FACIAL RECOGNITION AND CLASSIFICATION OF TERRACOTTA WARRIORS IN THE MAUSOLEUM OF THE FIRST EMPEROR USING DEEP LEARNING

YAN SHENG
School of Geomatics and Urban Spatial Informatics, Beijing University of Civil Engineering and Architecture, Beijing 100044, China,1641907305@qq.com

KEY WORDS: Terracotta,Army,Deep Learning,Facial Recognition,Facial Classification.

ABSTRACT:

The facial features of the Terracotta Warriors unearthed from the Mausoleum of the First Emperor of Qin are authentic depictions of the appearance of soldiers from the same period. Recognizing facial features to classify the Terracotta Warriors is one of the crucial aspects of archaeological research. Due to limitations in the collection of facial samples from the Terracotta Warriors, an enhanced SqueezeNet model is proposed for deep learning facial recognition. The FaceNet backbone feature extraction network has been improved by replacing the initial 7×7 convolution kernel with three 3×3 convolution kernels. The model's feature extraction layer is composed of alternating convolution layers, pooling layers, Fire modules, and pooling layers, with the introduction of an exponential function to smooth the shape of the loss function. Finally, facial classification of 295 Terracotta Warriors is accomplished using Agglomerative Clustering. The model demonstrates a facial recognition accuracy of 95.6%, showing a respective improvement of 4.1% and 2.8% compared to the classical SqueezeNet and Inception_ResNetV1 models. This approach better meets the requirements for facial recognition and classification of Terracotta Warriors, providing intelligent and efficient technical support for technological archaeology.

1. INTRODUCTION

The Terracotta Army of the Mausoleum of the First Emperor of Qin is listed on the UNESCO World Heritage List and acclaimed as the 'Eighth Wonder of the World.' It provides crucial physical evidence for the study of the history of the Qin Dynasty. The intricately carved facial features of the Terracotta Warriors are vivid and lifelike, showcasing an artistic characteristic of 'a thousand faces,' known for its realism. The research on their classification, feature recognition, and the similarities and differences with real people has garnered widespread attention from archaeologists and the public(Fei Yicheng,2017). In the archaeological report of Pit 1 of the Terracotta Army, the facial types are categorized into eight types such as 'Tian,' 'You,' 'Guo,' etc (Zhao Zhen,2015). Additionally, studies have utilized three-dimensional laser scanning to reconstruct high-precision models of the Terracotta Warriors' heads and faces, extracting features for comparative analysis with the facial characteristics of modern ethnicities (Hu Yungang,2022). In recent years, with the widespread application of deep learning in facial recognition, its application to the recognition and classification of Terracotta Warrior faces can intelligently yield effective results. This has significant implications for introducing artificial intelligence into archaeology.

At present, with the development of the DeepFace algorithm proposed by the Facebook team(Taigman Y,,2014)and the FaceNet algorithm introduced by the Google team(Schroff F, 2015), deep learning has successfully been applied in various fields such as access control systems(Yan Zifeng,2022), emotion recognition(Abiram R N,2021), gender recognition(Mansanet J,2016), and age estimation(Zhang Liangliang,2020). Key steps in deep learning algorithms for facial recognition and classification include facial detection and key point localization alignment, facial feature extraction, and facial clustering. Algorithms such as Cascade CNN(Qin H,2016), MTCNN(Zhang K,2016), and CMS-RCNN(Zhu C,2017) are mature methods for facial detection and key point localization alignment. Traditional feature extraction methods include PCA(Turk M,1991), LDA(Belhumeur P N,1997), LBP (Ahonen T,2006), but these methods require manually designed features, and the results are often influenced by human expertise. Deep Convolutional Neural Networks (CNNs) are the main models used for facial feature extraction, with commonly used models including DenseNet(Huang G,2017), VGG(Yan Z,2015), Inception-ResNetV1(Szegedy C,2015). However, these models contain a large number of parameters, and inadequate sample sizes can lead to poor model fitting and unsatisfactory results. In addition, the choice of loss function during model training directly impacts the results. For example, the Triplet Loss loss function used in the FaceNet algorithm aims to ensure that samples of the same category are closer together in the feature space, while samples of different categories are farther apart, focusing on difficult-to-classify samples. However, this approach may cause the model to overlook easily classifiable samples, thus reducing the model's generalization ability. For facial feature clustering, commonly used algorithms include K-Means(Hartigan J A,1979), DBSCAN(Ester M,1966), Agglomerative Clustering(Gowda K C,1978). Among them, Agglomerative Clustering is a merging-based hierarchical clustering algorithm, mainly used for handling data with unclear density differences, and it performs better for clustering small-scale data.

