Eurasia Journal of Science and Technology	ISSN: 2663-1024	Volume 4, Issue 2, pp 6-11	APR 2022

A DEEP SIAMESE NETWORK MODEL FOR LARGE-SCALE SIMILAR PICTURE COMPARISON

Jin Lu*

Guangdong Key Laboratory of Big Data Intelligence for Vocational Education, Shenzhen Polytechnic, Shenzhen 518055, Guangdong, China;
Corresponding Author's Email: lujin0808@szpt.edu.cn

Abstract
Comparative picture handling may be a troublesome issue within the field of computer vision, which is broadly utilized in confront acknowledgment, signature acknowledgment, question following, brilliantly restorative and other areas. This paper proposes a profound Siamese network demonstrate for comparative picture comparison, which can be utilized within the assignment of finding the relationship between two comparable things. Based on the conventional twin arrange design, the acknowledgment rate can reach more than 99.75% by altering the parameters and structure and utilizing two or more indistinguishable sub-nets with the same engineering and sharing the same parameters and weights.
Keywords: Siamese Network; Similar Picture Comparison

INTRODUCTION

Computer image processing is not as intuitive as human understanding and processing, the image in the computer can be seen as a matrix, the element in the matrix is a color value, this value is composed of RGB three parameters, the value range of these three parameters is 0 ~ 255 [1]. Of course, RGB is not the only representation of pictures, and other types will not be described in detail [2]. Because the range of 0 ~ 255 is too large, the dimension of the picture narcotization should be reduced [3]. Narcotization turns the image into only black and white, using the OTSU algorithm[4][5]. Black is presented by 1 and white is presented by 0, so a matrix is got which consistent of the numbers 0 and 1. Image contrast is a very important application in the field of computer vision, and the current common solution is to use the hash algorithm[6]. The advantage of using hash algorithm for image comparison is that once the hash database is established, duplicate images can be found quickly. However, for large-scale images, the efficiency of hash algorithm for image comparison will be greatly reduced. To this end, the deep Siamese network is applied to the picture comparison. As in figure 1, handwritten pictures comparing is hard to process.

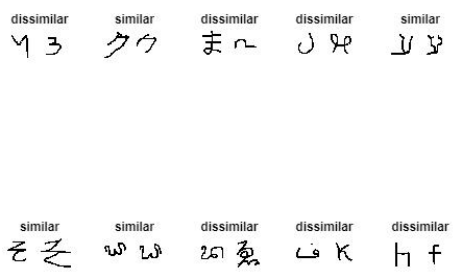

Figure 1 Comparison of similar handwritten pictures

A Siamese network may be a sort of profound learning organize that employments t wo or more indistinguishable sub-networks that have the same engineering and share the same parameters and weights.

Siamese network are ordinarily utilized in errands that in clude finding the relationship between two comparable things [7].

A few common applications for Siamese network incorporate facial acknowledgment, signature confirmation, or rewording distinguishing proof [8]. Siamese network perform well in these errands since their shared weights cruel there are less parameters to memorize amid preparing and they can deliver great comes about with a moderately little sum of preparing information [9]. Siamese network are especially valuable in cases where there are expansive numbers of classes with little numbers of perceptions of each. In such cases, there is not sufficient information to prepare a profound constit utional neural organize to classify images into these classes. Instep, the Siamese network can decide on the off chance that two pictures are within the same lesson [10]. Dey et. al., model an offline writer independent signature verification task with a convolutional Siamese network[11]. Guo et. al., propose dynamic Siamese network, via a fast transformation learning model that enables effective online learning of target appearance variation and background suppression from previous frames [12], which can be formally defined as establishing the correspondence between images of a person taken from different cameras at different times. Chung et. al., present a two stream constitutional neural network where each stream is a Siamese network [13]. Observing that Semantic features learned in an image classification task and Appearance features learned in a similarity matching task complement each other, He et.al., build a twofold Siamese network, named SA-Siam,

