DOI: 10.15514/ISPRAS-2023-35(2)-1

Real Application of CNN Interpretation Methods: Document Image Classification Model Errors' Detection and Validation

A.O. Golodkov, ORCID: 0000-0002-0417-2622 <golodkov@ispras.ru>
O.V. Belyaeva, ORCID: 0000-0002-6008-9671 <belyaeva @ispras.ru>
A.I. Perminov, ORCID: 0000-0001-8047-0114 <perminov@ispras.ru>

Ivannikov Institute for System Programming of the Russian Academy of Sciences,
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia

Abstract. In this paper, we consider the case of applying convolutional neural networks interpretation methods to ResNet 18 model in order to identify and justify model errors. The model is used in the problem of classifying the orientation of text documents images. First, using interpretation methods, an assumption was made as to why the neural network shows low metrics on data that differs from training images. The alleged reason was the presence of artifacts on the generated training images, caused by the use of an image rotation function. Further, using the Vanilla Gradient, Guided Backpropagation, Integrated Gradients, GradCAM methods and the invented metric, we managed to accurately confirm the hypothesis put forward. The obtained results helped to significantly improve the accuracy of the model.

Keywords: CNN Interpretation; Document Image Classification; Document Orientation Detection.

For citation: Golodkov A.O., Belyaeva O.V., Perminov A.I. Real Application of CNN Interpretation Methods: Document Image Classification Model Errors' Idetection and Validation. Trudy ISP RAN/Proc. ISP RAS, vol. 35, issue 2, 2023. pp. 7-18. DOI: 10.15514/ISPRAS-2023-35(2)-1

Реальное применение методов интерпретации свёрточных нейронных сетей: обнаружение и объяснение ошибок классификатора изображений документов

А.О. Голодков, ORCID: 0000-0002-0417-2622 <golodkov@ispras.ru>
О.В. Беляева, ORCID: 0000-0002-6008-9671 <belyaeva @ispras.ru>
А.И. Перминов, ORCID: 0000-0001-8047-0114 <perminov@ispras.ru>

Институт системного программирования им. В.П. Иванникова РАН,
109004, Россия, г. Москва, ул. А. Солженицына, д. 25

Аннотация. В данной статье рассматривается случай применения методов интерпретации свёрточных нейронных сетей к модели ResNet 18 с целью обнаружения и объяснения её ошибок. Сама модель используется для решения задачи определения ориентации изображений текстовых документов. Изначально с помощью методов интерпретации было выдвинуто предположение о причине низкого качества предсказаний модели на данных, отличных от примеров из обучающего набора. Предполагаемой причиной оказалось наличие артефактов на тренировочных данных, которые были сгенерированы с использованием функции поворота изображений. Далее, с помощью методов Vanilla Gradient, Guided Backpropagation, Integrated Gradients, GradCAM и предложенной метрики удалось точно обосновать выдвинутое предположение. Полученные результаты помогли значительно улучшить точность модели.

Ключевые слова: интерпретация свёрточных нейронных сетей; классификация изображений документов; определение ориентации документа

Для цитирования: Голодков, А.О., Беляева О.В., Перминов А.И. Реальное применение методов интерпретации свёрточных нейронных сетей: обнаружение и объяснение ошибок классификатора изображений документов. Труды ИСП РАН, том 35, вып. 2, 2023 г., стр. 7-18. DOI: 10.15514/ISPRAS–2023–35(2)–1

1. Introduction

Convolutional neural networks are known for high metrics demonstrated in classification [1], segmentation [2], and object detection [3] tasks. This is due to the large number of improvements proposed to the original architecture [4] and the growing computing power. At the same time, the speed of neural networks due to the use of GPU also makes them an excellent replacement for the classical methods. ResNet 18, as an example of a convolutional neural network, shows high metrics in the classification task, while having a small number of parameters compared to other models, which causes its high speed. Also, this model has a simple architecture compared to later models, so its interpretation is quite simple. For these reasons, ResNet18 was chosen as the base solution.

Despite the advantages of convolutional neural networks, they still have one significant drawback - the difficulty of explaining model predictions. In other words, often neural networks are considered as a black box [5]. It can be critical when debugging a model.