According to the excavation report of Pit 1 of the Terracotta Army in the Mausoleum of the First Emperor of Qin, over 1587 Terracotta Warriors have been unearthed, and the restoration and repositioning work for 714 of them have been completed. Considering various factors, this study utilized a Nikon D810 digital single-lens reflex (DSLR) camera to capture frontal and side view photographs of the faces of 295 Terracotta Warriors in the exhibition area of Pit 1, totaling 1209 images. From the perspective of deep learning, the facial data of the Terracotta Warriors can be considered as a small-sample dataset. The existing facial recognition models designed for large samples may not be suitable, necessitating the selection of a lightweight convolutional neural network suitable for small samples. Therefore, an improved model based on the SqueezeNet lightweight convolutional neural network model is proposed. This model replaces the original convolutional kernels with multiple small convolutional kernels and constructs a new feature extraction layer by alternately using convolution layers, pooling layers, Fire modules, and pooling layers. This architecture is designed to train facial sample data of the Terracotta Warriors. Additionally, improvements have been made to the triplet loss function by introducing an exponential function to smooth the shape of the loss function, addressing the issue of imbalance between challenging and easy samples.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-2024-205-2024 | © Author(s) 2024. CC BY 4.0 License.

2. DATA PREPROCESSING

Firstly, the MTCNN algorithm is applied to perform facial detection on the original images, outputting rectangular regions bounding the faces. Next, facial key points are matched to locate key features including eyes, nose tip, mouth, etc. Subsequently, facial images of the Terracotta Warriors are aligned to obtain uniformly sized frontal or side-view facial images, as shown in Figure 1.

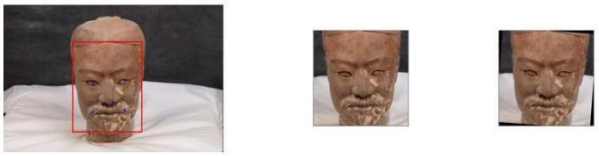

Figure 1. Terracotta warrior facial detection and key point localization alignment

After cropping and aligning the Terracotta Warrior images, they were annotated following the format of the Labeled Faces in the Wild (LFW) dataset and stored in the corresponding labeled folders, organizing them into a Terracotta Warrior Facial Dataset. Due to the limited number of samples, data augmentation techniques such as random rotation and random flipping were employed. In the end, the Terracotta Warrior Facial Dataset comprises a total of 12,439 photos from 295 statues, as illustrated in Figure 2.

Figure 2. Partial display of terracotta warrior facial dataset

3.FEATURE EXTRACTION AND CLUSTERING OF TERRACOTTA WARRIOR FACIAL FEATURES BASED ON THE IMPROVED FACENET SYSTEM

FaceNet is a deep learning framework for face recognition proposed by the Google team, utilizing the CNN+Triplet loss[3] approach. The structure of this framework is illustrated in Figure 3. The main feature extraction network of FaceNet employs the complex deep convolutional neural network model Inception_ResNetV1. This model is intricate, with parameters seven times that of the classical SqueezeNet model, resulting in a size of 135MB. It requires a substantial amount of training samples to ensure model fitting. Given the limited number of Terracotta Warrior samples compared to regular human facial samples, optimizations were made to enhance the algorithm's efficiency. The optimizations include: (1) Adoption of an improved lightweight SqueezeNet network model to replace the Inception_ResNetV1 network model as the feature extraction model for Terracotta Warriors; (2) Introduction of transfer learning concepts and utilization of L2 regularization and learning rate schedulers to accelerate model training, thereby enhancing model generalization and training stability; (3) Optimization of the Triplet Loss function to make it smoother, addressing the issue of imbalance between challenging and easy samples.

Figure 3 FaceNet framework architecture

3.1 SqueezeNet Convolutional Neural Network Model

SqueezeNet is a lightweight neural network model that, compared to the traditional AlexNet model, has only 1/50th of the parameters, resulting in a model size of 12.9 MB, while achieving similar network recognition performance(Alhichri H, 2019). The core building block of this model is the Fire Module, consisting of two parts: Squeeze and Expand, as illustrated in Figure 4. The Squeeze part utilizes a 1×1 convolutional kernel to reduce the dimensionality of the input feature map, thus reducing the number of parameters in the network. Meanwhile, the Expand part expands the feature map output from Squeeze by parallelly using 1×1 and 3×3 convolutional kernels, increasing the network's width(Huo Aiqing,2020). The overall structure of the SqueezeNet model, composed of 8 Fire modules along with standard convolutional layers, pooling layers, etc., is shown in Figure 5. The model parameters are listed in Table 1. Despite these advantages, the first convolutional layer in the SqueezeNet model, which uses a 7×7 convolutional kernel, is still relatively large, resulting in a complex network structure that may not be suitable for small-sample data.

