, heatmaps of misclassified images are investigated. Average accuracy on the test dataset is used as the evaluation metric for model performance.

By observing the generated heatmaps (see Fig. 2), we could grasp the impression which model can better highlight the object parts. In order to quantify that impression, we introduce an evaluation metric. The Heatmap Interpretability score (HI score) quantifies model interpretability based on its class activation heatmaps. The main idea of this metric is to reward meaningful image class activation heatmaps and penalize the heatmaps which are too off-the-marks. For example, if we look at the images in Fig. 2 the heatmaps of BagNet-33 look overall the most clustered among others, thus they will more likely get higher HI scores. Meanwhile, heatmaps of VGG-16 on the cup and the wine glass images are more clumped on the white pixels, thus their HI scores will get penalized.

$$HI = \frac{(TopAcc - ExcludedAcc) \times (TestAcc - ExcludedAcc)}{(1 + IntersectionRatio)} \tag{1}$$

The generated heatmaps are used to derive the required variables in this formula. First, we calculate the intersection of the generated heatmaps and the black pixels of the test images to produce *IntersectionRatio*. Then, in order to evaluate model fidelity, we generate two sets of images:

Set 1 - Filtered images with masked heatmaps. For each heatmap, we generate a mask by computing the top 20% highest heatmap intensity values then set them to 1 while setting the rest to 0. We then multiply the mask with each test image and setting the filtered pixel values to 1. This means that a resulting image only contains those pixels of the sketch that have the 20% highest heatmap intensity.

Set 2 - Unmeaningful images. We subtracted each test image with its corresponding mask, leaving a set of images which don't contain pixels of the top 20% highest heatmap intensity. This set of images should therefore possess 'unmeaningful' information.

The intuition is as follows: If the heatmaps are meaningful, a network would be able to classify a sketch where only the 20% most highlighted parts are shown (set 1). On the other hand, it should not be able anymore to classify an image where those highlighted parts are removed (set 2). We feed these two sets of images to each trained model in order to obtain new accuracy values, *TopAcc* and *ExcludedAcc* respectively. The difference between *TopAcc* and *ExcludedAcc* should produce either positive or negative values. The larger the difference, the higher the meaningfulness of the generated heatmaps. Meanwhile, the difference of *TestAcc* and *ExcludedAcc* explains the test accuracy drop after excluding the top 20% heatmap intensities portion. However, the HI score is penalized if *IntersectionRatio* is high, which in this case signifies inability of the heatmaps to spot meaningful object parts since then almost the whole sketch is highlighted.

Furthermore, we use a questionnaire to evaluate interpretability of BagNet-33. One image per class is randomly chosen from correctly predicted test images and included in the questionnaire. Respondents, mainly students (age 20-30), are asked to label as many object parts as possible highlighted by the given heatmaps. For example, three highlighted regions in the heatmap of a cup image may get manually labelled as 'cup handle', 'cup rim', and 'cup base'. We collected the object parts labels in order to measure which fraction of the highlighted regions could be identified by the respondents. The intuition is that highlighted regions that cannot be manually labeled or described are most likely not meaningful object parts. The labels are categorized manually into predetermined class' object parts, then each class' object parts recognizability, denoted by "% identified object parts", is calculated.

4. RESULTS

Performance

Table 2 provides an overview of the classification results. As can be seen, BagNet-33 surpasses VGG-16 model's average test accuracy but is outperformed by 3% by the other pre-trained models.

Table 2. Experiment results

Model	*TestAcc*	Heatmap intersection	*TopAcc*	*ExcludedAcc*	HI score
VGG-16	0.875	**0.902**	0.158	0.156	0.001
ResNet-50	0.976	0.747	0.660	0.780	-0.014
DenseNet-169	**0.979**	0.774	0.756	0.776	-0.002
BagNet-33	0.946	0.775	0.612	0.458	**0.043**

