applied sciences

MDPI

Article

The Influence of the Activation Function in a Convolution Neural Network Model of Facial Expression Recognition

Yingying Wang [1], Yibin Li [1], Yong Song [2,*] and Xuewen Rong [1]

[1] School of Control Science and Engineering, Shandong University, Jinan 250061, China; yywang89@126.com (Y.W.); liyb@sdu.edu.cn (Y.L.); rongxw@sdu.edu.cn (X.R.)
[2] School of Mechanical, Electrical and Information Engineering, Shandong University, Weihai 264209, China
* Correspondence: songyong@sdu.edu.cn

Received: 13 February 2020; Accepted: 5 March 2020; Published: 10 March 2020

Abstract: The convolutional neural network (CNN) has been widely used in image recognition field due to its good performance. This paper proposes a facial expression recognition method based on the CNN model. Regarding the complexity of the hierarchic structure of the CNN model, the activation function is its core, because the nonlinear ability of the activation function really makes the deep neural network have authentic artificial intelligence. Among common activation functions, the ReLu function is one of the best of them, but it also has some shortcomings. Since the derivative of the ReLu function is always zero when the input value is negative, it is likely to appear as the phenomenon of neuronal necrosis. In order to solve the above problem, the influence of the activation function in the CNN model is studied in this paper. According to the design principle of the activation function in CNN model, a new piecewise activation function is proposed. Five common activation functions (i.e., sigmoid, tanh, ReLu, leaky ReLus and softplus–ReLu, plus the new activation function) have been analysed and compared in facial expression recognition tasks based on the Keras framework. The Experimental results on two public facial expression databases (i.e., JAFFE and FER2013) show that the convolutional neural network based on the improved activation function has a better performance than most-of-the-art activation functions.

Keywords: facial expression recognition; convolutional neural network; activation function

1. Introduction

As is known to all, the development of computer technology has promoted the considerable progress of many different fields, such as artificial intelligence, pattern classification, machine learning and other research fields. A harmonious human–computer relationship is a necessary condition for achieving natural interaction. Mehrabiadu [1] pointed out that facial expressions convey 55 percent of the useful information in communication, while sound and language only convey 38 percent and seven percent, respectively. Therefore, a wealth of emotional information is passed by facial expressions. In order to realise a more intelligent and natural human-machine interaction, facial expression recognition has been widely studied in the past few decades [2–5], and it has attracted more and more researchers' attention. S Poria et al. [6] proposed a novel methodology for multimodal sentiment analysis, and this method consisted of harvesting sentiments from Web videos. Chaturvedi et al. [7] used deep learning to extract features from each modality and then projected them to a common AffectiveSpace that was clustered into different emotions.

In the era of big data, traditional machine learning methods cannot meet the needs of timeliness, performance and intelligence. Deep learning [8] has shown excellent information processing

capabilities, especially in classification, identification and target detection. More abstract high-level features or attribute features can be formed based on deep learning, which will improve the final accuracy of classification or prediction. The convolutional neural network [9], as a special deep learning architecture, can extract image features accurately. It has been widely used in academic circles and practical industrial applications, especially in different areas of the computer vision filed. Hayit Greenspan et al. [10] proposed an overview and the future promise of medical image analysis based on CNNs and other deep learning methodologies. Masoud Mahdianpari et al. [11] proposed a detailed investigation of state-of-the-art deep learning tools for classification of complex wetland classes using multispectral RapidEye optical imagery, and examined the capacities of seven well-known deep ConvNets, namely, DenseNet121, InceptionV3, VGG16, VGG19, Xception, ResNet50, and InceptionResNetV2, for wetland mapping in Canada. M Baccouche et al. [12] proposed a fully automated deep model, which learns to classify human actions without using any prior knowledge. In view of the advantages and applications of CNN in image recognition, this paper proposes a facial expression recognition method based on the CNN model.

The convolutional neural network is a non-fully connected multilayer neural network, which is generally composed of a convolution layer (Conv), down-sampling layer (or pooling layer) and full-connection layer (FC). Firstly, the raw image is convoluted by several filters on the convolution layer, which can get several feature maps. Then the feature is blurred by the down-sampling layer. Finally, a set of eigenvectors is acquired through a full connection layer. The architecture of convolutional neural network is represented in Figure 1.

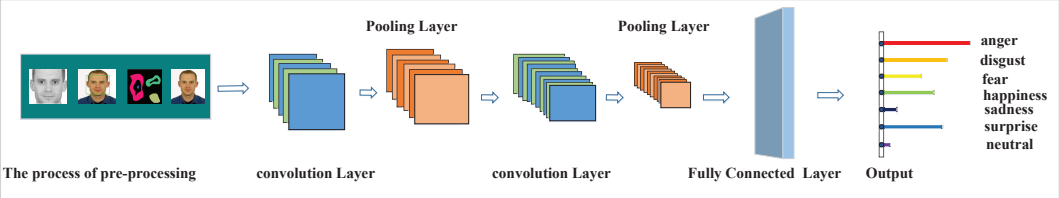


Figure 1. The general structure of a convolutional neural network.

In the practical application of the CNN model, there is a lot of room for improvement due to its complex structure. Many researchers have made a lot of effective ways to improve the recognition results of the CNN model. Some studies have been done on image classification methods [13,14]. Some studies have been done on the design of adaptive learning rate [15–17]. There are other studies which have been done on the design in the dropout layer [18–20]. All these above methods have improved the expression ability of the convolutional neural network to some extent. For the CNN model, the activation function is its core, which can activate the feature of neurons to solve nonlinear problems. A proper activation function has a better ability to map data in dimensions [21,22]. When the network has linear properties, the linear equation of the function and its combination only have the ability of linear expression, which will make the multilayer of the network have no meaning. An activation function is used to increase the expression ability of a neural network model, which can make the deep neural network truly have the significance of artificial intelligence. Considering the importance that the activation function plays in convolutional neural networks, the influence of the activation function on the recognition accuracy rate of facial expressions is studied in this paper.

The sigmoid function [23] and the tanh function [24] have been widely used in the convolution classification model during the beginning of deep learning research, but all of them are easy to make the convolution model appear the phenomenon of gradient diffusion. The coming of ReLu function [25] has effectively solved the above problem, and it has good sparsity. Krizhevsky et al. [26] firstly used ReLu as the activation funcation in the competition of ImageNet ILSVRC in 2012. Among common activation funcations, ReLu is best of them, but this function also has some shortcomings. Because the gradient of this function in the negative value is zero, and neurons in CNN model may undergo the phenomenon of "necrosis" during the training process.

Between 2013 and 2015, some researchers have proposed improved activation functions based on the phenomenon of "necrosis" that brought by ReLu function, such as: leaky ReLus [27], ELU [28], PReLu [29], tanh-ReLu [30], and so on.

Although great success have been made by the above improved functions in some special fields, the recognition result is unsatisfactory in facial expression recognition in this paper. In order to improve the accuracy rate of recognizing facial expressions, the influence and design principle of activation function in CNN model is studied, and a new activation function activation function is proposed in this paper. Experimental results on multiple facial expression data sets show that the accuracy rate by using the new activation function is much higher than that using common activation functions. With the same learning rate, the new function makes the model converge faster than other activation functions.

The paper is arranged as follows: After this introduction, related Work (Section 2) presents the importance of the activation function in the CNN model and some common activation functions, meanwhile, the design principle of the activation function and an improved activation function is proposed in this section. Section 3 focuses on the structure of the CNN model that used in this paper. Section 4 shows multiple experimental results. Finally, Section 5 summarizes and concludes this paper.

2. Activation Functions

The activation function refers to the feature of activated neurons can be retained and mapped out by a non-linear function, which can be used to solve nonlinear problems. The activation function is used to increase the expression ability of the neural network model, which can make the neural network has the meaning of artificial intelligence.