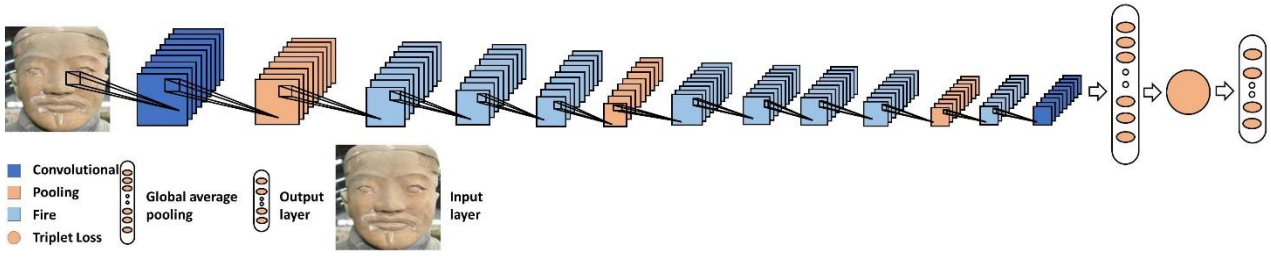

Figure 5 SqueezeNet structure diagram

Table 1 Classical SqueezeNet model parameters

Layer Name	Channel Count,	Convolutional Kernel Size	Convolution kernels	Pooling mode/nucleus size	Activation Function
Conv1	96	7×7	96	Maximum pooling /3×3/2	ReLU
Fire1	128	（3×3）and（1×1）	16 and 64		ReLU

Fire2	128	（3×3）and（1×1）	16 and 64		ReLU
Fire3	256	（3×3）and（1×1）	32 and 128	Maximum pooling /3×3/2	ReLU
Fire4	256	（3×3）and（1×1）	32 and 128		ReLU
Fire5	384	（3×3）and（1×1）	48 and 192		ReLU
Fire6	384	（3×3）and（1×1）	48 and 192		ReLU
Fire7	512	（3×3）and（1×1）	64 and 256	Maximum pooling /3×3/2	ReLU
Fire8	512	（3×3）and（1×1）	64 and 256		ReLU
Conv2	1000	（1×1）	1000		ReLU
Global-avgpool	1000			Global average pooling	

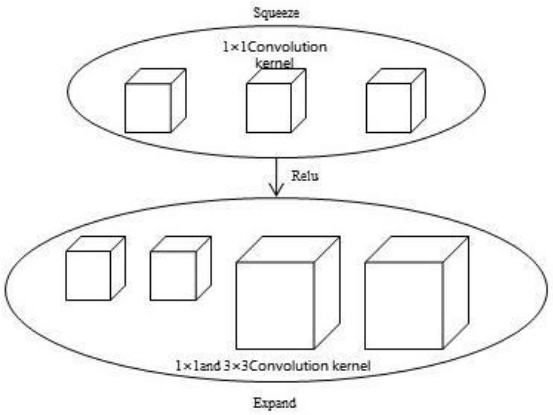

Figure 4 Fire Module structure

3.2 Improved SqueezeNet Convolutional Neural Network Model

In order to reduce the model parameters, increase the model's non-linear capability, and better utilize local features, multiple smaller convolutional kernels were employed instead of a large-sized convolutional kernel. Taking inspiration from the VGG16 model, which replaces a 7×7 convolutional kernel with three 3×3 convolutional kernels to achieve the same receptive field while increasing the network depth and reducing the number of parameters(Tammina S,2019), a similar approach was applied to the classic SqueezeNet model. In this improved model, three 3×3 convolutional kernels replaced one 7×7 convolutional kernel. The advantages include: (1) Parameter optimization - by replacing the first layer's 7×7 convolutional kernel with three 3×3 convolutional kernels, the model's parameter count is significantly reduced. If the size of the previous layer's feature map channels and the current layer's convolutional kernel is denoted as C, the parameter count for one 7×7 convolutional kernel is 7×7×C×C=49C2, while the parameter count for three 3×3 convolutional kernels is 3×(3×3×C×C)=27C2. This results in a reduction of parameters by approximately 1.8 times. (2) Introduction of more non-linearity - each layer of the 3×3 convolutional kernel includes an activation function, whereas the 7×7 convolutional kernel layer has only one activation function. This implies that the stacking of three 3×3 convolutional kernels introduces more non-linear transformations compared to a single 7×7 convolutional kernel, making the network more expressive and better at capturing complex relationships in the data. (3) Increased network depth - multiple 3×3 convolutional kernels increase the depth of the network, helping improve the model's performance in recognizing complex features. (4) Maintaining the receptive field - although the receptive field of each 3×3 convolutional kernel is relatively small, through multiple layers of stacking, a larger receptive field is captured, allowing the identification of larger-sized features.

