UDC 004.93
IRSTI 28.23.37

IMAGE PROCESSING TECHNOLOGY BASED ON CONVOLUTIONAL NEURAL NETWORK

Sh. Bakytnur, P.N. Abilova, N. Tasbolatuly
Astana International University, Nur-sultan, Kazakhstan
shuakbakytnur@gmail.com
ORCID ID:https://orcid.org/0000-0002-0511-7000

Abstract. A convolutional neural network is the structure of a neural network with multiple hidden layers that combines artificial neural network and deep learning. At present, the convolutional neural network, as a deep learning method widely used in various computer fields and effective in processing image information, can learn the features in the image after training, and complete the extraction and classification of image features. Therefore, it is widely used in computer vision research such as image processing and pattern recognition. Although the general structure of the convolutional neural network is very similar to the ordinary artificial neural network, it has a perfect effect on large-scale image datasets. This paper aims to sort out the convolutional neural network and its structural characteristics and advantages, introduce and analyze the basic working principle of the convolutional neural network, and realize the application example of CNN model training in image feature extraction and recognition.

Keywords: Deep learning; Convolutional neural network; image processing; image recognition

1. Introduction

With the rapid development of Internet information technology, processing more and more image data has become one of the popular research directions in the field of computer vision. As one of the ways to process image data, deep learning is also developing rapidly. It has been widely used in speech recognition, natural language processing, computer vision, and other fields, and has achieved great success. This article mainly introduces a neural network that is very effective for processing images-convolutional neural networks. The characteristic of convolutional neural networks is that the characteristics of each layer are obtained by the local area of the previous layer through the convolutional core excitation of shared weights. This feature makes convolutional neural networks more suitable for the learning and expression of image features than other neural network methods, and they have very good results on large-scale image data sets. This article will conduct comprehensive research on convolutional neural networks, which mainly includes the basic structure and working principle of convolutional neural networks, followed by examples of the application of convolutional neural networks.

2. Convolutional neural network and its structure

2.1 Convolutional neural network

Convolutional neural networks originally originated in 1962. Biologists Hubel and Wiesel [1] discovered a cell that covers the entire visual domain and is sensitive to local areas of the visual input space, called the sensory field. In 1980, Fukushima proposed a neurocognitive machine with a similar structure based on the sensory field [2]. Neocognitron is a self-organizing multi-layer neural network model that stimulates the response of the local sensory field of the previous layer to each layer. It is also the main learning method of convolutional neural networks in early learning. After that, in 1998, LeNet-5 proposed by Lecun[3] et al. used a back-propagation algorithm to conduct supervised training of neural network networks. The trained network converts the original image into a series of feature vectors through alternately connected convolutional layers and downsampling layers, and finally classifies the feature expression of the image through a fully connected neural network. This is the earliest convolutional neural network model. In 2012, A. AlexNet proposed by Krizhevsky [4] and others won the championship in the image classification competition of ImageNet, a large-scale image database, with a huge advantage of accuracy exceeding the second place by 11%, making convolutional neural networks the focus of academia, and it is widely used in optical character recognition, face recognition, image classification, identity recognition, traffic sign recognition, aircraft image recognition and even graph feature analysis.

Convolutional neural networks are a deep learning method specially designed for image classification and recognition developed on the basis of artificial neural networks.

Traditional artificial neural networks consist of three layers: an input layer and an output layer, with multiple hidden layers in the middle. There are several neurons in each layer, and each neuron in the latter layer between the two adjacent layers is connected to each neuron in the previous layer, and there's no association between the neurons within the same layer (*Figure 1*).