2.1. The Significance of the Activation Function

Since deep learning was put forward by Hinton in 2006, many researchers have made innovations from different directions for convolutional neural network. This paper mainly studies the effect of the optimization of activation function on improving the accuracy rate in facial expression classification. For a single layer perceptron [31], the binary classification operation can be easily performed, which can be seen in Figure 2.

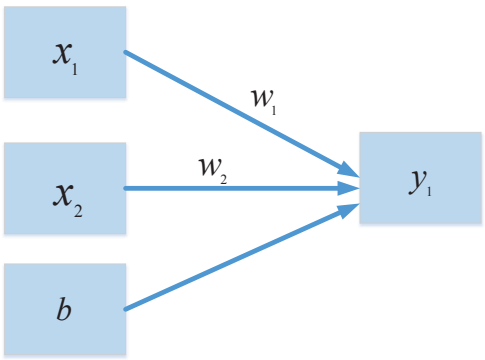


Figure 2. A single layer perceptron without activation function

In Figure 2, y_1 can be defined as:

$$y_1 = w_1 x_1 + w_2 x_2 + b \tag{1}$$

When $y_1 = 0$, the line for classification can be obtained. Since the problem of linear indivisibility cannot be handled by the single layer perceptron, the multiclass problem can be solved by the multilayer perceptron [32] based on Equation (2).

$$y_1 = \sum_{i=1}^{n} w_i x_i + b \tag{2}$$

But because the essence of the classifier is a linear equation, no matter the combination, it cannot deal with the classification problem of a non-linear system. Therefore, the activation function is introduced in the perceptron, which can be seen in Figure 3.

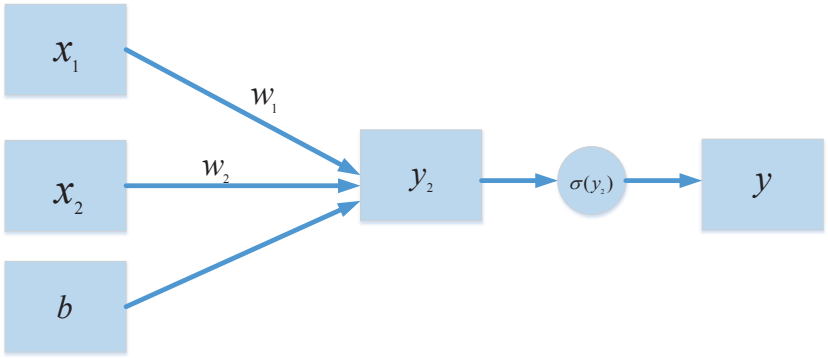


Figure 3. A single layer perceptron with an activation function.

In Figure 3, the output of the model can be defined as:

$$y_2 = w_1 x_1 + w_2 x_2 + b \tag{3}$$

$$y = \sigma(y_2) \tag{4}$$

The perceptron with an activation function can deal with the classification problem of the non-linear system by Equations (3) and (4).

2.2. The Comparison and Study of Traditional Activation Functions

The activation function is the core of a deep neural network's structure, and common activation functions include: sigmoid, tanh, ReLu and softplus, which can be seen in Figure 4.

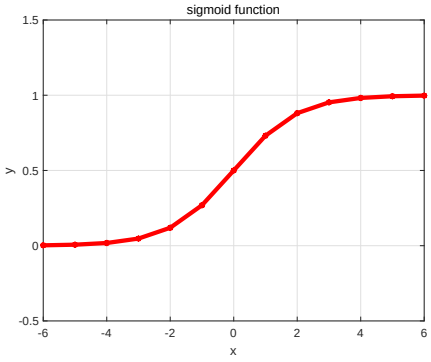
2.2.1. Common Activation Functions

The curve of sigmoid function is shown in Figure 4a, which is a common non-linear activation function. The output of this function is bounded, and it was widely used as the activation function in deep neural networks during the early age of deep learning. Although the characteristic of the sigmoid function is consistent with the synapses of neurons in neurology, and the derivative of this function is convenient to get, the function is rarely used nowadays due to its shortcomings. From the curve of sigmoid function, this function has the characteristic of soft saturability. That is, the slope of the graph tends to be zero when the input is very large or very small. When the slope of the function is close to zero, the gradient that passed to the underlying network becomes very small, which will make network parameters difficult to be trained effectively. Meanwhile, the direction of weight update only to one direction due to the output of this function is always positive, which will affect the convergence rate. The formula of sigmoid function is defined as:

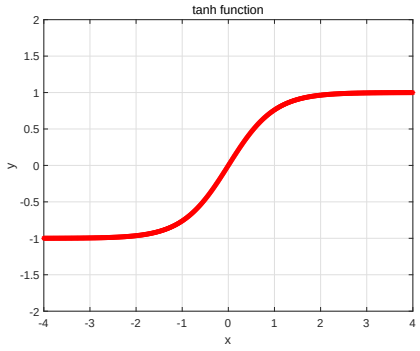
$$f(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

The tanh function is the updated version of the sigmoid function on the range, which is a symmetric function centred on zero. Its output is bounded, and it brings nonlinearity to the neural network. The curve of this function can be seen in Figure 4b. The convergence rate is higher than the sigmoid function, but the problem of gradient diffusion also exists. The formula of tanh function is defined as:

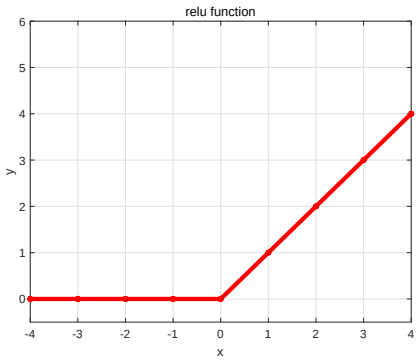
$$f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}} \tag{6}$$



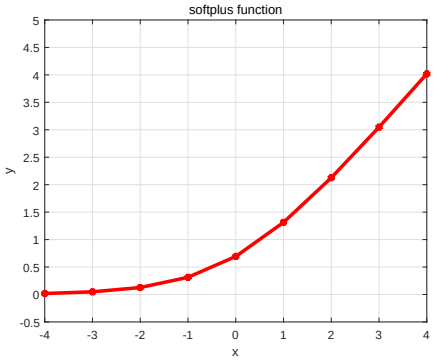
(**a**) The curve of sigmoid function



(**b**) The curve of tanh function



(**c**) The curve of ReLu function



(**d**) The curve of softplus function

Figure 4. The graphs of four common different activation functions. The names of the above curve functions are: the sigmoid function, the tanh function, the ReLu function and the softplus function.

The trendy activation function in neural network is the ReLu function, which is a piecewise function. The curve of this function is shown in Figure 4c. From the curve of this function, this function will force the output to be zero if the input value is less than or equal to zero. Otherwise, it will make the output value equal to the input value. The method of directly forcing some data to be zero can create a moderate sparse characteristic to some extent. Compared with the previous two functions, the ReLu function provides a much faster computing rate. Since ReLu is unsaturated, there is no gradient diffusion problem, unlike sigmoid and tanh functions. Although the ReLu function has great advantages, it also has some shortcomings. Since the derivative of the ReLu function is always zero when the input value is negative, it is likely to present neuronal necrosis when a neuron with a large gradient passes through the ReLu function, which will affect the final recognition result. The equation of this function is defined as:

$$f(x) = \max(0, x) \tag{7}$$

The softplus function, as shown in Figure 4d, is similar to ReLu function. From the curve of this function, the difference between the softplus function and the ReLu function can be seen clearly. This function has small reservations about values less than 0, which will decrease the possibility of neuronal death. But this function has much more computation than ReLu function. The formula of this function is defined as:

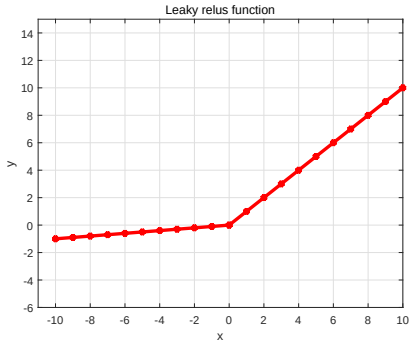
$$f(x) = \ln(1 + e^x) \tag{8}$$

In summary, the ReLu function is the best of the many extant activation functions. Although this function has many advantages in signal response, it only works in terms of forward propagation. It is easy to make the model output zero, and it cannot be trained again, because all the negative values are omitted. For example, if one value in the randomly initialised value (W) is negative, the characteristics

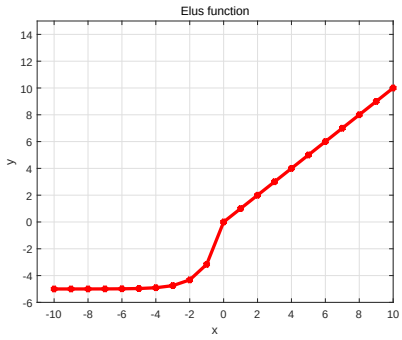
of the corresponding positive input are all shielded. In a similar way, the corresponding negative input values are activated instead. This is obviously not the desired outcome. Therefore, some variant functions have evolved on the basis of the original ReLu function.

2.2.2. Common Variations of ReLu Function

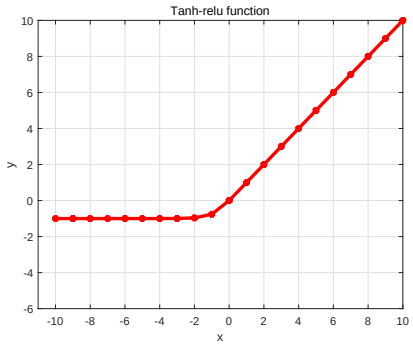
Many researchers have proposed some improved activation functions based on the phenomenon of "necrosis" that appeared with the ReLu function between 2013 and 2015, such as leaky ReLus, ELU, tanh-ReLu and softplus–ReLu. The curves of these variations can be seen in Figure 5.



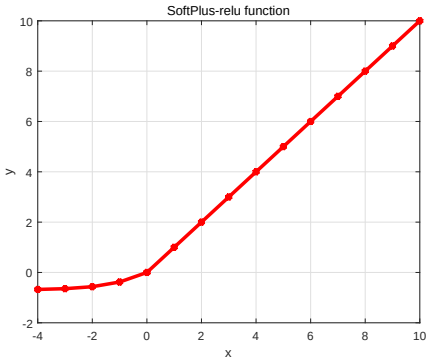
(**a**) The curve of leaky ReLus function



(**b**) The curve of Elus function



(**c**) The curve of Tanh-ReLu function



(**d**) The curve of softplus–ReLu function

Figure 5. The curves of variant functions based on the ReLu function.

The equations of these above activation functions are respectively defined as:

$$f(x) = \begin{cases} x & (x > 0) \\ a \cdot x & (otherwise) \end{cases} \tag{9}$$

$$f(x) = \begin{cases} x & (x \geqslant 0) \\ a \cdot (e^x - 1) & (otherwise) \end{cases} \tag{10}$$

$$f(x) = \begin{cases} \frac{1-e^{-2x}}{1+e^{-2x}} & (x < 0) \\ \max(0, x) & (otherwise) \end{cases} \tag{11}$$

$$f(x) = \begin{cases} \ln(1 + e^x) - \ln 2 & (x < 0) \\ \max(0, x) & (otherwise) \end{cases} \tag{12}$$

Although the above variation functions have achieved good recognition results on some data sets, the experimental results of facial expression recognition in this paper are not satisfactory. Therefore,

this paper analyses the activation function in the CNN model and designs a new activation function. Through the comparison of several experimental results, it is found that the performance of the new function is stable, and the accuracy of the test set is also improved to a certain extent on the basis of improving convergence.

2.2.3. Analysis and Research on the Design Method of Activation Function in the Convolution Neural Network Model

There are two parts in the training process of a convolutional neural network: forward propagation and back propagation. The forward propagation refers to a process in which the input signal passes through one or more network layers, and gets the actual output in the output layer. The back propagation is the process of making the actual output closer to the expected value by calculating the error between the actual output and desired output. By analysing the processes of forward propagation and back propagation, the role the activation function played in the training process of the convolutional neural network can be easily understood by us.

Since the activation function plays a similar role in each layer of the neural network model, this paper takes the convolutional layer as an example to analyse the role of the activation function in forward propagation and back propagation.

In the process of forward propagation, the output of the previous layer convolves by the convolution kernel, and the output of this layer can be gotten by the following equations:

$$u_j^l = \sum_{i \in M_j} x_j^{l-1} k_{ij}^l + b_j^l \tag{13}$$

$$x_j^i = f(u_j^l) \tag{14}$$

where x_j^{l-1} is the output feature map of the i channel in the previous layer; k_{ij}^l refers to the convolution kernel matrix; the net output u_j^l of the l layer can be calculated by the output feature map of the previous layer; the output x_j^l of the l layer and the j channel can be gotten through the activation function f .

According to Equations (13) and (14), the function of the activation function in the convolution layer is to reprocess the result of the convolution operation and sum up the convolution values during the forward propagation, which can make a nonlinearity relation between the input and the output of the convolution layer, and enhance the expression ability of features. The analysis of the forward propagation Illustrates that the activation function cannot be a constant function or other linear functions. Since each layer has an activation function, the output calculation of activation function should be as simple as possible to ensure the training speed of the model.

In the process of back propagation, the parameters for the convolution layer need to be tuned are convolution kernel parameters k and bias b. By calculating the loss between the actual output and the expected output and acquiring the partial derivatives of the loss, Δk and Δb can be gotten. The detailed process is listed as follows:

(a) Calculate the loss function E; the squared error loss function is selected in this paper.
(b) Calculate the sensitivity of the l layer. The sensitivity of the convolution layer l can be gotten by the sensitivity of the next sample layer $l+1$:

$$\delta_j^l = \frac{\partial E}{\partial u_j^l} = \beta_j^{l+1}(f'(u_j^l) o up(\delta_j^{l+1})) \tag{15}$$

where δ_j^l is the sensitivity of the j channel in l layer, β_j^{l+1} is the weight of the sample layer, f' is the derivative of the activation function, o refers to make multiply operation for each function and up stands for the up-sampling operation.

(c) The partial derivative of the parameter can be obtained by the sensitivity:

$$\frac{\partial E}{\partial b_j^l} = \sum_{u,v} (\delta_j^l)_{u,v} \tag{16}$$

$$\frac{\partial E}{\partial k_{ij}^l} = \sum_{u,v} (\delta_j^l)_{u,v} (x_i^{l-1})_{u,v} \tag{17}$$

(d) Update the parameters in the convolution layer

$$\Delta k_{ij}^l = -\eta \frac{\partial E}{\partial k_{ij}^l} \tag{18}$$

$$\Delta b^l = -\eta \frac{\partial E}{\partial b^l} \tag{19}$$

where η is the learning rate.

In the process of back propagation, there is a linear relationship between the final parameter update step size and the derivative of the activation function; therefore, the derivative of the activation function will directly affect the convergence speed of the convolutional neural network model. In the early training stage, parameters need to be updated quickly to the optimal values, which requires the derivative of the first half of the activation function to be big enough to accelerate the convergence speed of the model. Then the parameter update speed slows down and gradually approaches the optimal value, which requires the derivative of the second half of the activation function to be smaller and smaller, and the derivative approaches a value close to zero to ensure the convergence of the model.