To address the issue of facial texture feature loss when using small convolutional kernels, an approach was taken to increase the number of pooling layers while appropriately reducing the usage of Fire modules. Specifically, a new SqueezeNet convolutional neural network model was constructed by alternately using convolutional layers, pooling layers, three Fire modules, and pooling layers. The advantages of this approach include: (1) Appropriately increasing the number of pooling layers, which helps the network better capture fine features and contextual information. (2) Appropriately reducing the number of Fire module layers, which can effectively lower the parameter count and improve the computational efficiency of the network. The structure of the improved SqueezeNet model is shown in Figure 6, and its parameters are listed in Table 2.

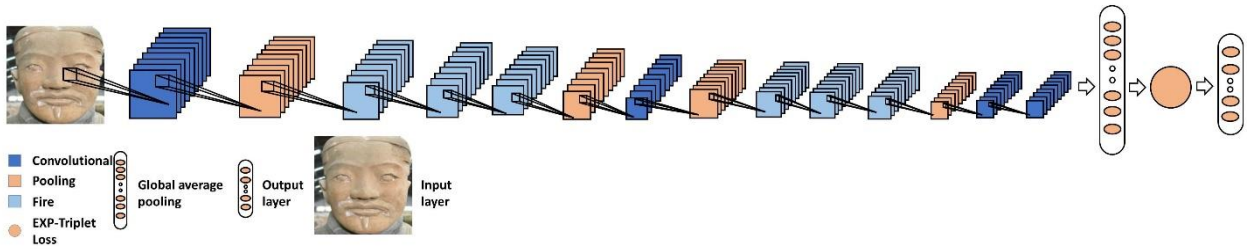

Figure 6 Structure diagram of the improved SqueezeNet model in this article

Table 2 Parameter of the SqueezeNet model in this article

Layer Name	Channel Count,	Convolutional Kernel Size	Convolution kernels	Pooling mode/nucleus size	Activation Function
Conv1	96	3×3	96	Maximum pooling/3×3/2	ReLU
Fire1	128	（3×3）and（1×1）	16 and 64		ReLU
Fire2	128	（3×3）and（1×1）	16 and 64		ReLU

Fire3	256	（3×3）and（1×1）	32 and 128	Maximum pooling/3×3/2	ReLU
Conv2	192	3×3	192	Maximum pooling/3×3/2	ReLU
Fire4	384	（3×3）and（1×1）	48 and 192		ReLU
Fire5	384	（3×3）and（1×1）	48 and 192		ReLU
Fire6	512	（3×3）and（1×1）	64 and 256	Maximum pooling/3×3/2	ReLU
Conv3	384	3×3	384		ReLU
Conv4	1000	1×1	1000		ReLU
Global-avgpool	1000			Global average pooling	

3.3 Optimization of Deep Learning Model Training

In the process of training deep learning models, various optimization strategies are employed, including loading pretrained weights, freezing and unfreezing layers, weight decay, and learning rate schedulers. Among them, loading pretrained weights is a commonly used optimization technique. Pretrained weights are typically obtained through training on an extensive dataset. These weights are then loaded as initial weights and fine-tuned on a specific task's dataset to meet the requirements of that particular task.Freezing and unfreezing layers are primarily applied in the process of transfer learning and fine-tuning models. Weight decay is a regularization technique, often implemented by setting the weight_decay parameter in the optimizer, commonly interpreted as L2 regularization. The main purpose of weight decay is to penalize the complexity of the model by imposing penalties on large weights, preventing the model from becoming overly complex and thereby reducing the risk of overfitting.Learning rate is a crucial hyperparameter that controls the pace of updating model parameters. Dynamic adjustment of the learning rate is an effective optimization strategy that allows for timely adjustments based on the model's training progress. This dynamic adjustment helps enhance the training performance and convergence speed of the model.