At the moment, there are a large number of methods for interpreting neural networks, almost all of them are aimed at connecting parts of the input data with the prediction of the model [6]. For example, in image processing tasks, these methods help to highlight areas of the input image that led to a particular prediction.

In this paper, it is proposed to consider an example of applying existing methods for interpreting neural networks to identify the causes of systematic errors of a neural network in the problem of classifying the orientation of images of text documents. The very definition of the orientation of a text document is necessary for the correct recognition of text in the image. And often it is the image orientation correction module that is one of the first in the pipeline for text recognition. That is, the quality of text selection strongly depends on the work of this module. Therefore, fixing the errors of this module is necessary, but it may not be a trivial task at all.

In general, this paper considers several interpretation methods that allowed us to identify artifacts in the original training data. Also, the paper compares interpretation methods using several quality assessment metrics, one of which is proposed in this paper.

2. Related work

Consider interpretation methods for neural networks designed for image processing. According to work [7], existing methods for convolutional neural networks interpretation can be divided into two groups:

- Techniques to understand the decision-making process of a neural network by correlating the output of a neural network with areas of input data to see which parts of the input data most determine the output. Such methods can also be called local;
- Methods that allow us to consider in more detail the contents of the neural network and interpret how the internal layers' process data (not necessarily data in our subject area, but in the general case). Such methods could be called global.

The first group includes methods based on gradients as well as methods that allow you to build areas importance map of the input image. The main representatives of gradient methods are Vanilla Gradient [8], Guided Backpropagation [9], Integrated Gradients [10], GradCAM [11] and XRAI [12].

The second group includes methods for visualizing extracted features, the main of which is the Convolutional Features Visualization [13] method. In next sections global and local methods are described in more detail.

2.1 Vanilla Gradient

One of the first methods to explain neural networks is Vanilla Gradient [8]. For explaining the result of the classifier's work on the image, the method visualizes a matrix, which represents partial derivatives of the neural network output with respect to the input image. In the resulting matrix, the intensity of each pixel reflects the derivative of the output with respect to the corresponding image pixel. It is assumed that the greater the intensity of a pixel on a heat map, the more it affects the prediction of the classifier, i.e. the more this pixel is responsible for assigning it to a particular class.

2.2 Guided Backpropagation

A modification of the Vanilla Gradient method is the Guided Backpropagation method [9]. It differs from Vanilla Gradient in that all negative gradients are set to zero to build a feature map. It is assumed that positive gradients are positively correlated with the prediction of a particular class. In practice this makes the selected feature map less noisy. This method, like Vanilla Gradient, allows you to see which areas of the input image are most responsible for a particular neural network prediction.

2.3 Integrated Gradients

Another modification of the Vanilla Gradients method is the Integrated Gradients method [10]. This method differs from Vanilla Gradients and Guided Backpropagation in that it allows you to explore the neural network, as a function, not only at the point corresponding to the input image, but at some interval. It is assumed that the neural network, as some complex mathematical function, can fall into a local extremum when a certain input image is supplied. In this case, when calculating partial derivatives with respect to the input image, zero values can be obtained, which will not carry information about the importance of image areas for predicting a certain class. To consider a neural network on a certain interval, it is proposed to use the Integrated Gradients method on different combinations of inputs $x' + \alpha(x - x')$, where x is the original input image, x' is an auxiliary image that does not contain any information (e.g. white noise or a completely black image), and α is a coefficient from 0 to 1, indicating in what proportion the auxiliary image is added. That is, the neural network function is considered on the interval $[x',\ x]$ and integrated over α. The final formula for the value map looks like this:

$$IntegratedGradinets_i(x) = (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x_i' + \alpha \times (x_i - x_i'))}{\partial x_i} d\alpha$$

Here $F(x)$ is a mathematical function representing a neural network.

This method allows to accumulate gradient values for different inputs, which gives a more stable map of values and allows you to more accurately see the most important areas of the image for prediction.