for real-time object tracking [14]. Hanif propose improved architecture for two-channel and Siamese networks [15]. Zheng et. al., propose a new deep architecture for person re-identification [16]. Visual tracking addresses the problem of localizing an arbitrary target in video according to the annotated bounding box, Shen et. al., present a novel tracking method by introducing the attention mechanism into the Siamese network to increase its matching discrimination [17]. Li et. al., propose a novel gradient-guided network to exploit the discriminative information in gradients and update the template in the siamese network through feed-forward and backward operations [18]. Zou et. al., propose a high-performance model based on a deep Siamese network (SiamFC-R22) for real-time visual tracking [19].

The Siamese network model has been widely used to solve the large-scale similar picture comparison problem. However, in large-scale similar picture comparison, there are many redundant and irrelevant pictures which cannot be removed. This paper proposes a new deep siamese network model for solving this problem by using the deep convolutional neural network with a novel feature extraction method that incorporates both spatial information and temporal information of images into one unified vector representation.

Solution

The network architecture of is shown in Figure 2. The network consists of two layers, a convolutional layer and a pooling layer. The convolutional layer is used to extract features from the input image, which are then passed through the pooling layer to reduce the dimensionality of the feature space. In this model, we use VGG16 as our feature extractor and ResNet18 as our pooling module [20].

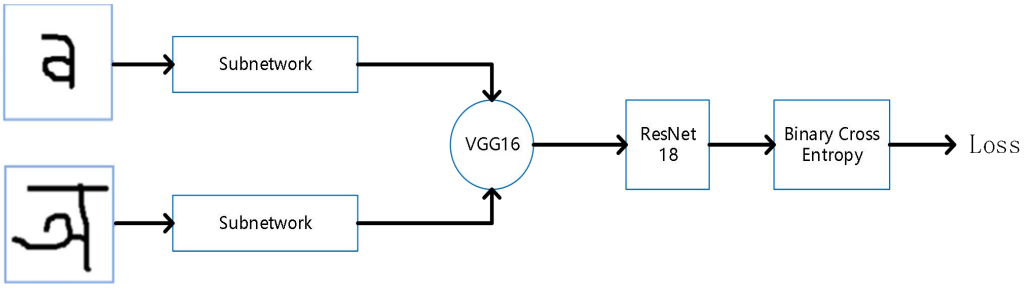

Figure 2 Deep neural network architecture

In this paper, two identical subnets are defined as a series of fully connected layers with ReLU layers by creating a network that accepts 28 × 28 × 1 images and outputs two feature vectors for simplified feature representation. The network reduces the dimension of the input image to 2, a value that is easier to draw and visualize than the initial dimension of 784. For the first two fully connected layers, specify an output size of 1024 and use the he weight initializer. For the final fully connected layer, specify an output size of 2 and use the he weight initializer. Here, in order to implement a custom training loop to train the network and enable automatic differentiation, the layer map needs to be converted to a dlnetwork object. In this paper, the model gradient function accepts the Siamese data network object data network, for small batches of input data X1 and X2, and label to label, the function returns the loss relative to the gradient of the learnable parameters in the network and the loss of contrast between the reduced dimensionality features of paired images.

The model gradient function used in this paper uses a mini-batch of twin data network objects dlnet and input data dlX1 and dlX2 with a pair of labels. The function returns the loss value and the gradient of the loss with respect to the learnable parameters of the network. The goal of the twin network is to output a feature vector for each image such that the feature vectors of similar images are similar and the feature vectors of different images are significantly different. In this way, the network can distinguish between the two inputs by finding the output of the last fully connected layer, i.e., the contrast loss between feature 1 and feature 1 from the feature vectors of paired image 1 and paired image 2, respectively. The contrast loss for a pair is given by [21]:

loss=12yd2+12(1−y)max(margin−d,0)2

Where y is the value of the pairing label (y = 1 for similar images and y = 0 for different images) and d is the Euclidean distance between the two eigenvectors F1 and F2: d = F1 F2 2. The margin parameter is used as a constraint: if two images in a pair are not similar, then their distance should be at least the margin, otherwise it results in a loss. In the case of similar images, the first term may be non-zero and minimized by reducing the distance between the image features F1 and F2. In the case of dissimilar images, the second term may be non-zero and minimized by increasing the distance between image features, at least to the edge. The smaller the value of the margin, the smaller the restrictions on the different pairs before a loss occurs.