From the above analysis, it can be seen that:

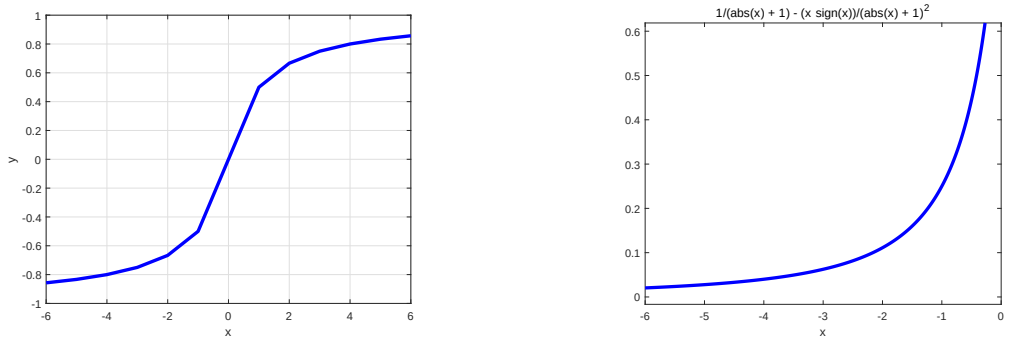
(1) The derivative of the first half of the activation function should be large enough to enable the parameters to be rapidly updated to the vicinity of the optimal value.
(2) The derivative of the second half gradually reduces to a value close to zero, so as to realise fine-tuning of parameters.

Based on the theory analysis above-mentioned, the derivative of the first half of the activation function should be large enough to enable the parameters to be rapidly updated to the vicinity of the optimal values. The derivative of the second half gradually reduces to a value close to zero, so as to realise fine-tuning of parameters. A single activation function cannot satisfy these two points at the same time; hence, the activation function needs to be pieced together to realise this request. A new piecewise activation function is proposed in this paper based on the above analysis. The first half curve is controlled by softsign function, and the curve of this function can be seen in Figure 6a. The equations of softsign function is defined as:

$$f(x) = \frac{x}{1 + |x|} \tag{20}$$

Compared with common activation functions sigmoid, tanh and softplus, the slope of the curve near 0 in the left half axis is larger, and it will approach the optimal value faster. The curve of the derivative of softsign function can be seen in Figure 6b, and its biggest advantage is that it can keep changing and maintain a value greater than zero, which makes the model continue to converge. The second half curve is controlled by ReLu function, which can reserve some good characteristics of ReLu function. But the combination function that is only composed of the softsign function and ReLu function cannot satisfy the design principle being analysed in Section 2.2.3, and the experimental results in this paper (using this combination function) are unsatisfactory. In order to slow the upward

trend of the curve for preventing the gradient explosion problem, an adjustable log function is added in the region of $x > k(k > 0)$.



(**a**) (**b**)

Figure 6. The curve of the softsign function. (**a**) The curve of the original function. (**b**) The derivative curve of softsign function in the negative-half axis.

The new activation function can be seen in Figure 7, and the equation of the new variation of the activation function can be defined as:

$$f(x) = \begin{cases} \dfrac{x}{1+|x|} & x \leqslant 0 \\ \max(0, x) & k \geqslant x > 0 \\ \log(a * x + 1) + |\log(a * k + 1) - k| & x > k \end{cases} \tag{21}$$

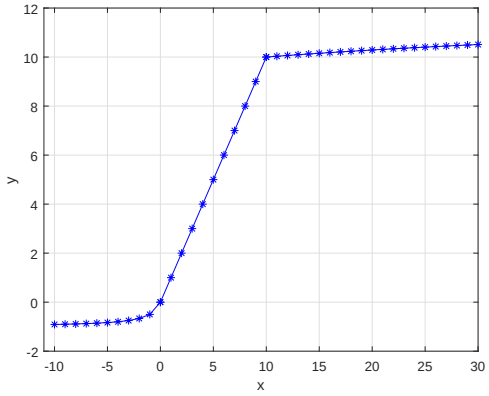


Figure 7. The new activation function.

Advantages of the the new activation function(SL-ReLu):

(1) With the gradual deepening of training, it is possible to bring about the problem of neuron death, which will make the weight fail to be updated normally. In order to solve the problem, the softsign function has been used in the region of $x < 0$. The addition of the new function can avoid the problem of mass neuronal death, because the new function is designed to selectively activate many negative values, not mask a large number of signals in the negative half of the shaft. From the trend of the derivative curve in Figure 6b, the derivative of the softsign function changes faster in the region near zero. This characteristic indicates that this function is more sensitive to data, and it is more beneficial to solve the gradient disappearance problem that is caused by the derivative at both ends being zero. In addition, in the negative axis, the derivative

of softsign function keeps changing, and decreases slowly, which can effectively reduce the occurrence of the phenomenon of non-convergence of the training model.

(2) ReLu function is used in the region of $0 < x < k$. Combined with the curve in the positive half axis of ReLu function, the combined activation function can have some characteristics of ReLu function. The new combined function accelerates the convergence speed of the model, and greatly reduces the possibility of gradient disappearance.

(3) Meanwhile, in order to reduce the problem of gradient explosion produced by a large amount of data in deep network, an adjustable log function is applied in the region of $x > k$. The aim of the log function is to slow the upward trend to prevent the gradient explosion problem that is brought about by the large amount of data in a deep network.

3. The Design of the CNN Model Based on the Transfer Learning Method

3.1. The Introduction of the Transfer Learning Method

Although the CNN model has a good performance in image classification tasks, a large number of training samples are needed in training process. In reality, the access to a large number of labelled training samples requires a large amount of manpower and material resources, which is difficult for this task. The use of the transfer learning method [33] solves the above problem, which allows the existing knowledge to be transferred to another similar task with a small number of labelled samples. The detailed description of transfer learning can be seen in Figure 8.

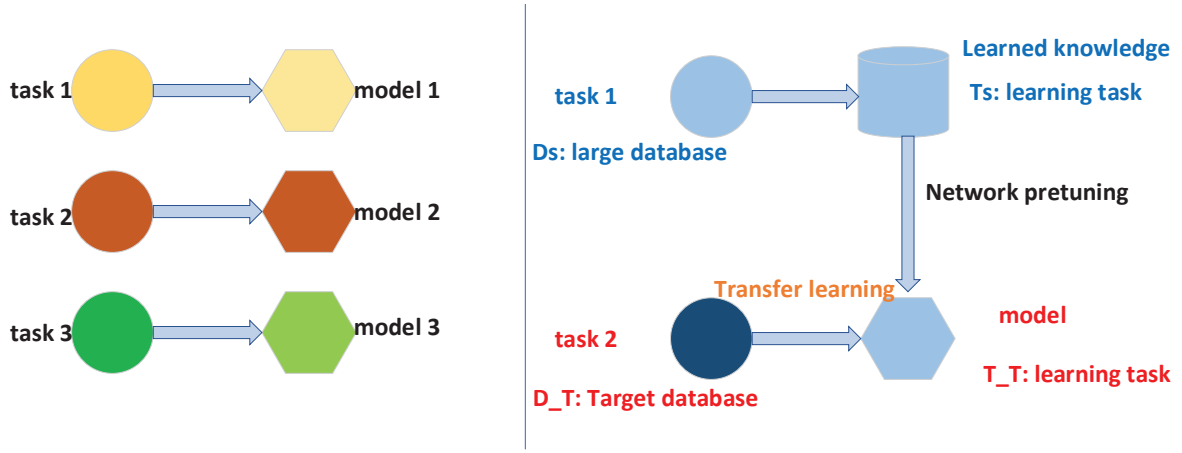


Figure 8. The comparison of two different learning methods. The learning method on the left is that every new leaning should start from zero. The learning method on the right is that each new learning is based on the previous experience, and it does not require a large training sample.