2.4 GradCAM

Method of interpreting neural networks most frequently encountered in modern literature is the GradCAM [11]. It also produces a heat map showing which parts of an image contribute to specific class prediction the most. The principle of operation of this method can be schematically represented in the form of the following list:

- A convolutional layer is selected. Usually this is the last convolutional layer of the neural network, since it highlights the most important information;
- Matrix of the selected layer activation is saved during forward propagation;

- Partial derivatives are taken from the resulting array relative to the input image and the matrix of derivatives is stored on the selected layer;
- Two saved matrices are multiplied to form a heat map.

3. Interpretation methods justification

Four methods were chosen for the experiments, namely Vanilla Gradient, Guided Backpropagation, Integrated Gradients and GradCAM. These methods are suitable for the specifics of the task and are suitable for working with images of text documents.

XRAI is not suitable for the specifics of the task, because it uses an image segmentation algorithm that separates objects by color and shape. Since the images we work with consist mostly of white backgrounds and black text, meaningful segmentation is not possible: there are not enough different colors and different shaped areas in the image. This method is more suitable, for example, for images of the ImageNet type, since there are objects of different shapes and colors on the images and the background is not monotonous.

Convolutional Features Visualization is also not suitable for interpretation in our case, because the feature images that this method renders themselves require interpretation and do not reveal any human-understandable features in the input images. Examples are shown in the Fig. 1.

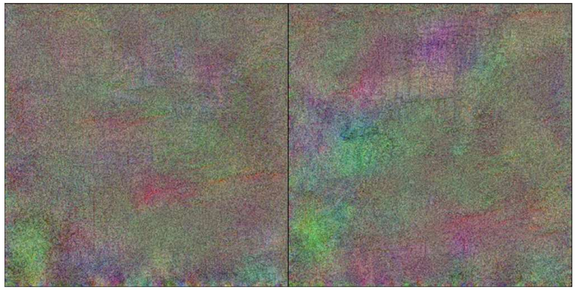
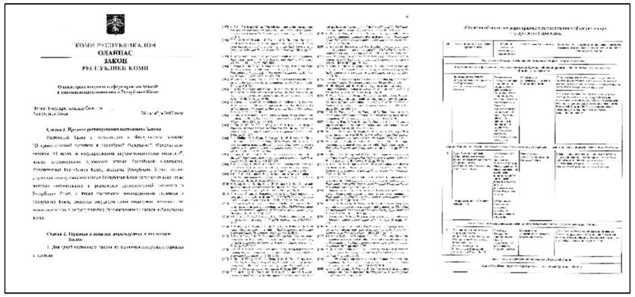


Fig. 1. Examples of visualized features for two convolutional filters of ResNet 18

4. Data set

To solve the problem of determining the document image orientation a data set was generated in order to train and test the selected model. The following is a detailed description of the data set.

2370 images of documents were taken as initial data, initially having a strictly vertical orientation (i.e. 0°). Among the documents have included in the data set, the following types can be distinguished: articles, terms of reference, statements, regulations, laws.



КОМИ РЕСПУБЛИКАЛОН
ОЛАНПАС
ЗАКОН
РЕСПУБЛИКИ КОМИ

Fig. 2. Examples of documents used in the data set. Left – one column text and a small image. Middle – two column document. Right –table document

Document are characterized by black text on a white background, Manhattan document page layout, and can also contain tables and pictures. Text on different types of documents can consist of one of two columns. Examples are shown in the Fig. 2.

From the images described above, a data set was further compiled for training the classifier using the following augmentation method: each image was rotated by angles that are multiples of 90° and saved as a separate copy. That is, a document that initially had a vertical orientation (i.e. 0°) was presented in four possible orientations in the data set: 0°, 90°, 180° and 270°.

This image rotation was applied so that all document orientation directions occur the same number of times in the data set. Thus, out of 2370 images, a set containing 9480 images was obtained. The training set consisted of 7580 images, while the test set consisted of 1900. It is worth noting that the data set was partitioned by source images, meaning rotated versions of the same image only occur in one part of the data set.

In the process of forming the training data set, a rotation function based on affine transformations was initially used. That is, for the corresponding rotation angle, a transformation matrix was formed, which was multiplied by the image matrix.