For the issue of a relatively small-scale Terracotta Warriors dataset, a more lightweight convolutional neural network (CNN) is adopted. Simultaneously, the training process incorporates the concept of transfer learning to expedite model training and enhance its performance. L2 regularization and a learning rate scheduler are employed to improve the model's generalization ability and training stability. The specific implementation details are as follows:

(1)Loading Pretrained Weights Training: Utilizing a well-trained face model on the large-scale CASIA-WebFace dataset, the pretrained weights are loaded into the backbone network. This approach allows the Terracotta Warriors model to initiate training from these pretrained weights, accelerating the convergence of the model. The incorporation of pretrained weights provides a valuable starting point for the model, leveraging the knowledge learned from the diverse and extensive CASIA-WebFace dataset to boost the efficiency of training on the Terracotta Warriors dataset.

(2)Freezing and Unfreezing Training: The training process is divided into freezing and unfreezing stages. During these stages, the weights of the pretrained model's backbone network are respectively set as non-trainable and trainable. When the weights are non-trainable, only the fully connected layers of the model are trained on the Terracotta Warriors facial data. This preserves the feature extraction capabilities of the face model and accelerates the training speed since only the final layers are updated. When the weights are trainable, the facial data of the Terracotta Warriors is used to train the entire model based on the pretrained face model weights. This fine-tuning process improves the performance of the recognition task by adjusting the pretrained face model to better suit the characteristics of the Terracotta Warriors dataset.

(3)Weight Decay and Learning Rate Scheduler: The Adam optimizer is employed with the use of the weight_decay parameter to introduce L2 regularization. This regularization technique is crucial for constraining the complexity of the model and mitigating the risk of overfitting. Simultaneously, a learning rate scheduler is implemented. After each training epoch, the learning rate is multiplied by 0.94, gradually decreasing the learning rate. This dynamic adjustment of the learning rate enhances training stability and convergence speed, ensuring effective model optimization over the course of training. The combination of weight decay and the learning rate scheduler contributes to improved generalization ability and stability during the training of the Terracotta Warriors model.

3.4 Optimization of Triplet Loss Function

The Triplet Loss function is a commonly used loss function in tasks such as face recognition and image retrieval. However, the Triplet Loss function tends to focus excessively on samples that are difficult to classify, causing the model to overlook those that are easy to classify, thereby affecting the model's generalization ability. Specifically, when there is an imbalance between difficult and easy samples, the training speed of the model may decrease, ultimately leading to the model's failure to converge.

In order to address this issue, an optimization has been applied to the Triplet Loss function by introducing an exponential function to smooth the shape of the loss function. The relative order of samples is adjusted using the exponential function instead of sorting them based on distance, making the loss function smoother and partially alleviating the problem of gradient explosion. Gradient explosion refers to a situation in which gradient values become extremely large during the backpropagation process, leading to significant updates in network weights and making the network unstable. The optimized formula is as follows (1):

$$L(A, P, N) = \log(1 + \exp(\text{beta} * (d(A, P) - d(A, N) + \text{alpha}) \quad (1)$$

- A represents the feature representation of the anchor sample
- P represents the feature representation of the positive sample belonging to the same category as the anchor
- N represents the feature representation of the negative sample not belonging to the same category as the anchor
- d(x, y) represents the distance between sample *x* and sample *y*
- beta is a hyperparameter used to control the slope of the exponential function, thereby adjusting the smoothness of the loss
- alpha is a hyperparameter used to control the difference between the distance from the anchor sample to the positive sample and the distance from the anchor sample to the negative sample

3.5 Cluster Algorithm Selection

While feature extraction plays a decisive role in clustering results, the choice of an appropriate clustering algorithm is equally

important. Agglomerative hierarchical clustering is a hierarchical clustering algorithm that groups samples in a dataset into different clusters. The algorithm adopts a bottom-up strategy, where each sample is initially treated as an independent cluster and then gradually merges the most similar clusters until all samples belong to the same cluster or reach a preset stopping condition. This process utilizes distance or similarity measures to assess the similarity between samples or clusters. Common distance metrics include Euclidean distance, Manhattan distance, and cosine similarity, among others.In comparison to the K-Means clustering algorithm, agglomerative hierarchical clustering does not require the pre-specification of the number of clusters, providing greater flexibility. Compared to the DBSCAN clustering algorithm, it exhibits superior performance in handling data with less distinct density differences.Therefore, agglomerative hierarchical clustering, utilizing Euclidean distance as the distance metric, has been selected as the clustering algorithm for the facial features of the terracotta warrior heads.

3.6 Results Verification Method

The feature extraction accuracy is verified from two perspectives: model accuracy and Euclidean distance. Additionally, the credibility of clustering results is validated using the K-means clustering method in SPSS software.