Transfer learning can be defined as: A source domain D_s, learning task T_s, a target domain D_T and learning task T_T. A deep CNN model is pretrained on the database in the source domain D_s based on the learning task T_s, and the pretrained deep CNN model will be retrained on the data set of the target domain D_T based on its learning task T_T. The final purpose of the transfer learning method is to use the existing knowledge in D_s and T_s to improve the learning ability of prediction function $f_T(\cdot)$ in the target domain D_T.

3.2. The CNN Model Based on the Transfer Learning Method

Inception-v3 has been trained by Google on the large image database (i.e., ImageNet), and can be used directly for image classification tasks. There are approximately 25 million parameters in this model, and 5 billion multiply and add instructions will be taken to classify one image. For a modern personal computer without a GPU, the Inception-v3 model can quickly classify an image. There are 15 million images that belong to 22,000 categories in the ImageNet dataset. Its subset contains 1 million

images and 1000 categories, which corresponds to the current most authoritative image classification competition, LSVRC. Several weeks may be spent on training this model with a normal personal computer (PC); therefore, it is not possible to train the deep model on a normal PC. The pretrained Inception-v3 model is used in this paper for facial expression classification. This pretrained model can be downloaded online and it can be used to classify facial expression images. The flowchart of the CNN model can be seen in Figure 9.

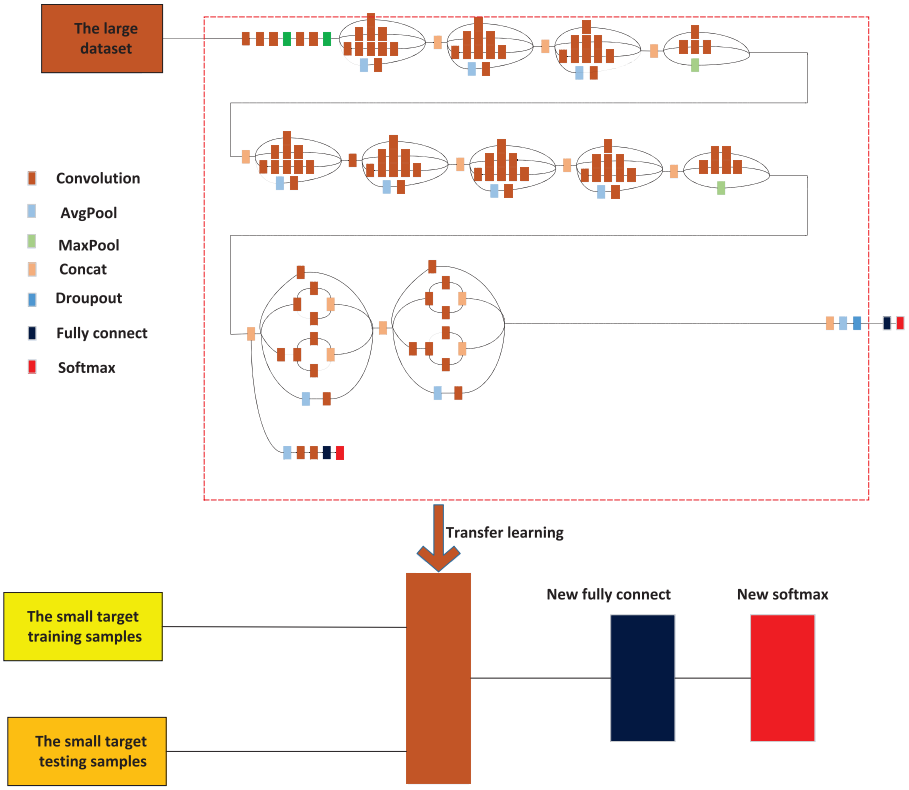


Figure 9. The flowchart of transfer learning method. The parameters in the red dotted line were trained in the large sample, and these parameters can be directly transferred to the new small training samples for fine-tuning, and the new fully connect layer and the new softmax layer are retrained by the new target training samples.

4. Experiments and Results

4.1. Databases

(1) JAFFE database [34]: The database was published in 1998, and it is a relatively small database. This database includes 213 images that were produced by 10 Japanese women, and each person has seven emotional images: disgust, anger, fear, happiness, sadness, surprise and neutrality. Since the size of this data set is too small to train a better convolutional neural network model, a reliable data augmentation method needs to be used for the database. Table 1 shows the number of new samples produced by some traditional data augmentation methods, such as the geometric transformation method and colour space method. Figure 10 shows some images regarding the database.

Figure 10. Some images of JAFFE database. From left to right: anger, disgust, fear, happiness, neutrality, sadness and surprise.

Table 1. This table describes the sample number of JAFFE database.

JAFFE Dataset	
Expression Labels	**Samples**
angry	6031
disgust	5264
fear	5911
happy	4640
neutral	5391
sad	5873
surprise	5907

(2) The Facial Expression Recognition 2013 (FER-2013) database [35]: The database includes 35,887 different images. The training set consists of 28,709 examples. The public test set used for the leaderboard consists of 3589 examples. The final test set consists of another 3589 examples. The data consists of 48 $\times$ 48 pixel grayscale images of faces. There are seven expressions labelled in this database: neutral, happy, sad, surprised, angry, disgusted and fearful. Table 2 shows the number of training samples in this database. Figure 11 shows some images about the database.

Figure 11. Some images of FER2013 database. From left to right: anger, disgusted, fearful, happy, normal, sad and surprised.

Table 2. This table describes the sample number of FER2013 database.

FER2013 Dataset		
Expression Labels	**Training Samples**	**Testing Samples**
angry	3995	491
disgust	436	55
fear	4097	528
happy	7215	879
normal	4965	626
sad	4830	594
surprise	3171	416

4.2. Results

Figure 12 shows the corresponding confusion matrixes with the learning rate 0.001. This paper adopts the method of cross validation to improve the reliability of the identification results. All facial expression samples were divided into two subsets: one is the sample set and the other is the test set. All face image samples were divided into five parts by using the method of k-fold cross validation (k

= 5), among which four parts were used as training samples and one part was used as test samples. The experiment was repeated for five times, and the mean value was taken as the final experimental result. Table 3 shows the detailed average accuracy rates based on cross validation method. Table 4 shows the comparison between the average accuracy rate produced by the new function and most state-of-the art-methods.