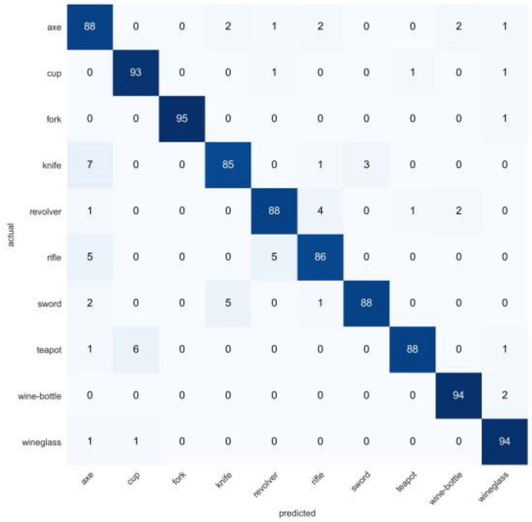


Figure 1. Confusion matrix of BagNet-33 predictions

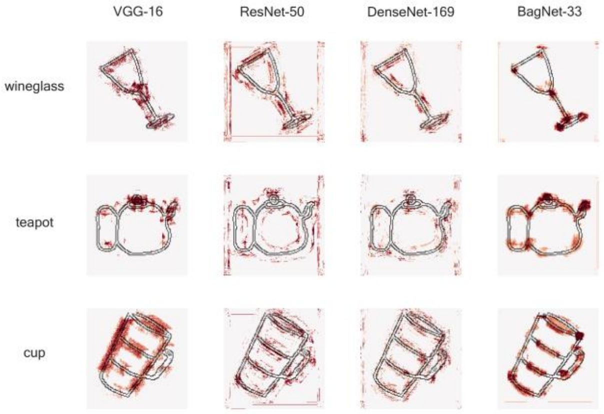


Figure 2. Class activation maps of different models

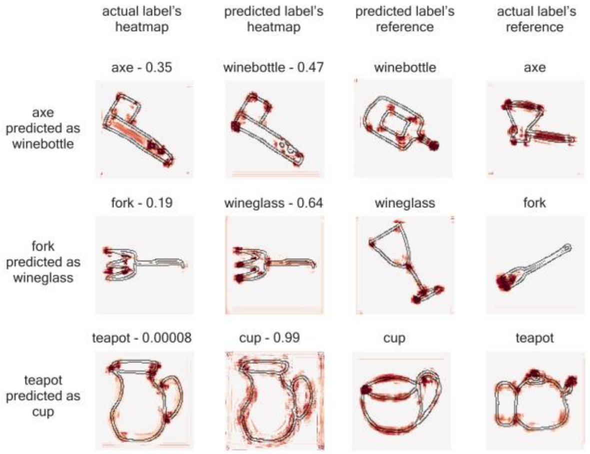
In the experiment design, some object categories with resembling shapes are intentionally hand-picked to investigate how the produced model generalizes on these look-alike categories. Although the overall model performance is good, some look-alike categories have misclassified outcomes as expected. As can be seen in the confusion matrix of BagNet-33 (cf. Figure 1), *rifle* is mostly misclassified as *revolver*, *sword* is mostly misclassified as *knife*, and *teapot* is mostly misclassified as *cup*. The misclassified results were mostly due to similar spatial characteristics, e.g., blade parts of swords and knives, similar object parts' presence in both the revolver and rifle class, as well as circular edges at the bottom of both cups and teapots. These spatial characteristics are explained through class activation maps further below and could be observed in Figure 3.

Learned representations

The class activation maps of each model are shown in Figure 2. The darker shaded area of an activation map indicates the more important patches which the model considers. Both ResNet-50 and DenseNet-169 have more scattered activation maps, but VGG-16 and BagNet-33 have successfully learned more localized activation maps. Overall, BagNet-33 model's class activation maps seem to be able to portray object parts' highlights in the most fine-grained manner. This result is similar to what Brendel et al. [4] have shown on the ImageNet dataset with BagNet-33. Nevertheless, even though the sketch dataset has less-rich features, BagNet-33 model is able to identify properties such as edges and shapes. These properties could be considered as the learned representation of the model and could be used as the explainable elements of the model as well.

Looking at misclassified class activation maps

Taking a step back to the misclassified classes, the class activation maps provide evidences on what classes the images should belong to. On the first row of Figure 3, an *axe* is predicted as a *wine-bottle*. If we observe the blade part of the *axe*, its shape resembles the brand label of the *wine-bottle*. Thus, it is likely that the model sees this part as the stronger evidence for the *wine-bottle* class. Next, on the second row a *fork* is misclassified as a *wineglass*. The edge part of a *fork* is usually smaller as depicted by the actual label's reference image, however, this particular fork has a wider edge. Once again, the model sees this property as a stronger evidence for the predicted class. Finally, the last example's classes belong to the classes which were intentionally picked due to look-alikeness. In a glance, the *teapot* image has characteristics of a cup, i.e. having a rim and doesn't have a spout. Furthermore, the predicted heatmaps are grouped around the edges of the teapot which are usually observed in heatmaps of a *cup*. From these examples, one could say that the observed spatial characteristics of misclassified images tend to be supported by the heatmaps of the predicted classes.