The recognition accuracy of the Terracotta Warrior test set is the most commonly used metric for evaluating the model's performance. A higher accuracy indicates a better-trained model. The size of the trained model indirectly reflects the number of network parameters, with a smaller model suggesting a more concise network structure and fewer parameters. The formula for calculating model accuracy is as follows (2):

$$accuracy = \frac{TP+TN}{TP+FN+TN+FP} \quad (2)$$

The comparison of Euclidean distance involves extracting features from different photos of the same Terracotta Warrior using different models and calculating the Euclidean distance between the features. A smaller Euclidean distance between different photos of the same Terracotta Warrior indicates a better model.

4. DATA PROCESSING AND ANALYSIS

4.1 Dataset Partitioning

A random selection of 236 Terracotta Warrior data out of 295 statues is used for dataset partitioning according to the typical machine learning split ratio of 7:3, creating a training set and a validation set. The remaining 59 Terracotta Warrior photos are designated as the test set for conducting Terracotta Warrior clustering experiments. Detailed data distribution is shown in Table 3.

Table 3 Data information table

Dataset type	Terracotta warriors and horses	Total number of photos (the number of photos that are not enhanced）
Training set	166	8800（800）
Validation set	70	3850（350）
Test set	59	59

4.2 Training Environment

The primary hardware and software configuration for the training experiment of the Terracotta Warrior feature extraction network model is detailed in Table 4.

Table 4 Configuration information table

Number	Name	Disposition
1	CPU	Intel(R) Core(TM) i7-11700
2	GPU	NVIDIA T600
3	Operating system	Windows 11
4	Programming language	Python
5	Deep learning framework	PyTorch

4.3 Clustering Results

Facial features are extracted using the improved SqueezeNet, and the Agglomerative clustering algorithm is applied to cluster the existing 295 Terracotta Warriors. The clustering results are presented in Table 5. Additionally, a separate clustering is performed on the test set of 59 Terracotta Warriors, and the results are illustrated in Figure 7.

Table 5 Clustering results of terracotta warrior facial features

Terracotta Warriors Clustering Results		Number of clusters
Cluster1	002.jpg, 008.jpg, 010.jpg, 023.jpg, 025.jpg, 070.jpg, 071.jpg, 073.jpg, 081.jpg, 082.jpg, 083.jpg, 085.jpg, 086.jpg, 087.jpg, 090.jpg, 091.jpg, 092.jpg, 100.jpg, 114.jpg, 123.jpg, 125.jpg, 126.jpg, 130.jpg, 135.jpg, 142.jpg, 152.jpg, 153.jpg, 155.jpg, 161.jpg, 169.jpg, 173.jpg, 174.jpg, 176.jpg, 177.jpg, 183.jpg, 186.jpg, 191.jpg, 192.jpg, 193.jpg, 198.jpg, 199.jpg, 201.jpg, 203.jpg, 204.jpg, 206.jpg, 207.jpg, 208.jpg, 211.jpg, 214.jpg, 221.jpg, 231.jpg, 233.jpg	52
Cluster2	009.jpg, 013.jpg, 027.jpg, 067.jpg, 084.jpg, 088.jpg, 089.jpg, 094.jpg, 096.jpg, 097.jpg, 098.jpg, 107.jpg, 113.jpg, 117.jpg, 124.jpg, 128.jpg, 129.jpg, 131.jpg, 137.jpg, 139.jpg, 141.jpg, 143.jpg, 144.jpg, 145.jpg, 146.jpg, 149.jpg, 156.jpg, 157.jpg, 159.jpg, 160.jpg, 167.jpg, 179.jpg, 180.jpg, 189.jpg, 195.jpg, 196.jpg, 200.jpg, 202.jpg, 212.jpg, 217.jpg, 218.jpg, 222.jpg, 225.jpg, 232.jpg, 281.jpg, 284.jpg	46
Cluster3	018.jpg, 024.jpg, 029.jpg, 032.jpg, 034.jpg, 038.jpg, 042.jpg, 045.jpg, 050.jpg, 053.jpg, 057.jpg, 058.jpg, 062.jpg, 064.jpg, 065.jpg, 069.jpg, 076.jpg, 187.jpg, 237.jpg, 239.jpg, 240.jpg, 241.jpg, 242.jpg, 243.jpg, 245.jpg, 246.jpg, 247.jpg, 248.jpg, 249.jpg, 250.jpg, 251.jpg, 252.jpg, 254.jpg, 255.jpg, 256.jpg, 257.jpg, 258.jpg, 260.jpg, 261.jpg, 262.jpg, 263.jpg, 264.jpg, 265.jpg, 266.jpg, 267.jpg, 268.jpg, 269.jpg, 270.jpg, 271.jpg, 272.jpg, 273.jpg, 275.jpg, 276.jpg, 277.jpg, 279.jpg, 280.jpg, 282.jpg, 283.jpg, 288.jpg, 289.jpg, 290.jpg, 291.jpg	63
Cluster4	003.jpg, 006.jpg, 011.jpg, 014.jpg, 015.jpg, 017.jpg, 028.jpg, 030.jpg, 035.jpg, 036.jpg, 037.jpg, 039.jpg, 040.jpg, 047.jpg, 054.jpg, 060.jpg, 072.jpg, 075.jpg, 077.jpg, 078.jpg, 079.jpg, 080.jpg, 148.jpg, 215.jpg, 236.jpg, 238.jpg, 259.jpg, 274.jpg, 278.jpg, 292.jpg	30