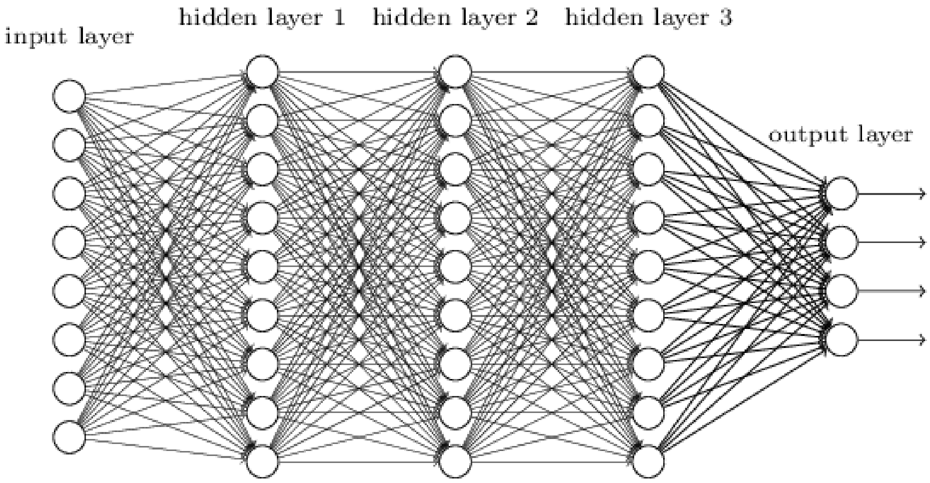

Figure- 1. An example of an artificial neural network model with an input layer, output layer, and 3 hidden layers [5].

When this kind of neural network is trained, each neural network can be used, but because the neurons in the two adjacent layers are all connected, the image processing speed will be limited by the many parameters generated. Therefore, based on artificial neural networks, convolutional neural networks add partially connected convolutional layers and dimension reduction layers in front of the original fully connected layers, and all the upper and lower neurons of CNN are not directly connected, but through the ‘convolution kernel’ as an intermediary. The same convolution kernel is shared among all images, and the image still retains its original positional relationship after the convolution operation. For images, if there is no convolution operation, the number of parameters learned is disaster-level. The reason why CNN is used for image recognition is precise because the CNN model limits the number of parameters and excavates the characteristics of the original structure. In addition to snappily training images through this special convolutional structure, CNN can also reduce the quantum of memory enthralled by the deep network and palliate the overfitting problem of the model.

At the same time, compared with fully connected neural networks similar to CNN, CNN has the characteristics of local connectivity and weight sharing. This kind of neural network not only reduces the training parameters a lot but also conforms to the characteristics that the closer the pixels in the natural image, the greater the impact on the pixels next to it. The weight sharing of the convolutional neural network constitutes the convolutional nucleus, and after it is convoluted with a given image, certain features of the image can be extracted. In the process of image processing, through the training of many different convolution kernel parameters, different image features of the same image can be automatically extracted [6].

2.2 Structure of convolutional neural network

The structure level of convolutional neural networks is more complex than traditional neural networks. The convolutional neural network consists of an input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer to form the basic structure. The input and output layers are responsible for receiving and outputting. Convolutional layers and pooling layers generally alternate in pairs. Since the neurons of the characteristic surfaces in the convolutional layer are locally connected to their inputs, the corresponding weights are weighted and summed with the local inputs, and the bias values are added to obtain the input values of the neurons. This process is equivalent to the convolutional process, and the convolutional neural network is also named after it.

Convolutional neural networks belong to feedforward neural networks. Through convolution, pooling, and nonlinear activation perform mapping, high-level semantic information is stripped from the original data layer by layer, which is a feedforward operation. After each convolutional layer, there will be a pooling layer. This is because the information is mapped from low-dimensional to high-dimensional at this time. There are many parameters and the dimension is too high. It is not suitable as the input of the next layer of neurons, so the output of this layer must be processed in a dimensionality reduction, so the pooling operation is introduced. However, it's easy to get overfitting and indeed lead to a dimensional disaster, If you don't reduce the dimension of the data.

3. Principles of convolutional neural network extraction of image features and recognition

Convolutional neural network recognition of an image divides a complete picture into many small parts, extracts the features in each small part, after multiple or parallel automatic feature extraction, and then summarizes each image feature together. The image recognition process can be completed with a high degree of accuracy once the similarity is compared.

3.1. Extraction and recognition of image features

3.1.1Establish a convolutional layer to extract preliminary features.

After the input layer converts an input picture into numerical data that the computer can understand, the convolutional layer will divide the picture data received from the input layer into blocks and extract features from each block. The convolutional nucleus is at the core of the convolutional layer and is responsible for extracting local features from the picture. Each convolution core has a constant bias in the convolution layer, which is a numerical matrix. The elements of the matrix plus the bias contribute to the weight of the convolution layer, and the weight is used for updating the network iteratively. Weight sharing and a local sensory field each convolution operation only needs to consider the colors, contours, textures, and other information present in the part of the area where it is performed; the size of the local sensory field is the scope of the convolution kernel. There is only one convolution layer. In multilayer convolutional networks, it can be fed back layer by layer. As a result of repeated iterations, the size of the sensory field in the original input image, as well as that of the multi-layer convolution layer, can be determined. All previous convolution layers of the layer are related to the convolution kernel size. This is known as weight sharing. Each convolution kernel is unchanged except for the weight update after each iteration. If the value of each convolution kernel is different, the convolution kernel is different as well. The convolution kernels extract features from the image if some convolution kernels extract the color characteristics of the image, contour characteristics, etc. The convolutional layer is made up of multiple feature surfaces. Multiple neurons from each feature surface and each of these neurons are connected to the local area of the feature surface of the previous layer through the convolutional nucleus. The feature map of the previous layer is convoluted by a learnable convolution kernel, and then the output feature map is obtained through an activation function. In the output feature map, the values of multiple feature maps can be combined and convolved.