5. Interpretation quality estimation

The main way to estimate the quality of interpretation heat maps was proposed in paper [14]. The main idea of this method is to evaluate how accurately heat maps reflect the most important areas of the image for prediction. Thus, leaving the 20% of the most intense pixels in the image, according to the heat map, it is assumed that the class prediction score will change slightly if the heat map well indicated the most important areas. If the class prediction score changes significantly, it can be concluded that the heat map did not accurately reflect the most important areas. The exact formula is given below.

$$ADS\ (\%) = \frac{1}{N}\sum_{i=1}^{N}\frac{\max(0, Y_i^c - O_i^c)}{Y_i^c} * 100.$$

Here i is the index of picture in data set, c is the index of class in the data set, Y_i^c is the class prediction score on the original image, O_i^c is the class prediction score on the image, where, according to the heat map, 20% of the most intense pixels are left, the remaining pixels are painted white, N is the number of images in the data set. The example of heat map overlay is shown on Fig. 3.

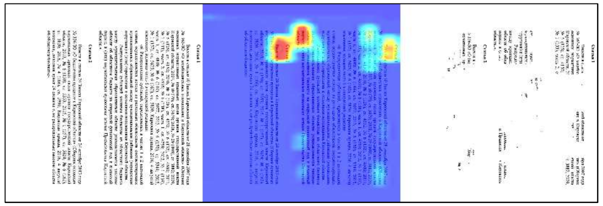


Fig. 3. Left - original input image. Middle - heat map obtained with interpretation method overlaid on image. Right - 20% most intense pixels left on original image according to heat map

6. Heat map result analysis

The first ResNet 18 interpretation experiment was aimed at identifying possible areas of images that lead to mispredictions. GradCAM was chosen as the interpretation method for the first experiment, and the last convolutional layer was chosen as the studied layer of the neural network, since it contains information about the most accurate features of the image. As the images on which the interpretation was carried out, images from the test part and images outside the data set were selected. In the resulting heat maps, red indicates the areas "more important" for prediction, and purple — "less important".

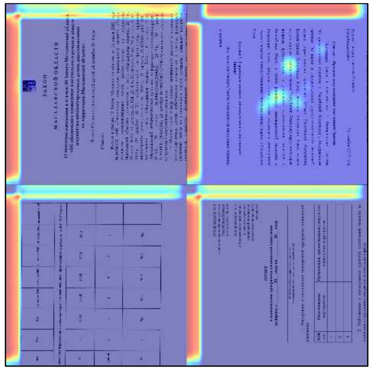


Fig. 4. Several examples of GradCAM heat maps obtained in the first interpretation experiment

Most of the received heat maps contained the same pattern – light stripes along the edges of the images. This means that the neural network "pays attention" to these corner areas. This result does not seem predictable, since, in theory, the neural network should highlight features from the entire image. In the following sections, this effect is discussed in more detail.

6.1 Alleged problem

In connection with the obtained heat maps, which indicated stripes along the edges as important for predicting image areas, an assumption arose about the presence of such artifacts in the images. At the same time, these artifacts, according to the assumption, should be contained either only in the training data set, or only outside it, since the pictures inside these groups are processed by the network in a similar way.

Using a difference visualization program, it was found that the same image, taken from the data set, and the same one rotated in a graphics editor, differ from each other by a shift of one pixel either vertically or horizontally, forming a black stripe at the edge of the image. In this case, this stripe is contained in the image from the data set.

By assumption, it was this artifact in the data set that influenced the fact that the neural network "pays attention" to the areas where these stripes are contained. The very reason for the occurrence of the artifact is discussed in more detail in the next part. Also, further experiments follow, allowing to check the put forward assumption.

6.2 Image rotation function

The *WarpAffine* rotation function of the OpenCV library, originally used in data set generation, is based on affine transformations. A rotation matrix is built from the rotation angle and applied to the matrix of the original image. For each pixel in the original image, the coordinates in the rotated image are calculated. But due to the rounding of floating-point numbers to integers, there is a shift by one pixel, as a result of which an artifact appears in the image in the form of a black stripe at the edge of the image. The location of the line also depends on the angle of rotation. Schematically, the location of the stripes is shown in the Fig. 5.