Experimental process

This article uses MATLAB R2021's own data set. By loading the training data consisting of handwritten digit images, the function digitTrain4DArrayData loads the digit images and their labels by loading a 28 × 28 × 1 × 5000 array containing 5000 single-channel images, each of which has a size of 28 × 28. Each pixel has a value between 0 and 1. YTrain is a classification vector containing the labels of each observation, which are numbers from 0 to 9, corresponding to the values of the written numbers to display a randomly selected image.

In order to train the network, the data is divided into

similar or dissimilar image pairs. As used herein, similar images are defined as having the same label, while different images have different labels. The function getSiameseBatch creates a random pair of similar or dissimilar images, Paired Image 1 and Paired Image 2. The function also returns the label pair label, which identifies whether the image pair is similar or dissimilar to each other. Similar image pairs have pairLabel = 1 and dissimilar image pairs have pairLabel = 0.We create a small representative set of five pairs of images, creating a new batch of 180 pairs of images for each iteration of the training loop. This ensures that the network is trained on a large number of random image pairs with approximately equal proportions of similar and dissimilar pairs.

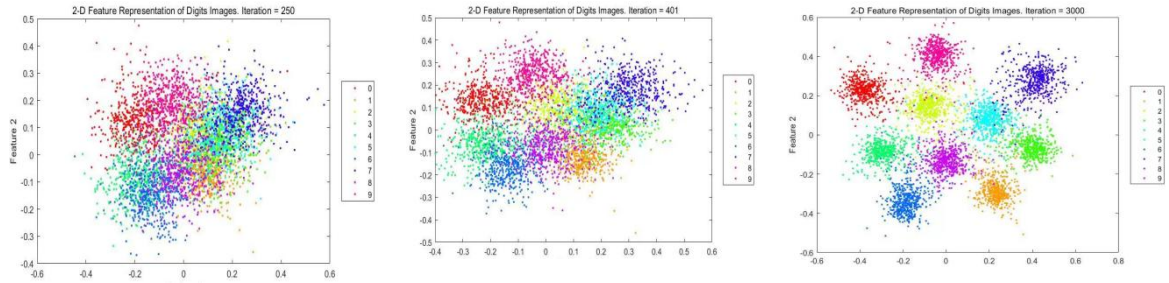

Figure 3 Training process

The experiments of 250, 401 and 3000 times of training were selected respectively. Through training, the network has now learned to represent each image as a two-dimensional vector. As can be seen from the reduced feature map of the test data (Figure 3), in this two-dimensional representation, images of similar numbers are clustered together.

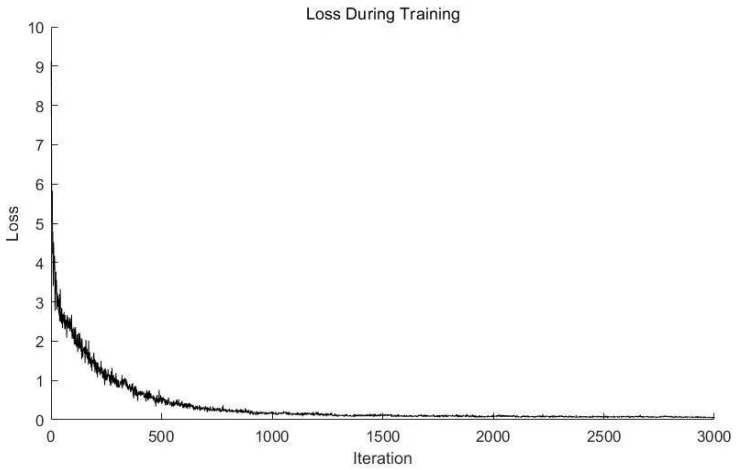

Figure 4 Change of loss function in simulation proces

It can be seen from Figure 4 that with the increase of the number of training times, the loss function converges at about 2000 training times and is optimized at about 3000 training times.