3.1.2 Create a pooling layer to extract the key features.

After each convolutional layer, there will be a pooling layer. This is because the information is mapped from low-dimensional to high-dimensional at this time. There are many parameters, and the dimension is too high. It is not suitable as the input of the next layer of neurons, so the output of this layer must be processed in a dimensionality reduction, so the pooling operation is introduced. The pooling layer is more effective than convolutional in reducing the dimensions of data. The pooling layer obtains features with spatial immutability by reducing the resolution of the feature surface, which plays the role of secondary feature extraction. Doing so can not only greatly reduce the number of calculations. If the pooling layer does not remove the unimportant feature information in the feature matrix, it can easily cause overfitting and even lead to dimensional disaster. The commonly used downsampling layers are maximum downsampling, random downsampling, mean downsampling, etc. Mean downsampling is to take the average value of the downsampling part to replace all the values in this part; when maximum downsampling is performed, the maximum value is taken in the sampling area; when random downsampling is performed, random values are taken in the sampling area according to certain algorithm criteria.[7] The main roles of downsampling are to reduce the spatial size of the data body, reduce the number of parameters in the network, reduce the overhead of computing resources, and more effectively control overfitting.

3.1.3 Establish a full connection layer to summarize the functions of each part

In the convolutional network structure, at least one fully connected layer is connected after the last pooling layer. Each neuron in the completely connected layer is completely connected to all neurons in the former layer. The fully connected layer can integrate local information with category differentiation in the convolutional layer or the pooling layer. The output value of the last completely connected layer is passed to the output layer. In actual use, the fully connected layer can be realized by a global convolution operation; that is, a convolution kernel of the same size as the image matrix output of the previous layer is used to do convolution operations with the output of the previous layer so that a matrix can be mapped to a number, and a combination of multiple such convolution kernels can map the image matrix output of the previous layer into a fixed-length feature vector. In general, the length of the feature vector corresponds to the number of categories classified. This feature vector is obtained by highly purifying the image features obtained through multiple convolutional layers, pooling layers, and activation functions. This feature vector has high-level feature information, that is, it contains the combined information of all the features of the input image after various operations. This information is the most characteristic feature of the image. Therefore, the image can be classified by outputting the probability value of the specific category to which the image belongs through this information.

From the different and common features of an image, a neural network can be used to identify a specific image among tens of millions. After analyzing the image processing process of convolutional neural networks, the special hierarchical structure of CNN has a very good effect on image feature extraction. In summary, the convolutional layer of CNN is responsible for extracting local features in the image; the pooling layer is used to significantly reduce the order of parameters (dimensionality reduction), and the fully connected layer is similar to the part of a traditional neural network and is used to output the desired results.

3.2 Example of building a simple convolutional neural network model to process images

To realize the training operation of convolutional neural networks, this paper builds an image classification model of convolutional neural networks based on the TensorFlow deep learning framework and studies the steps of data preprocessing, model design, and construction, iterative training, and predictive evaluation. Compared with TensorFlow, Keras uses the least program code and takes the least time to make a deep learning model, train, estimate the delicacy, and make prognostications. Enter the CIFAR-10 data set in the Python environment, normalize the picture, data enhancement, and other preprocessing, and use Keras to construct an improved

VGG16 convolutional neural network structure to model and predict the CIFAR-10 image data set, and finally compare the accuracy and many different models by comparing different batch sizes.

Keras is an open-source advanced deep learning library. Its design refers to Torch, written in Python language, supports GPU and CPU, and is a highly modular neural network library. At present, Keras provides two back-end engines: Theano and TensorFlow. On top of the two, Keras provides APIs that allow users to focus more on model design and conduct model experiments faster. These APIs encapsulate many small components from TensorFlow and Theano in the form of modules, so the network that can be built using these two can also be built through Keras, and there is basically no performance loss. The biggest advantage of using the Keras framework is that it can save more time when building a new network structure.