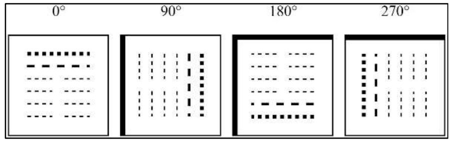


Fig. 5. Visualization of artifacts that appear when using the WarpAffine function

The rotation function, based on the transposition and reflection of the image matrix, does not create artifacts in the form of stripes, since in fact it simply rearranges the elements of the matrix.

6.3 Interpretation metrics introduction

The following metric was proposed for assessing the quality of interpretation of the neural network model to test the hypothesis put forward: the ratio of the average pixel intensity in the corner area (indicated in red) and the average pixel intensity in the central area (indicated in blue). The dimensions that define these regions vary in experiments. For *A*, these are 50 and 110 pixels; for *B*, these are 1150 and 1090 pixels, respectively. Let's call this metric corner to main ratio (CMR).

Recall that it is in the corner area that black stripes are contained that appear when using the rotation based on affine transformations. Thus, the metric clearly shows how much the pixels in the corner area are “more important” for the classifier than in the central area, that is, how much these bands are “more important” for the classifier than the rest of the image on a prediction step.

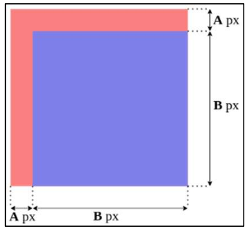


Fig. 6. Splitting the image into corner (marked in red) and central (marked in blue) areas

6.4 Fixed data set

After assuming the influence of artifacts of the rotation function on the work of the classifier, the data set was obtained from the original images in the same way, but using a different rotation function. Now, a combination of transpose and reflection of the image matrix was used as the rotation function. These images do not contain artifacts, since this rotation function uniquely rearranges the elements of the image matrix. The neural network was trained on a new data set for further experiments. In next experiments we will refer to these neural network weights as «*fixed model*» since this model does not have such unpredictable behavior. And we will refer to these pictures as “*Trans+Ref*” since such rotation function was used to generate them. We will also call the weights on which unexpected behavior of the neural network was detected an «*error model*», and pictures that were used to train it “*Affine*” since this rotation function was used.

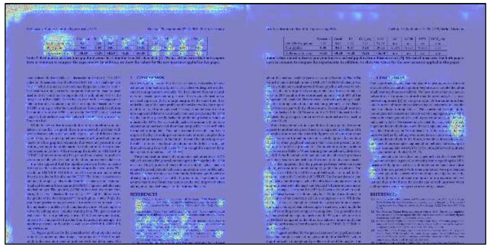


Fig. 7. The heat map obtained with Vanilla Gradients for Trans+Ref image. Left is a heat map for error model, right is a heat map for fixed model

7. Results of interpretation experiments

For experiments, 1000 images were selected from the data set, of which 500 were the results of the *WarpAffine* rotation function, and another 500 were the results of transposition and reflection. As a result, the corner to main ratio metric was averaged over the entire data set. The calculation of this metric was carried out for two different values of A from Fig. 6: for 110 and 50 pixels. The higher the corner to main ratio, the more important the corner area is compared to the main part of the image.

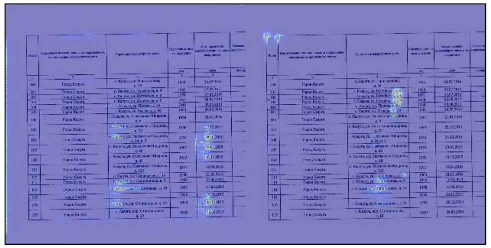


Fig. 8. The heat map obtained with Guided Backpropagation for Trans+Ref image. Left is a heat map for error model, right is a heat map for fixed model

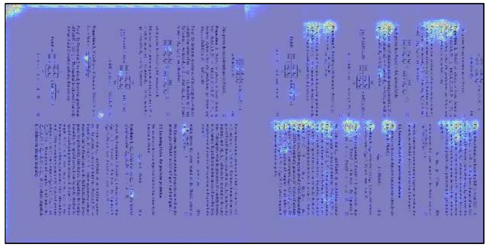


Fig. 9. The heat map obtained with Integrated Gradients for Trans+Ref image. Left is a heat map for error model, right is a heat map for fixed model