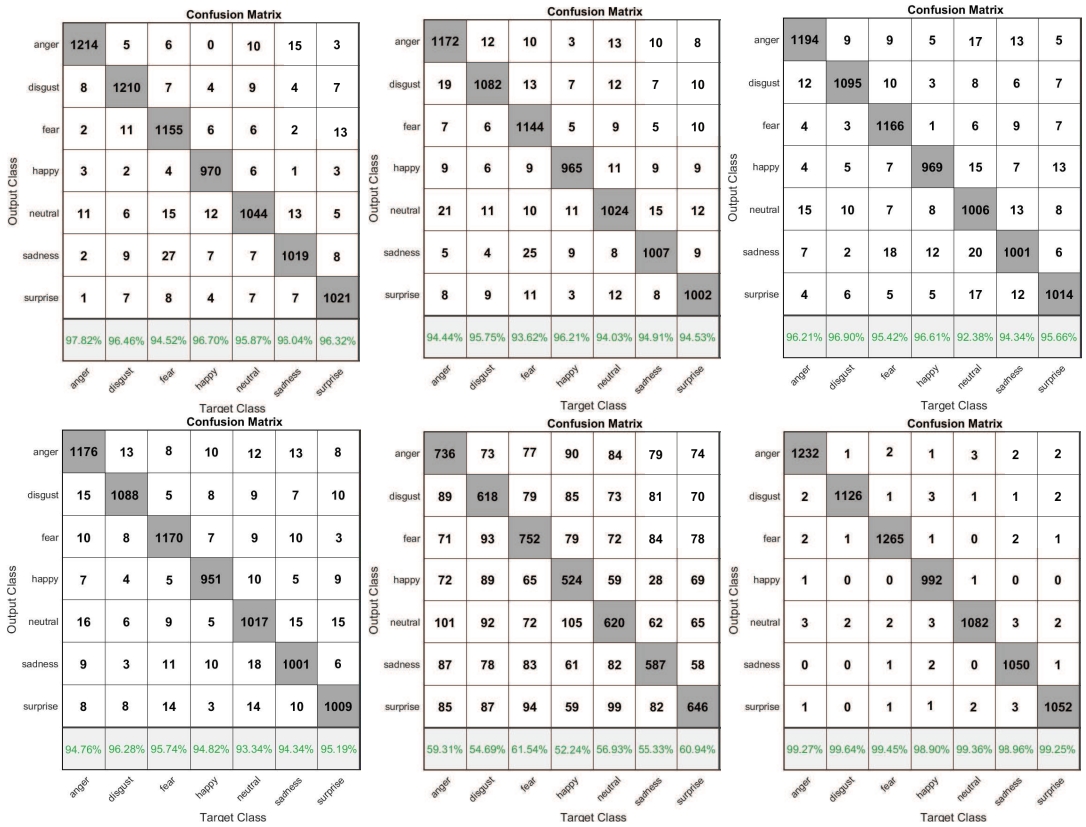


Figure 12. The confusion matrixes based on six different variant activation functions for JAFFE database. From left to right, the corresponding activation functions are: sigmoid, tanh, ReLu, leaky ReLus, softplus–ReLu and the new activation function (LS–ReLu).

Table 3. This table summarises the average accuracy rates produced by different common activation functions and the new activation function in facial expression recognition on JAFFE database.

JAFFE Dataset						
Learning Rate	**Sigmoid**	**Tanh**	**ReLu**	**LeakyReLus**	**Softplus-ReLu**	**New (LS–ReLu)**
learning rate = 0.001	96.25	94.75	95.38	94.95	57.43	99.91

Table 4. This table summarises the current state of the art in facial expression recognition on JAFFE database.

JAFFE Dataset		
Author	**Method**	**Acuuuracy (%)**
AI Abdullah [36]	FLDA+KNN	95.09
M.K.Mohd Fitri Alif [37]	Fused CNN	83.72
André TeixeiraLopes [38]	CNN+preprocessing	82.10
Ping Liu [39]	BDBN	68.00
CaifengShan [40]	LBP+SVM	41.30
Shan [41]	deep learning	99.3
Yee [42]	LapCLTP	98.78
Nandi [43]	Circumcenter-Incenter-Centroid Trio Feature	97.18
This paper	new method	99.66

From the confusion matrixes in Figure 12 and Table 3, the new activation function can ensure the training model a higher recognition rate than that of other common activation functions. From Table 3, the accuracy rate that produced by the new activation function is 3.66%, 5.16%, 4.53%, 4.96% and 42.48% higher than those of the sigmoid function, tanh function, ReLu function, leaky ReLus function and softplus–ReLu function. Table 4 indicates that the new method has a better performance than that of most state-of-the art-methods.

For FER2013 database, Figure 13 shows the corresponding confusion matrixes with the learning rate 0.001. Table 5 shows the detailed average accuracy rates based on Figure 13. Table 6 shows the comparison between the average accuracy rate produced by the new function and those of most state-of-the art-methods.

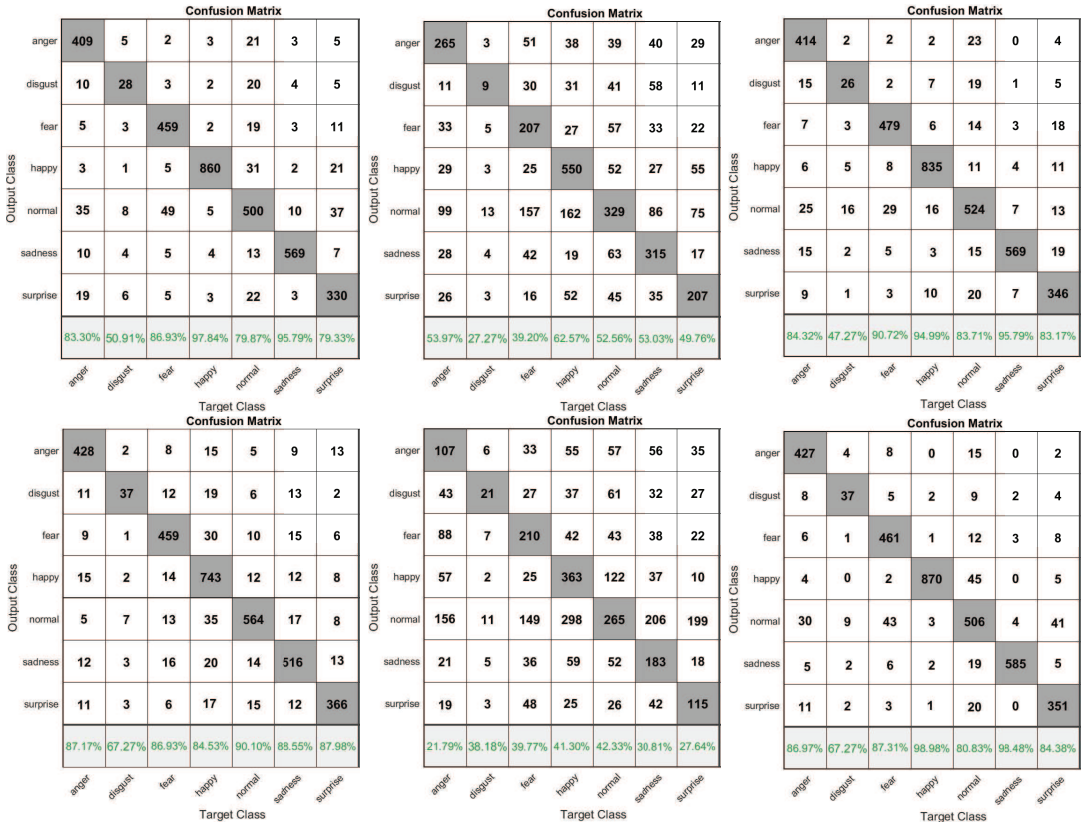


Figure 13. The confusion matrixes of the same CNN structure with six different variant activation functions for FER2013 database. From left to right, the corresponding activation functions are: sigmoid, tanh, ReLu, leaky ReLus, softplus–ReLu, and the new activation function (LS–ReLu).

Table 5. This table summarises the average accuracy rates produced by different common activation functions and the new activation function in facial expression recognition on FER2013 database.

FER2013 Dataset						
Learning Rate	**Sigmoid**	**Tanh**	**ReLu**	**LeakyReLus**	**Softplus-ReLu**	**New (LS–ReLu)**
learning rate = 0.001	87.91	44.52	80.33	86.74	35.22	90.16

From the confusion matrixes in Figure 13 and Table 5, the new activation function can ensure the training model a higher recognition rate than those of other common activation functions. From Table 5, the accuracy rate produced by the new activation function is 2.21%, 45.60%, 9.79%, 3.38% and 54.9% higher than those of sigmoid function, tanh function, ReLu function, leaky ReLu function and softplus–ReLu function. Table 6 indicates that the new method has a better performance than those of most state-of-the art-methods.

Table 6. This table summarises the current state of the art in facial expression recognition on FER2013 database.