3.2.1. Development environment

This practice uses the Python + TensorFlow + Keras development environment for programming and model training. Among them, the Python programming language has a clear structure, a rich standard library, and a strong third-party ecosystem, which can efficiently implement complex machine learning algorithms; TensorFlow is a powerful deep learning open-source framework developed by Google, which can easily perform high-performance numerical calculations; Keras belongs to TensorFlow's advanced API, which encapsulates multiple module components for deep learning, which can efficiently and quickly build complex neural network models.

3.2.2 Model design

The CIFAR-10 data set contains 60,000 natural images, which are divided into 10 types. Contains 50,000 training pictures and 10,000 test pictures. The data in the data set exists in an array (stored in rows, each row represents an image). The first 1024 bits are the R-value, the middle 1024 bits are the G value, and the last 1024 bits are the B value. In this article, the experimental data set is simply cut and whitened, and the pixel values are sent to the neural network for training[8]. In this article, a deep convolutional network model is designed based on the CIFAR-10 data set. The architectural parameters of the model are shown in figure 2.

```
8   def create_model(input_shape):
        # building the model
        model = Sequential()
        model.add(Conv2D(filters=32, kernel_size=(3, 3), padding="same", input_shape=input_shape))
        model.add(Activation("relu"))
        model.add(Conv2D(filters=32, kernel_size=(3, 3), padding="same"))
        model.add(Activation("relu"))
        model.add(MaxPooling2D(pool_size=(2, 2)))
        model.add(Dropout(0.25))
        model.add(Conv2D(filters=64, kernel_size=(3, 3), padding="same"))
        model.add(Activation("relu"))
        model.add(Conv2D(filters=64, kernel_size=(3, 3), padding="same"))
        model.add(Activation("relu"))
        model.add(MaxPooling2D(pool_size=(2, 2)))
        model.add(Dropout(0.25))
        model.add(Conv2D(filters=128, kernel_size=(3, 3), padding="same"))
        model.add(Activation("relu"))
        model.add(Conv2D(filters=128, kernel_size=(3, 3), padding="same"))
        model.add(Activation("relu"))
        model.add(MaxPooling2D(pool_size=(2, 2)))
        model.add(Dropout(0.25))
        # flattening the convolutions
        model.add(Flatten())
        # fully-connected layer
        model.add(Dense(1024))
        model.add(Activation("relu"))
        model.add(Dropout(0.5))
        model.add(Dense(num_classes, activation="softmax"))
        # print the summary of the model architecture
        model.summary()
        # training the model using adam optimizer
        model.compile(loss="sparse_categorical_crossentropy", optimizer="adam", metrics=["accuracy"])
        return model
```

Figure- 2. The CNN model structure in python environment (In order to train the model for accuracy, the Adam optimizer is used)