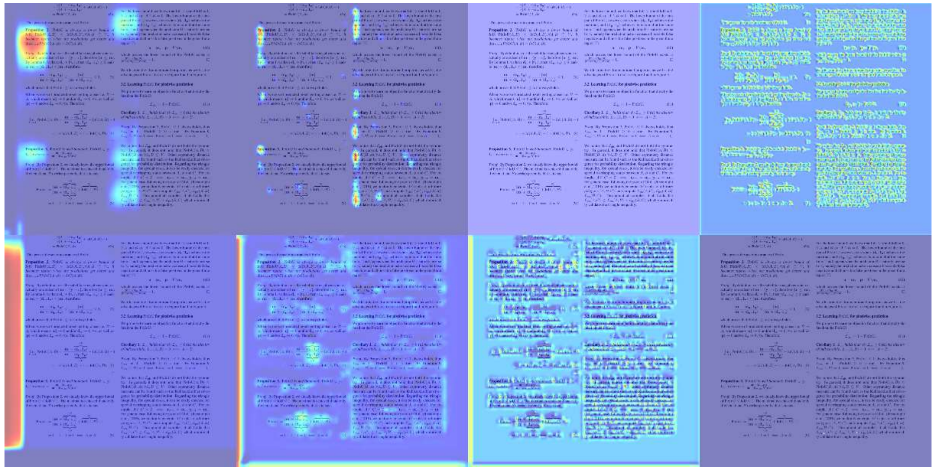


Fig. 10. The heat map obtained with GradCAM for the output layer of each of the four ResNet 18 blocks. Left is the conv_4, right is the conv_1. Bottom row contains heat maps for error model, Top row contains heat maps for fixed model. Trans+Ref image was used.

It is visible on Fig. 7, 8, 9, and 10 that the *error model* definitely "pays more attention" to the corner region of an input image however it does not contain artifacts since it is *Trans+Ref* image.

For the GradCAM method, CMR metric is calculated using heat maps obtained on the last convolutional layers of each of the four main blocks. In the Fig. 11, these blocks are named *conv_4*, *conv_3*, *conv_2* and *conv_1*, respectively.

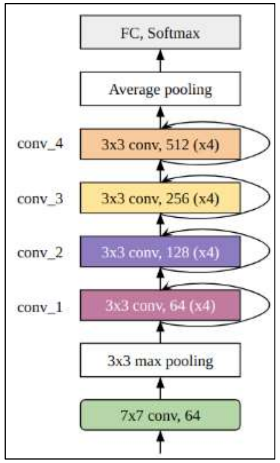


Fig. 11. Schematic representation of the ResNet 18 architecture

7.1 Artifact influence revealed by experiments

As can be seen from the Tables 1, 2, 3, and 4, CMR values are always grater for the *error model* regardless of rotation function used to generate an input image. The results of four chosen methods reveal the same trends, which strengthens the evidence.

Table 1. Corner to main ratio values for Vanilla Gradients method

Corner region width	*Affine* image		*Trans+Ref* image	
	error model	*fixed model*	*error model*	*fixed model*
110 px	**3,3268**	1,2885	**2,9800**	1,2986
50 px	**6,0972**	1,4316	**5,5656**	1,4549

Table 2. Corner to main ratio values for Guided Backpropagation method

Corner region width	*Affine* image		*Trans+Ref* image	
	error model	*fixed model*	*error model*	*fixed model*
110 px	**1,1163**	0,8657	**1,2541**	0,8860
50 px	**1,8716**	0,8876	**2,1454**	0,9268

Table 3. Corner to main ratio values for Integrated Gradients method

Corner region width	*Affine* image		*Trans+Ref* image	
	error model	*fixed model*	*error model*	*fixed model*
110 px	**3,7876**	1,2900	**5,1387**	1,3758
50 px	**6,9799**	1,4006	**10,0370**	1,5589

These results confirm our assumption about the effect of artifacts in the data set on the behavior of the classifier. It turns out that it were these artifacts that became the most important feature for classification, and when this feature did not appear on the image, this led to errors. The proof of this is the accuracy of the *error model* when processing *Affine* images. For example, this is reflected in Tables 5, 6 in the columns "Acc before heat map overlay". Namely, when processing *Trans+Ref* images, the accuracy is 0.5, while when processing *Affine* images, the accuracy is 1.0. At the same time, for the *fixed model*, when processing *Trans+Ref* images, the accuracy is 0.998, and when processing *Affine* images, the accuracy is 1.0.