FER2013 Dataset		
Author	**Method**	**Acuuuracy (%)**
Liu [39]	ECNN	69.96
Minchul Shin [44]	Raw+CNN	62.2
Minchul Shin [44]	Hist+CNN	66.67
Minchul Shin [44]	Is+CNN	62.16
Minchul Shin [44]	DCT+CNN	56.09
Minchul Shin [44]	DOG+CNN	58.96
Amani Alfakih [45]	multi-view DCNN	72.27
YAR H [46]	Gender+CNN	94
Agrawal [47]	CNN+kernel size and number of filters	65
this paper	new method	90.16

4.3. Extensibility

In order to verify the extensibility of the new activation function, a convolutional neural network model is constructed according to the structure design of convolutional neural network, which is used to verify the recognition effect of different activation functions in data sets. The The schematic diagram of the seven-layer convolutional neural network model is shown in the Table 7. Figure 14 shows the different recognition rates of the CNN model with different activation functions. From Figure 14, the new activation function still performs well.

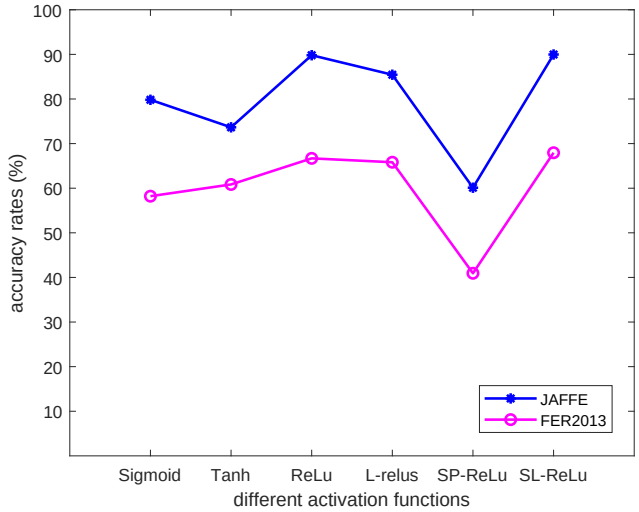


Figure 14. Experimental results based on the new created CNN model with six different activation functions.

At the same time, a face attitude dataset (CMU-PIE) [48] is used to further verify the influence of the new activation function on the convolutional neural network model, which can be seen in Figure 15. CMU-PIE face dataset includes 40,000 photos from 68 people, in five poses. Figure 16 shows the experimental results. From Figure 16, the the new activation function can make the CNN model have a good performance.

Table 7. The structure of convolutional neural network.

Layer	Input	Kernel Size	Output
Conv	96×96	5×5	92×92
Conv	92×92	5×5	88×88
Pool	88×88	2×2	44×44
Conv	44×44	3×3	42×42
Pool	42×42	2×2	21×21
Conv	21×21	3×3	19×19
Conv	19×19	3×3	17×17
Conv	17×17	5×5	13×13
Conv	13×13	3×3	11×11
Conv	11×11	2×2	5×5
FC			
Softmax			

Figure 15. Some images of CMU-PIE database.

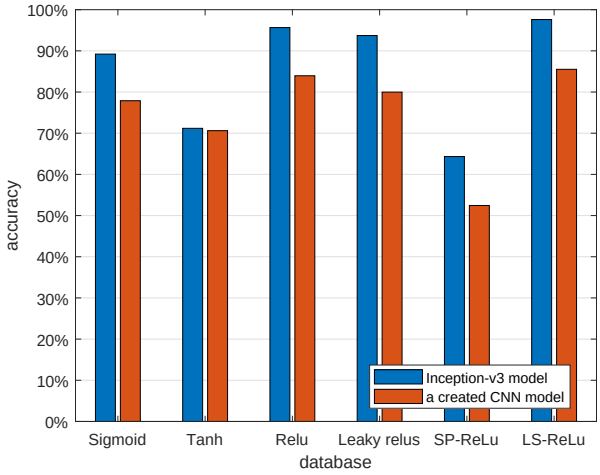


Figure 16. The recognition results of CMU-PIE database based on two different CNN models.

5. Conclusions

CNN model is widely used in image classification tasks. Due to the complexity of this model, there is a lot of room for improvement, and many researchers have proposed a number of methods to improve the accuracies of different CNN models. The activation function is an important part of convolutional neural network, which can map out the non-linear characteristic. From the perspective of the activation function, this paper studies the influence of the activation function in the CNN model on facial expression recognition, and proposes a new variant function based on the ReLu function. This new activation function not only preserves some of the features of ReLu function, but also makes full use of the advantages of an adjustable log function and the softsign function according to the design principle of the activation function. The neutral network based on LS–ReLu function can avoid the over-fitting problem of the model in the training process and reduce the oscillations problem. Figures 12 and 13 can demonstrate the advantages of the new function in detail. Tables 4 and 6 show that the facial expression recognition system proposed in this paper has a better performance than most state-of-the-art methods.

Author Contributions: Y.L. and Y.W. conceived the research and conducted the simulations; Y.W. designed and implemented the algorithm; Y.S. analyzed the data, results and verified the theory; X.R. collected a large number of references and suggested some good ideas about this paper; all authors participated in the writing of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Yibin Li, grant number 61673245. and This research was also funded by Yong Song, grant number 61573213.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mehrabian, A.; Russell, J.A. *An Approach to Environmental Psychology*; The MIT Press: Cambridge, MA, USA, 1974.
2. Wang, K.; Peng, X.; Yang, J. Region attention networks for pose and occlusion robust facial expression recognition. *IEEE Trans. Image Process.* **2020**, *29*, 4057–4069. [CrossRef] [PubMed]
3. Wood, A.; Rychlowska, M.; Korb, S.; Niedenthal, P. Fashioning the face: Sensorimotor simulation contributes to facial expression recognition. *Trends Cogn. Sci.* **2016**, *20*, 227–240. [CrossRef] [PubMed]
4. Otwell, K. Facial Expression Recognition in Educational Learning Systems. U.S. Patent 10,319,249, 11 June 2019.
5. Cambria, E. Affective computing and sentiment analysis. *IEEE Intell. Syst.* **2016**, *31*, 102–107. [CrossRef]
6. Poria, S.; Cambria, E.; Howard, N.; Huang, G.B.; Hussain, A. Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing* **2016**, *174*, 50–59. [CrossRef]