Experimental results

All the experiments were carried out on the workstation. The equipment parameters were intel I9-10900k, 64GB, 2 * RTX3090, 1 T pcie3.0 SSD, and the experimental platform was Matlab 2017 B. During the experiment, the CIFAR-10 data set was used to train the constructed convolutional neural network. The model is trained using a custom training loop. The training data is cycled and the network parameters are updated in each iteration. The Get Image Batch function, defined in the Batch section of, extracts a batch of image pairs and labels. Converts the image data to a array object of the underlying type single. For GPU training, the image data is converted to a GPU array object, the model gradient is evaluated using medieval and the model gradient function, and the network parameters are updated using the update function.

Table1: Experimental results

Number of convolution layers	Number of convolution kernels	Number of pooling layers	Recognition rate （%）
1	8	1	94.84
1	16	1	97.64
1	32	1	96.6
2	8	2	92.68
2	16	2	98.88
2	32	2	98.32
3	8	3	89.24
3	16	3	99.72
3	32	3	99

It can be seen from the experimental results (Table 1 and Figure 4) that the recognition rate changes during the simultaneous adjustment of the number of convolution layers, the number of convolution kernels, and the number of pooling layers. Experiments show that when the number of convolution layers is 3, the number of convolution kernels is 16, and the number of pooling layers is 3, the recognition rate is the highest, which can reach 99. 72%

.

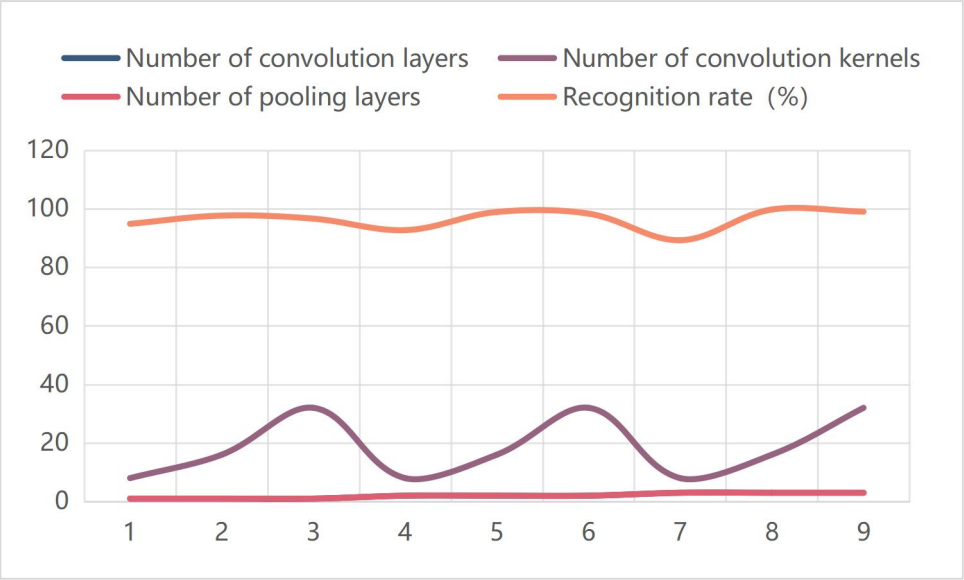

Figure 5 Change of loss function in simulation proces

From the experimental results (Figure 5 and Figure 6), it can be seen that the recognition rate can be partially improved by blindly increasing the number of convolution layers and cycles, but the recognition rate is close to 100% after 250 cycles.