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-2024-205-2024 | © Author(s) 2024. CC BY 4.0 License.

Cluster5	004.jpg, 005.jpg, 012.jpg, 016.jpg, 020.jpg, 021.jpg, 022.jpg, 026.jpg, 033.jpg, 041.jpg, 043.jpg, 046.jpg, 048.jpg, 049.jpg, 051.jpg, 052.jpg, 056.jpg, 059.jpg, 061.jpg, 063.jpg, 066.jpg, 068.jpg, 074.jpg, 163.jpg, 166.jpg, 171.jpg, 178.jpg, 181.jpg, 235.jpg, 244.jpg, 285.jpg, 295.jpg	32
Cluster6	104.jpg, 105.jpg, 108.jpg, 112.jpg, 115.jpg, 116.jpg, 118.jpg, 121.jpg, 132.jpg, 140.jpg, 150.jpg, 151.jpg, 154.jpg, 158.jpg, 162.jpg, 164.jpg, 170.jpg, 172.jpg, 175.jpg, 185.jpg, 188.jpg, 194.jpg, 210.jpg, 213.jpg, 216.jpg, 223.jpg, 227.jpg, 228.jpg, 229.jpg, 230.jpg, 234.jpg	31
Cluster7	019.jpg, 044.jpg, 099.jpg, 102.jpg, 110.jpg, 111.jpg, 119.jpg, 120.jpg, 122.jpg, 127.jpg, 134.jpg, 136.jpg, 138.jpg, 168.jpg, 209.jpg, 224.jpg, 226.jpg	17
Cluster8	001.jpg, 007.jpg, 031.jpg, 055.jpg, 093.jpg, 095.jpg, 101.jpg, 103.jpg, 106.jpg, 109.jpg, 133.jpg, 147.jpg, 165.jpg, 182.jpg, 184.jpg, 190.jpg, 197.jpg, 205.jpg, 219.jpg, 220.jpg, 253.jpg, 286.jpg, 287.jpg, 293.jpg, 294.jpg	25