Table 4. Corner to main ratio values for GradCAM method

Corner region width		*Affine* images		*Trans+Ref* images	
		error model	*fixed model*	*error model*	*fixed model*
110 px	*conv_4*	**2,3177**	0,8230	**2,3445**	0,8263
	conv_3	**1,5510**	1,0686	**1,7669**	1,0744
	conv_2	**1,0826**	0,8793	**1,0429**	0,8916
	conv_1	**0,9085**	0,7238	**0,9157**	0,7645
50 px	*conv_4*	**2,1987**	0,6608	**2,2196**	0,6625
	conv_3	**2,2600**	1,1567	**2,4967**	1,1735
	conv_2	**1,0512**	0,8998	**1,0259**	0,9017
	conv_1	**0,9354**	0,7511	**0,9516**	0,7855

Table 5. Results of the experiment for interpretation quality estimation for the methods Vanilla Gradient, Guided Backpropagation, Integrated Gradients. Bold indicates ADS less than 10 percent

	***Trans+Ref* image**				***Affine* image**			
	Acc before heat map overlay	**Acc after heat map overlay**	**Acc diff**	**ADS, %**	**Acc before heat map overlay**	**Acc after heat map overlay**	**Acc diff**	**ADS, %**
	Vanilla Gradient							
error model	0,500	0,5020	0,002	19,32	1,000	0,9980	0,002	**0,58**
fixed model	0,998	0,8880	0,110	13,24	1,000	0,8920	0,108	12,56
	Guided Backpropagation							
error model	0,500	0,5100	0,010	**9,79**	1,000	0,8240	0,176	19,88
fixed model	0,998	0,9960	0,002	**0,34**	1,000	1,0000	0,000	**0,15**
	Integrated Gradients							
error model	0,500	0,4700	0,030	34,28	1,000	0,8760	0,124	14,69
fixed model	0,998	0,7500	0,248	27,91	1,000	0,7580	0,242	26,97

Table 6. Results of the experiment for interpretation quality estimation of GradCAM heat map for each of four ResNet 18 blocks. Bold indicates ADS less than 10 percent

		***Trans+Ref* image**				***Affine* image**			
		Acc before heat map overlay	**Acc after heat map overlay**	**Acc diff**	**ADS, %**	**Acc before heat map overlay**	**Acc after heat map overlay**	**Acc diff**	**ADS, %**
error model	*conv_1*	0,500	0,4580	0,042	24,46	1,000	0,6520	0,348	36,39
	conv_2	0,500	0,3940	0,106	34,71	1,000	0,5820	0,418	44,66
	conv_3	0,500	0,3800	0,120	39,18	1,000	0,7740	0,226	24,12
	conv_4	0,500	0,4220	0,078	34,91	1,000	0,7680	0,232	25,82
fixed model	*conv_1*	0,998	0,7800	0,218	26,65	1,000	0,8100	0,190	23,36
	conv_2	0,998	0,8440	0,154	19,77	1,000	0,8200	0,180	20,97
	conv_3	0,998	0,7960	0,202	23,13	1,000	0,8180	0,182	21,66
	conv_4	0,998	0,8840	0,114	16,21	1,000	0,8740	0,126	16,65

7.2 Quality estimation

In order to assess how accurately the interpretation heat maps were obtained, it is proposed to evaluate the ADS metric described in section 5. This metric will also allow us to understand which of the interpretation methods used turned out to be more accurate.

From the results of the experiment in Tables 5 and 6 we can conclude that the most accurate heat maps were obtained by the Guided Backpropagation method, since The ADS of this method turned out to be the smallest. Moreover, difference in accuracy after heat map overlay is also smaller for Guided Backpropagation. This means that the heat maps of this method most accurately highlight important areas in the image, and therefore masking the image while leaving these most important areas does not lead to a noticeable decrease in the quality of the model.