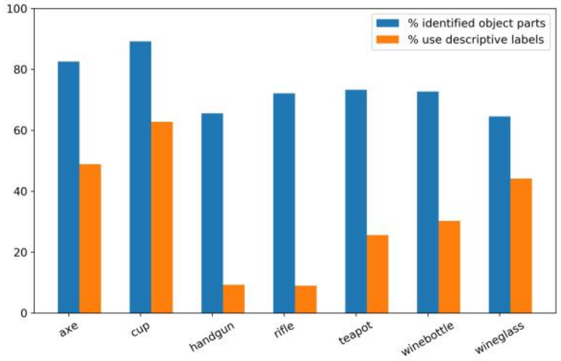


Figure 3. Misclassified labels' class activation maps. The mentioned decimal values are obtained from softmax outputs of the BagNet-33 model. The references show correctly predicted images of their corresponding classes

Figure 4. User evaluation results. On average, the respondents could identify 74.3% of object parts across all classes. Meanwhile, 32.8% of the labelling attempts include descriptive words, e.g. bottom part of the cup, or upper curvy side.

Quantifying model interpretability

From Table 2, heatmaps of VGG-16 seem to cover most of the line-strokes (90.2% heatmap intersection), meaning that VGG highlights almost the whole object instead of specific object parts. This corresponds to a high *IntersectionRatio* in Eq. 1. This is further confirmed by the extreme accuracy drop of VGG-16 when its produced top 20% heatmap intensities portion is excluded from the test images (*ExcludedAcc*). The generated heatmaps of ResNet and DenseNet could also be considered not meaningful since the accuracy of images *without* top 20% of heatmap intensities is higher than the accuracy of images that contain only top 20% heatmap intensities portion (*ExcludedAcc* vs *TopAcc*). The fact that both models' accuracies do not drop as significantly as BagNet and VGG-16 when the top 20% heatmap intensities portion is excluded supports the previous statement.

BagNet is the only architecture which has a significant positive margin between both top 20% heatmap intensities portion and the excluded one, with 15% accuracy margin respectively. This means the top 20% heatmap intensities portion of BagNet can represent the test image features better. Moreover, BagNet's produced heatmaps do not cover all line-strokes, leaving some object parts to be unimportant. These two factors signify BagNet heatmaps to be more interpretable in explaining object parts than the rest of the pre-trained models. All in all, these breakdowns are reflected well on the HI score of each model.

User evaluation of model interpretability

The questionnaire was filled by 45 participants which were slightly dominated by male (64.4%) and 26-30 age group (60%). In the end, 3 classes (*knife*, *sword*, and *fork*) were excluded from the questionnaire to avoid bias, because they only consist of one shaded object part. Two responses were also excluded due to irrelevant and random terms in their responses.

Respondents were asked to label the shaded object parts to measure object parts recognizability. This metric, which is defined by "% identified object parts" in Figure 4, shows that respondents could identify most of the highlighted parts with 74.3% average percentage value. The top classes of this metric are *cup*, *axe*, and *teapot*. Objects which are commonly found in daily basis or relatively easy to describe tend to have higher object parts' recognizability. Furthermore, respondents' justification on how unimportant the shaded object parts and also respondents' lack of specific vocabulary knowledge could explain why there were some unidentified object parts.

Related to lack of vocabulary knowledge, the respondents frequently also had to replace specific vocabularies with longer descriptive words, such as ”the bottom part of the cup” and ”the tip of the knife”. All attempts to describe the labels of a particular class are compiled into a new metric called "% use descriptive labels" as depicted in Figure 4. On average, 32.8% of the labels contain descriptive words which indicate respondents' difficulty in labelling some object parts.

There is a relationship between "% identified object parts" and "% use descriptive labels" metric. The respondents tend to use descriptive words in order to label classes which have high object parts recognizability, such as *cup* and *axe* class. Some of the descriptive words include "circumference of the cup", "place for mouth", and "top part of the axe". Most of the object parts which were labeled using descriptive words belong to the edges of the objects. On the other hand, the handgun class, which has emerged as one of the classes with the least recognizable object parts, has the lowest usage of descriptive words. Most of the respondents either perfectly labeled the parts of a handgun or incorrectly labeled some parts. These contrasting results show that human could still identify the object parts even though they might not have the knowledge on the corresponding vocabularies.

5. DISCUSSION

Our experiment shows that BagNet has a good performance on the sketch dataset with close to 95% average test accuracy but still fell behind ResNet and DenseNet by a 3% margin. We argue that this difference is acceptable considering the more localized class activation maps BagNet could offer. The sketch dataset was used to investigate BagNet's performance on less-rich-features dataset. Since the performance was proven to be good, BagNet could be applied to similar datasets too and is expected to give comparable performance.

Moving on to the class activation maps result, BagNet has the most localized class activation maps compared to other models.The activation maps between classes give an impression that the learned heatmaps are sufficient only for distinguishing the classes and not being able to learn the general representation of the object itself. For example, considering the heatmaps of class *sword* and *knife*, the heatmaps only focus on the tip of both the sword and knife. In this case, the model sees the tip part as the most distinguishing factor between the two. However, humans would also consider the handle shape of both classes to be important.