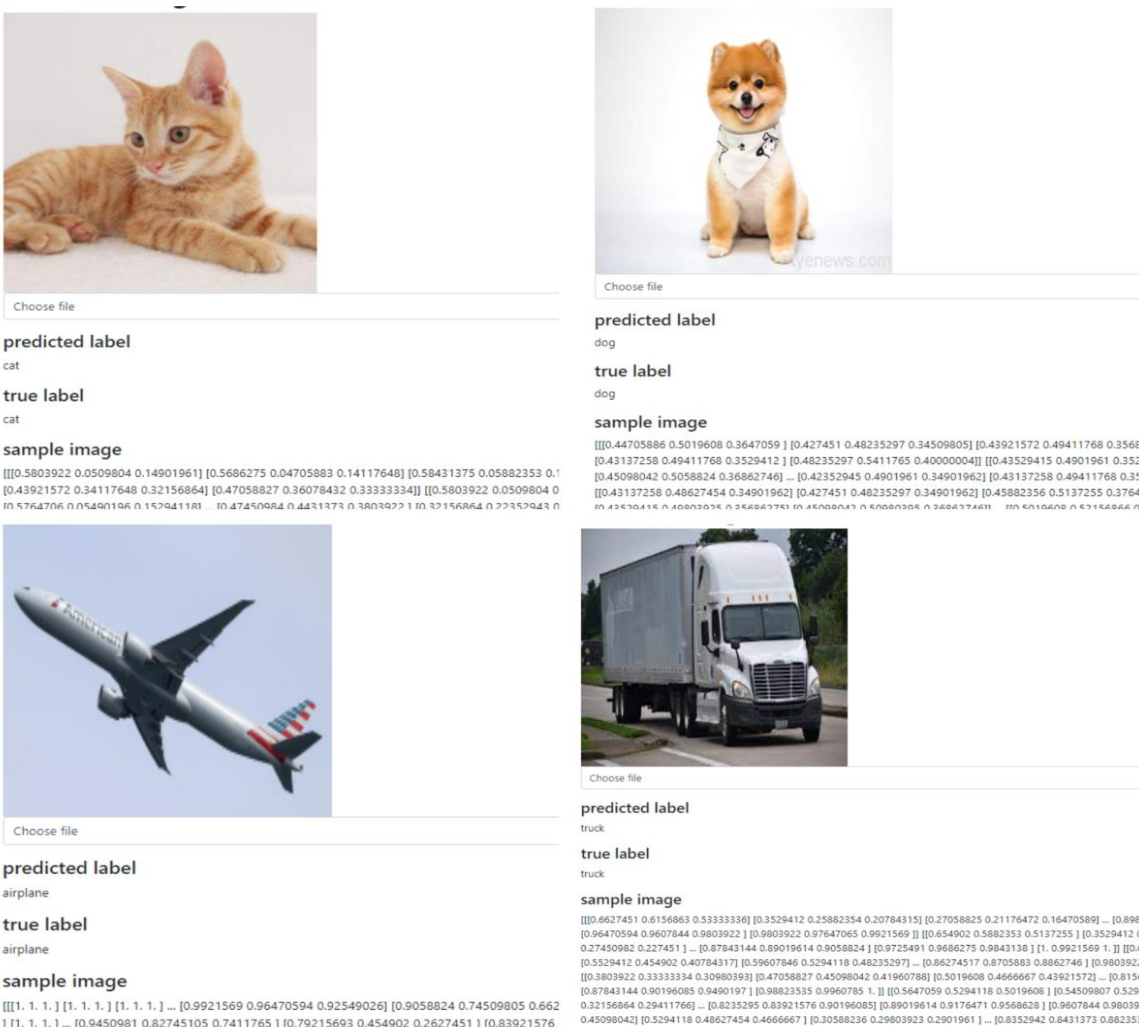

Figure- 3. A simple image recognition Web page based on the Python+TensorFlow+Keras development environment, some of the results obtained are shown

3.2.3 Example analysis

In this paper, the TensorFlow deep learning framework is used to train the CNN model, and it is trained and tested on the CIFAR-10 data set. The training method of cross-verification of the data set is used to iteratively obtain the optimal model (figure 3).Through practice, although the model training method has not reached the high accuracy rate of the CIFAR-10 data set, it provides some ideas for the realization and training of the CNN model.

4. Conclusions

This paper studies the image processing technology grounded on convolutional neural networks, combs the convolutional neural networks and their structural characteristics, builds a simple CNN model, combines the existing image algorithms based on convolutional neural networks, and uses CIFAR-10 as the data set to train its application analysis in image feature extraction and recognition. It can be concluded that compared with traditional neural networks, and convolutional neural networks, the convolutional structure of CNN and its own local connectivity and weight sharing characteristics greatly reduce the complexity of the network model. When entering a multidimensional image, the advantages of this feature are even more obvious. It avoids the process of feature extraction and data reconstruction in traditional recognition algorithms, making the calculation more concise. In general, convolutional neural networks have unique advantages in the field of image processing and computer vision.

References

[1] Hubel D. H, Wiesel T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of physiology, 1962, 160(1). 106.

[2] Fukushima K, Miyake S. Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position[J]. Pattern recognition, 1982. 15(6). 455-469.

[3] Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P. Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE, 1998. 86. 2278-2324.

[4] Krizhevsky A, Sutskever I, Hinton G. E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012. 25.

[5] Dürr O, Sick B, Murina E. Probabilistic deep learning: With python, keras and tensorflow probability [M]. Manning Publications, 2020.

[6] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. *86*(11). 2278-2323. https://doi.org/10.1109/5.726791

[7] Shahid Md. https://towardsdatascience.com/covolutional-neural-network-cb0883dd6529,2021.4.

[8] Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images[J]. 2009.