7. Chaturvedi, I.; Satapathy, R.; Cavallari, S.; Cambria, E. Fuzzy commonsense reasoning for multimodal sentiment analysis. *Pattern Recognit. Lett.* **2019**, *125*. [CrossRef]
8. Charniak, E. *Introduction to Deep Learning*; The MIT Press: Cambridge, MA, USA, 2019.
9. Howard, A.G. Some improvements on deep convolutional neural network based image classification. *arXiv* **2013**, arXiv:1312.5402.
10. Greenspan, H.; Van Ginneken, B.; Summers, R.M. Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Trans. Med Imaging* **2016**, *35*, 1153–1159. [CrossRef]
11. Mahdianpari, M.; Salehi, B.; Rezaee, M.; Mohammadimanesh, F.; Zhang, Y. Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery. *Remote Sens.* **2018**, *10*, 1119. [CrossRef]
12. Baccouche, M.; Mamalet, F.; Wolf, C.; Garcia, C.; Baskurt, A. Sequential deep learning for human action recognition. In *International Workshop on Human Behavior Understanding*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 29–39.
13. Ng, W.; Minasny, B.; Montazerolghaem, M.; Padarian, J.; Ferguson, R.; Bailey, S.; McBratney, A.B. Convolutional neural network for simultaneous prediction of several soil properties using visible/near-infrared, mid-infrared, and their combined spectra. *Geoderma* **2019**, *352*, 251–267. [CrossRef]
14. Savvides, M.; Luu, K.; Zheng, Y.; Zhu, C. Methods and Software for Detecting Objects in Images Using a Multiscale Fast Region-Based Convolutional Neural Network. U.S. Patent 10,354,362, 16 July 2019.
15. Zhao, H.; Liu, F.; Zhang, H.; Liang, Z. Research on a learning rate with energy index in deep learning. *Neural Netw.* **2019**, *110*, 225–231. [CrossRef]
16. Luo, L.; Xiong, Y.; Liu, Y.; Sun, X. Adaptive gradient methods with dynamic bound of learning rate. *arXiv* **2019**, arXiv:1902.09843.
17. Yedida, R.; Saha, S. A novel adaptive learning rate scheduler for deep neural networks. *arXiv* **2019**, arXiv:1902.07399.
18. Cai, S.; Gao, J.; Zhang, M.; Wang, W.; Chen, G.; Ooi, B.C. Effective and efficient dropout for deep convolutional neural networks. *arXiv* **2019**, arXiv:1904.03392.
19. Labach, A.; Salehinejad, H.; Valaee, S. Survey of Dropout Methods for Deep Neural Networks. *arXiv* **2019**, arXiv:1904.13310.
20. Hou, S.; Wang, Z. Weighted channel dropout for regularization of deep convolutional neural network. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019.
21. Maguolo, G.; Nanni, L.; Ghidoni, S. Ensemble of Convolutional Neural Networks Trained with Different Activation Functions. *arXiv* **2019**, arXiv:1905.02473.
22. Dubey, A.K.; Jain, V. Comparative Study of Convolution Neural Network's ReLu and Leaky-ReLu Activation Functions. In *Applications of Computing, Automation and Wireless Systems in Electrical Engineering*; Springer: Singapore, 2019; pp. 873–880.
23. Bawa, V.S.; Kumar, V. Linearized sigmoidal activation: A novel activation function with tractable non-linear characteristics to boost representation capability. *Expert Syst. Appl.* **2019**, *120*, 346–356. [CrossRef]
24. Lohani, H.K.; Dhanalakshmi, S.; Hemalatha, V. Performance Analysis of Extreme Learning Machine Variants with Varying Intermediate Nodes and Different Activation Functions. In *Cognitive Informatics and Soft Computing*; Springer: Singapore, 2019; pp. 613–623.
25. Eckle, K.; Schmidt-Hieber, J. A comparison of deep networks with ReLu activation function and linear spline-type methods. *Neural Netw.* **2019**, *110*, 232–242. [CrossRef]
26. Freire-Obregón, D.; Narducci, F.; Barra, S.; Castrillón-Santana, M. Deep learning for source camera identification on mobile devices. *Pattern Recognit. Lett.* **2019**, *126*, 86–91. [CrossRef]
27. Liu, Y.; Wang, X.; Wang, L.; Liu, D. A modified leaky ReLu scheme (MLRS) for topology optimization with multiple materials. *Appl. Math. Comput.* **2019**, *352*, 188–204. [CrossRef]
28. Zuo, Z.; Li, J.; Wei, B.; Yang, L.; Fei, C.; Naik, N. Adaptive Activation Function Generation Through Fuzzy Inference for Grooming Text Categorisation. In Proceedings of the 2019 IEEE International Conference on Fuzzy Systems, New Orleans, LA, USA, 23–26 June 2019.

29. Tsai, Y.H.; Jheng, Y.J.; Tsaih, R.H. The Cramming, Softening and Integrating Learning Algorithm with Parametric ReLu Activation Function for Binary Input/Output Problems. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–7.
30. Yang, J.; Duan, A.; Li, K.; Yin, Z. Prediction of vehicle casualties in major traffic accidents based on neural network. In *AIP Conference Proceedings*; AIP Publishing: Chongqing, China, **2019**; Volume 2073, p. 020098.
31. Shynk, J.J. Performance surfaces of a single-layer perceptron. *IEEE Trans. Neural Netw.* **1990**, *1*, 268–274. [CrossRef]
32. Tang, J.; Deng, C.; Huang, G.B. Extreme learning machine for multilayer perceptron. *IEEE Trans. Neural Networks Learn. Syst.* **2015**, *27*, 809–821. [CrossRef] [PubMed]
33. Akçay, S.; Kundegorski, M.E.; Devereux, M.; Breckon, T.P. Transfer learning using convolutional neural networks for object classification within x-ray baggage security imagery. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 1057–1061.
34. Lyons, M.J.; Akamatsu, S.; Kamachi, M.; Gyoba, J.; Budynek, J. The Japanese female facial expression (JAFFE) database. In Proceedings of the Third International Conference on Automatic Face and Gesture Recognition, Nara, Japan, 14–16 April 1998; pp. 14–16.
35. Carrier, P.L.; Courville, A.; Goodfellow, I.J.; Mirza, M.; Bengio, Y. *FER-2013 Face Database*; Technical report; Universit de Montral: Montral, QC, Canada, 2013.
36. Abdullah, A.I. Facial Expression Identification System Using fisher linear discriminant analysis and K-Nearest Neighbor Methods. *ZANCO J. Pure Appl. Sci.* **2019**, *31*, 9–13.
37. Alif, M.M.F.; Syafeeza, A.; Marzuki, P.; Alisa, A.N. Fused convolutional neural network for facial expression recognition. In Proceedings of the Symposium on Electrical, Mechatronics and Applied Science 2018, Melaka, Malaysia, 8 November 2018; pp. 73–74.
38. Lopes, A.T.; de Aguiar, E.; De Souza, A.F.; Oliveira-Santos, T. Facial expression recognition with convolutional neural networks: Coping with few data and the training sample order. *Pattern Recognit.* **2017**, *61*, 610–628. [CrossRef]
39. Liu, P.; Han, S.; Meng, Z.; Tong, Y. Facial expression recognition via a boosted deep belief network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, America, 24–27 June 2014; pp. 1805–1812.
40. Shan, C.; Gong, S.; McOwan, P.W. Facial expression recognition based on local binary patterns: A comprehensive study. *Image Vis. Comput.* **2009**, *27*, 803–816. [CrossRef]
41. Selitskaya, N.; Sielicki, S.; Jakaite, L.; Schetinin, V.; Evans, F.; Conrad, M.; Sant, P. Deep Learning for Biometric Face Recognition: Experimental Study on Benchmark Data Sets. In *Deep Biometrics*; Springer: Cham, Switzerland, 2020; pp. 71–97.
42. Yee, S.Y.; Rassem, T.H.; Mohammed, M.F.; Awang, S. Face Recognition Using Laplacian Completed Local Ternary Pattern (LapCLTP). In *Advances in Electronics Engineering*; Springer: Singapore, 2020; pp. 315–327.
43. Nandi, A.; Dutta, P.; Nasir, M. Recognizing Human Emotions from Facial Images by Landmark Triangulation: A Combined Circumcenter-Incenter-Centroid Trio Feature-Based Method. In *Algorithms in Machine Learning Paradigms*; Springer: Singapore, 2020; pp. 147–164.
44. Shin, M.; Kim, M.; Kwon, D.S. Baseline CNN structure analysis for facial expression recognition. In Proceedings of the 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), New York, NY, USA, 26–31 August 2016; pp. 724–729.
45. Alfakih, A.; Yang, S.; Hu, T. Multi-view Cooperative Deep Convolutional Network for Facial Recognition with Small Samples Learning. In *International Symposium on Distributed Computing and Artificial Intelligence*; Springer: Cham, Switzerland, 2019; pp. 207–216.
46. Yar, H.; Jan, T.; Hussain, A.; Din, S.U. Real-Time Facial Emotion Recognition and Gender Classification for Human Robot Interaction Using CNN. Available online: https://www.academia.edu/41996316/Real-Time_Facial_Emotion_Recognition_and_Gender_Classification_for_Human_Robot_Interaction_using_CNN (accessed on 9 March 2020).

47. Agrawal, A.; Mittal, N. Using CNN for facial expression recognition: A study of the effects of kernel size and number of filters on accuracy. *Vis. Comput.* **2020**, *36*, 405–412. [CrossRef]
48. Vishwakarma, V.P.; Dalal, S. A novel non-linear modifier for adaptive illumination normalization for robust face recognition. *Multimed. Tools Appl.* **2020**, 1–27. [CrossRef]