The introduced *HI score* is able to rank model interpretability based on its class activation heatmaps. We think this score would also be applicable for *rich-features* dataset. However, the *IntersectionRatio* needs to be adapted for more complex datasets that have non pixel-wise labels, e.g. bounding boxes. We also think a certain threshold value of *HI score* could be determined by investigating its values on more datasets.

Putting this factor aside, the learned heatmaps could in fact be used to explain counterfactual examples. Coming back to the *knife* and *sword* class example, based on the learned heatmap, one could say an image is classified as a sword because it has two curvy edges on the tip of the object or an image is not a knife because it doesn't have an intersection between a straight edge and a curvy edge on the tip of the object. Currently this explanation is produced manually, however, a neural network could be trained to perform this. In this case, the image patches which have the highest class activation value along with their object parts annotation could be used as features.

The conducted user evaluation was limited to 7 images, a larger scale user evaluation could however be performed to retrieve more labels for more images. Currently, several object parts were described by several terms on the questionnaire. For instance, a cup rim was described as circumference of the cup, sipping part, and cup mouth. Therefore, these collected user labels on each class object parts can be compiled into one dictionary of stemming vocabulary and then used as a stemmer for preprocessing the results of a larger scale user evaluation. Besides, they could also be used to produce textual explanations why certain objects are classified into some classes. The manual labels can also act as a counterfactual explanation why certain objects are not classified as other classes. Another variety for user evaluation is to let respondents guess the class label, given only small image patch which represents the highest class activation value. This way, a more direct object parts recognizability could be measured.

6. CONCLUSION

This paper presents an evaluation of an existing interpretable CNN algorithm, namely BagNet, which could learn bag of visual words as its feature representation. We applied BagNet to the sketch dataset to investigate the resulting model's interpretability on a *less-rich-features* dataset. Based on the resulting class activation maps, BagNet is able to localize object parts well represented by edges and also shapes. The conducted experiment compared BagNet performance with non-interpretable pre-trained models, i.e. ResNet, VGG, and DenseNet on the sketch dataset. According to the experiment results, BagNet performed adequately, surpassing VGG and only falling short behind ResNet and DenseNet by 3% margin. We also introduced HI score which quantifies CNN interpretability based on the generated heatmaps of the test images. Considering the interpretability of BagNet, which is quantified by our newly introduced HI score, its small test accuracy gap is acceptable.

Finally, the conducted user study produced two metric values to measure model interpretability from user perspective, i.e., object-parts recognizability and descriptive labels usage. On average, 74.3% of object parts were successfully identified by the respondents. Among all of the collected labels, 32.8% of the labelling attempts include descriptive words which indicate user difficulty in mentioning specific vocabulary to describe parts of the objects. All in all, most of the object parts could be identified well by the respondents, which implies a considerably good interpretability of BagNet, even though some respondents replaced specific vocabularies with descriptive words.

REFERENCES

[1] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L., "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision* **115**(3), 211–252 (2015).

[2] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R., "Intriguing properties of neural networks," *ICLR* (2014).

[3] Zeiler, M. D. and Fergus, R., "Visualizing and Understanding Convolutional Networks," *ECCV* (2014).

[4] Brendel, W. and Bethge, M., "Approximating CNNs with bag-of-local-features models works surprisingly well on imagenet," *ICLR* (2019).

[5] Chen, C., Li, O., Barnett, A., Su, J., and Rudin, C., "This looks like that: deep learning for interpretable image recognition," *CoRR* **abs/1806.10574** (2018).

[6] Zhang, Q., Wu, Y. N., and Zhu, S., "Interpretable Convolutional Neural Networks," *CVPR* (2018).

[7] Eitz, M., Hays, J., and Alexa, M., "How do humans sketch objects?," *ACM Trans. Graph.* **31**(4), 44:1–44:10 (2012).

[8] Simonyan, K. and Zisserman, A., "Very deep convolutional networks for large-scale image recognition," *ICLR* (2015).

[9] He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition," *CVPR* (2016).

[10] Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q., "Densely connected convolutional networks," *CVPR* (2017).

[11] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A., "Learning deep features for discriminative localization," *CVPR* (2016).

[12] Yu, Q., Yang, Y., Liu, F., Song, Y.-Z., Xiang, T., and Hospedales, T. M., "Sketch-a-Net: A Deep Neural Network that Beats Humans," *International Journal of Computer Vision* **122**(3), 411–425 (2017).

[13] Zhang, Q.-s. and Zhu, S.-c., "Visual interpretability for deep learning: a survey," *Frontiers of Information* Technology *& Electronic Engineering* **19**(1), 27–39 (2018).

[14] Konam, S., Quah, I., Rosenthal, S., and Veloso, M. M., "Understanding convolutional networks with APPLE : Automatic patch pattern labeling for explanation," *CoRR* **abs/1802.03675** (2018).