КОНВУЛЬСИЯЛЫҚ НЕЙРОНДЫҚ ЖЕЛІ НЕГІЗІНДЕ КЕСКІНДЕРДІ ӨҢДЕУ ТЕХНОЛОГИЯСЫ

Бақытнұр Ш., Әбілова П.Н., Тасболатұлы Н.
Астана Халықаралық университеті, Нұрсұлтан, Қазақстан
shuakbakytnur@gmail.com
ORCID ID:https://orcid.org/0000-0002-0511-7000

Аңдатпа. Конволюциялық нейрондық желі жасанды нейрондық желі мен терең оқуды біріктіретін бірнеше жасырын қабаттары бар нейрондық желі құрылымы. Қазіргі уақытта конвульсиялық нейрондық желі әр түрлі компьютерлік салаларда кеңінен қолданылатын және кескін туралы ақпаратты өңдеуде тиімді терең оқыту әдісі ретінде жаттығудан кейін кескіннің ерекшеліктерін зерттеп, кескін белгілерін алу мен жіктеуді аяқтай алады. Сондықтан ол кескіндерді өңдеу және үлгіні тану сияқты компьютерлік көру зерттеулерінде кеңінен қолданылады. Конвульсиялық нейрондық желінің жалпы құрылымы әдеттегі жасанды нейрондық желіге өте ұқсас болғанымен, ол үлкен масштабтағы кескін мәліметтерімен жақсы жұмыс істейді. Бұл мақаланың мақсаты-конвульсиялық нейрондық желіні, оның құрылымдық сипаттамалары мен артықшылықтарын түсіну, конвульсиялық нейрондық желінің негізгі принципін ұсыну және талдау, сонымен қатар кескіннің белгілерін алу және тану үшін CNN моделін оқытудың мысалын қолдану.

Түйін сөздер: терең оқыту; конвульсиялық нейрондық желі; суретті өңдеу; суретті тану.

ТЕХНОЛОГИЯ ОБРАБОТКИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ

Бакытнур Ш., Абилова П.Н., Тасболатулы Н.
Международный университет Астана, Нурсултан, Казахстан
shuakbakytnur@gmail.com
ORCID ID:https://orcid.org/0000-0002-0511-7000

Аннотация. Сверточная нейронная сеть структура нейронной сети с несколькими скрытыми слоями, которая сочетает в себе искусственную нейронную сеть и глубокое обучение. В настоящее время сверточная нейронная сеть, как метод глубокого обучения, широко используемый в различных компьютерных областях и эффективный при обработке информации об изображении, может изучать особенности изображения после обучения и завершать извлечение и классификацию признаков изображения. Поэтому он широко используется в исследованиях компьютерного зрения, таких как обработка изображений и распознавание образов. Хотя общая структура сверточной нейронной сети очень похожа на обычную искусственную нейронную сеть, она прекрасно работает с крупномасштабными наборами данных изображений. Цель этой статьи - разобраться в сверточной нейронной сети, ее структурных характеристиках и преимуществах, представить и проанализировать основной принцип работы сверточной нейронной сети, а также реализовать пример применения обучения модели CNN для извлечения и распознавания признаков изображения.

Ключевые слова: Глубокое обучение; Сверточная нейронная сеть; обработка изображений; распознавание изображений.

Сведения об авторах

Анг.: Bakytnur Shuak - Student's Astana international university, Nur-Sultan, Kazakhstan.

Каз.: Бақытнұр Шуақ- Астана Халықаралық университеті студенті, Нұр-Сұлтан, Қазақстан.

Рус.: Бакытнур Шуак, студент Международного университета Астана, Нур-Султан, Казахстан.

Анг.: Abilova Perizat Nurkhatovna - Senior lecturer, master's degree Astana international university, Nur-Sultan, Kazakhstan.

Каз.: Абилова Перизат Нурхатқызы - Астана Халықаралық университеті аға оқытушысы, Магистр, Нұр-Сұлтан, Қазақстан.

Рус.: Абилова Перизат Нурхатовна- магистр, старший преподаватель Международного университета Астана, Нур-Султан, Казахстан.

Анг.: Abilova Perizat Nurkhatovna - Senior lecturer, PhD, Astana international university, Nur-Sultan, Kazakhstan.

Каз.: Тасболатұлы Нұрболат - Астана Халықаралық университеті аға оқытушысы, PhD, Нұр-Сұлтан, Қазақстан.

Рус.: Тасболатулы Нурболат - PhD, старший преподаватель Международного университета Астана, Нур-Султан, Казахстан.