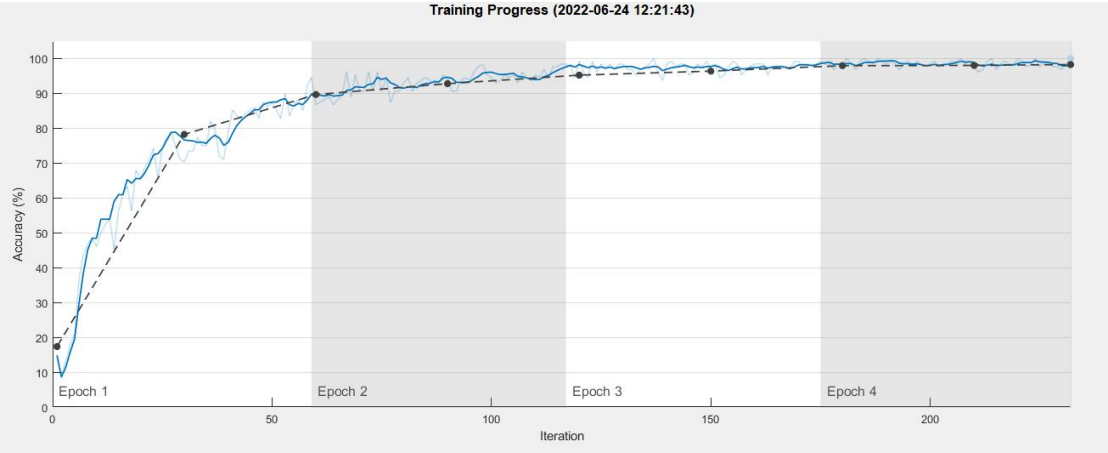

Figure 6 Recognition rate trend

CONCLUSION

In this paper, an improved Siamese network model is proposed, which can be applied to large-scale similar graph classification. The whole model has two inputs, the two inputs are fed into two neural networks, and the two neural networks respectively map the inputs to a new space to form a representation of the inputs in the new space. Experiments show that the recognition rate can reach 99.72%, which is in the leading level among similar models.

FUNDING

This work is partially supported by Shenzhen Education Science "14TH FIVE-YEAR PLAN"2021 Subject: Research on online learning emotion analysis and intelligent tutoring based on collaborative perception of multi-modal education data(ybzz21015), Key technology research and innovative application demonstration of intelligent education(2019KZDZX1048), Guangdong Key Laboratory of Big Data Intelligence for Vocational Education(2019GKSYS001), Shenzhen Vocational Education Research Center Jointly Established by the Ministry and the Province(6022240004Q).

REFERENCE

[1] Syed Muhammad Arsalan Bashir; Yi Wang; Mahrukh Khan; Yilong Niu; "A Comprehensive Review of Deep Learning-based Single Image Super-resolution", PEERJ. COMPUTER SCIENCE, 2021.

[2] Jiawei Zhang; Chen Li; Md Mamunur Rahaman; Yudong Yao; Pingli Ma; Jinghua Zhang; Xin Zhao; Tao Jiang; Marcin Grzegorzek; "A Comprehensive Review of Image Analysis Methods for Microorganism Counting: from Classical Image Processing to Deep Learning Approaches", ARTIFICIAL INTELLIGENCE REVIEW, 2021.

[3] Dhanamjayulu Chittathuru; N. NizhalU.; Praveen Kumar Reddy Maddikunta; Thippa Reddy Gadekallu; Celestine Iwendi; Chuliang Wei; Qin Xin; "Identification of Malnutrition and Prediction of BMI from Facial Images Using Real-time Image Processing and Machine Learning", IET IMAGE PROCESS., 2022.

[4] Bhimavarapu Usharani; "Hypertensive Retinopathy Classification Using Improved Clustering Algorithm and The Improved Convolution Neural Network", ADVANCES IN SYSTEMS ANALYSIS, SOFTWARE ENGINEERING, AND HIGH PERFORMANCE COMPUTING, 2022.