8. Conclusion

In this work, we have considered how interpretation methods can be applied to a neural network in case of unexpected behavior. We have shown that interpretation can help identify the causes of erroneous behavior and further help improve model accuracy. At the beginning, thanks to the interpretation methods, an assumption was made about the influence of artifacts in the training data set on the behavior of the model. It was found that these artifacts in the form of black stripes with a width of one pixel occur at the edges of images due to the use of a rotation function based on affine transformations. A metric was introduced to test the proposed assumption. With the help of four interpretation methods and this metric, it was possible to accurately prove the hypothesis. All methods showed similar results. Using the average drop in score metric, we managed to choose the best of the four methods.

References

[1] Wang J., Yang Y. et al. CNN-RNN: A Unified Framework for Multi-label Image Classification. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2285-2294.

[2] Milletari F., Navab N., Ahmadi S.A. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In Proc. of the Fourth International Conference on 3D Vision (3DV), 2016, pp. 565-571.

[3] Xie X., Cheng G. et al. Oriented R-CNN for Object Detection. In Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 3500-3509.

[4] He F., Liu T., Tao D. Why ResNet Works? Residuals Generalize. IEEE Transactions on Neural Networks and Learning Systems, vol. 31, issue 12, 2020, pp. 5349-5362.

[5] Buhrmester V., Münch D., Arens M. Analysis of Explainers of Black Box Deep Neural Networks for Computer Vision: A Survey. Machine Learning and Knowledge Extraction, vol. 3, issue 4, 2021, pp. 966-989.

[6] Li G., Yu Y. Visual Saliency Detection Based on Multiscale Deep CNN Features. IEEE Transactions on Image Processing, vol. 25, issue 11, 2016, pp. 5012-5024.

[7] Barredo-Arrieta A., Díaz-Rodríguez N. et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information Fusion, vol. 58, 2020, pp. 82-115.

[8] Simonyan K., Vedaldi A., Zisserman A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv preprint arXiv:1312.6034, 2013, 8 p.

[9] Springenberg J.T., Dosovitskiy A. et al. Striving for Simplicity: The All Convolutional Net. arXiv preprint arXiv:1412.6806, 2014, 14 p.

[10] Sundararajan M., Taly A., Yan Q. Axiomatic Attribution for Deep Networks. In Proceedings of the 34th International Conference on Machine Learning, 2017, pp. 3319-3328.

[11] Selvaraju R.R., Cogswell M. et al. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In Proc. of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618-626.

[12] Kapishnikov A., Bolukbasi T. et al. XRAI: Better Attributions Through Regions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 4948-4957.

[13] Olah C., Mordvintsev A., Schubert L. Feature Visualization, 2017. Available at: https://distill.pub/2017/feature-visualization/?ref=hackernoon.com, accessed May 18, 2023.

[14] Desai S., Ramaswamy H.G. Ablation-CAM: Visual Explanations for Deep Convolutional Network via Gradient-free Localization. In Proc. of the IEEE Winter Conference on Applications of Computer Vision (WACV), 2020, pp. 972-980.

Information about authors / Информация об авторах

Alexander Olegovich GOLODKOV – Graduate of the Moscow Institute of Physics and Technology, senior laboratory assistant. Research interests: optical character recognition, computer vision.

Александр Олегович ГОЛОДКОВ – выпускник Московского физико-технического института, старший лаборант. Сфера научных интересов: оптическое распознавание символов, компьютерное зрение.

Oksana Vladimirovna BELYAEVA is a PhD student, researcher. Research interests: document layout analysis, document structure analysis, digital image processing, neural network data processing, image pattern recognition, face recognition.

Оксана Владимировна БЕЛЯЕВА является аспирантом, стажером-исследователем. Научные интересы: анализ шаблонов документов, анализ структуры документов, цифровая обработка изображений, нейросетевая обработка данных, распознавание образов компьютерного зрения, распознавание лиц.

Andrey Igorevich PERMINOV is a PhD student, researcher. His research interests include digital signal processing, neural network data processing, generation of artificial data.

Андрей Игоревич ПЕРМИНОВ является аспирантом, стажером-исследователем. Его научные интересы включают цифровую обработку сигналов, нейросетевую обработку данных, создание искусственных данных.