[5] Shinichiro Mori; "Deep Architecture Neural Network-based Real-time Image Processing For Image-guided Radiotherapy", PHYSICA MEDICA : PM : AN INTERNATIONAL JOURNAL DEVOTED TO THE APPLICATIONS OF PHYSICS TO MEDICINE AND BIOLOGY : OFFICIAL JOURNAL OF THE ITALIAN ASSOCIATION OF BIOMEDICAL PHYSICS (AIFB), 2017.

[6] S Shirly; K Ramesh; "Review On 2D And 3D MRI Image Segmentation Techniques", CURRENT MEDICAL IMAGING REVIEWS, 2017.

[7] Stephen Lynch; "Image Processing with Python", 2018.

[8] Thierry Bouwmans; Sajid Javed; Hongyang Zhang; Zhouchen Lin; Ricardo Otazo; "On The Applications of Robust PCA in Image and Video Processing", PROCEEDINGS OF THE IEEE, 2018.

[9] Xi-Wei Yao; Hengyan Wang; Zeyang Liao; Ming-Cheng Chen; Jian Pan; Jun Li; Kechao Zhang; Xingcheng Lin; Zhehui Wang; Zhihuang Luo; Wenqiang Zheng; Jianzhong Li; Meisheng Zhao; Xinhua Peng; Dieter Suter; "Quantum Image Processing And Its Application To Edge Detection: Theory And Experiment", ARXIV, 2018.

[10] Gene Cheung; Enrico Magli; Yuichi Tanaka; Michael Ng; "Graph Spectral Image Processing", ARXIV, 2018.

[11] Sounak Dey; Anjan Dutta; J. Ignacio Toledo; Suman K. Ghosh; Josep Llados; Umapada Pal; "SigNet: Convolutional Siamese Network For Writer Independent Offline Signature Verification", ARXIV, 2017.

[12] Qing Guo; Wei Feng; Ce Zhou; Rui Huang; Liang Wan; Song Wang; "Learning Dynamic Siamese Network For Visual Object Tracking", ICCV, 2017.

[13] Dahjung Chung; Khalid Tahboub; Edward J. Delp; "A Two Stream Siamese Convolutional

Neural Network For Person Re-Identification", ICCV, 2017.
[14] Anfeng He; Chong Luo; Xinmei Tian; Wenjun Zeng; "A Twofold Siamese Network For Real-Time Object Tracking", CVPR, 2018.
[15] Shehzad Muhammad Hanif; "Patch Match Networks: Improved Two-channel and Siamese Networks for Image Patch Matching", PATTERN RECOGNIT. LETT., 2019.
[16] Meng Zheng; Srikrishna Karanam; Ziyan Wu; Richard J. Radke; "Re-Identification With Consistent Attentive Siamese Networks", CVPR, 2019.
[17] Jianbing Shen; Xin Tang; Xingping Dong; Ling Shao; "Visual Object Tracking By Hierarchical Attention Siamese Network", IEEE TRANSACTIONS ON CYBERNETICS, 2019.
[18] Peixia Li; Boyu Chen; Wanli Ouyang; Dong Wang; Xiaoyun Yang; Huchuan Lu; "GradNet: Gradient-Guided Network For Visual Object Tracking", ICCV, 2019.
[19] Qijie Zou; Yue Zhang; Shihui Liu; Jing Yu; "A Real-Time Object Tracking Model Based on Deeper Siamese Network", 2020 3RD INTERNATIONAL CONFERENCE ON UNMANNED SYSTEMS (ICUS), 2020.
[20] Guo Q, Wei F, Zhou C, et al. Learning Dynamic Siamese Network for Visual Object Tracking[C]// 2017 IEEE International Conference on Computer Vision (ICCV). IEEE Computer Society, 2017.
[21] Zheng H, Gong M, Liu T, et al. HFA-Net: High frequency attention siamese network for building change detection in VHR remote sensing images[J]. Pattern Recognition, 2022, 129.