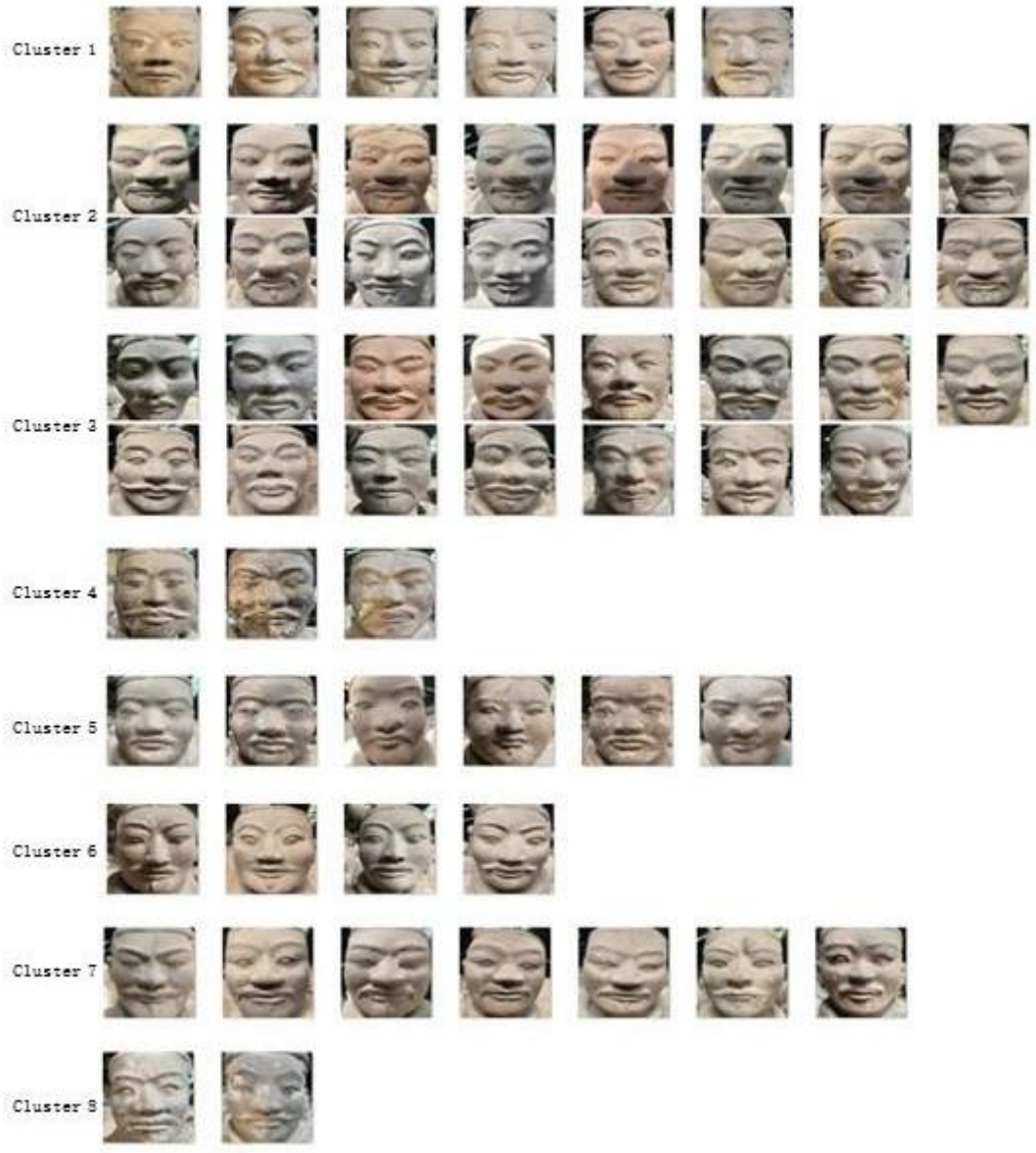

Figure 7. Clustering results of the test set (partial)

4.4 Terracotta Warrior Feature Extraction Results and Analysis

Two different feature networks, namely Inception_ResNetV1 and SqueezeNet, are employed for accuracy comparison with the proposed method. Inception_ResNetV1 is the original backbone network for feature extraction in FaceNet, while SqueezeNet is a lightweight feature extraction network. The evaluation metrics during the experimental process include the recognition accuracy on the custom Terracotta Warrior test set and the model size. The specific performance of the three backbone networks is detailed in Table 6.

Table 6 Comparative performance of facial recognition for three network models

Network model	Accuracy/%	Model size/M
Inception_ ResNetV1	92.8	135
SqueezeNet	91.5	18.1
Improved SqueezeNet	95.6	12.9

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-2024-205-2024 | © Author(s) 2024. CC BY 4.0 License.

From Table 6, it can be observed that the accuracy of the improved model is increased by 2.8% and 4.1% compared to the control network model, while the model parameters are reduced by 116.9M and 5.2M, respectively. The improved model not only significantly reduces the number of parameters but also enhances accuracy. Therefore, the improved network demonstrates faster operational efficiency and improved detection accuracy.

In the experiment comparing Euclidean distances, facial correction and cropping were applied to images before testing. When two different photos were tested, the results for the three main feature extraction networks are presented in Table 7 and Figure 8.

Table 7 Comparative euclidean distance results of three network models on different test sets

Network model	Self-made training set	Self-made test set
Inception_ ResNetV1	0.3513	0.3501
SqueezeNet	0.4581	0.4588
Improved SqueezeNet	0.2081	0.2337

In Figure 8, Inception_ResNetV1, SqueezeNet, and the improved SqueezeNet all have input dimensions of 160×160. As indicated in Table 7, the improved model exhibits smaller Euclidean distance test results, with values of 0.2081 and 0.2337 for the custom training set and test set, respectively. This suggests that the network model's judgment of the same Terracotta Warrior is more accurate after the improvement.

Figure 8. Euclidean distance test results of three network models on different test sets t

4.5 Optimization Results and Analysis of Triplet Loss Function

The Terracotta Warrior dataset was trained using both the traditional Triplet Loss function and the optimized Triplet Loss function. The loss for each epoch was recorded, and the training loss curves were plotted for comparison, as shown in Figure 9.

Training loss curves serve as a monitoring tool for tracking the model's training progress. From the graph, it is evident that the optimized Triplet Loss function accelerates model convergence during the initial 50 frozen epochs and controls convergence smoothly during the subsequent 50 unfrozen epochs, resulting in a more gradual decline in loss.

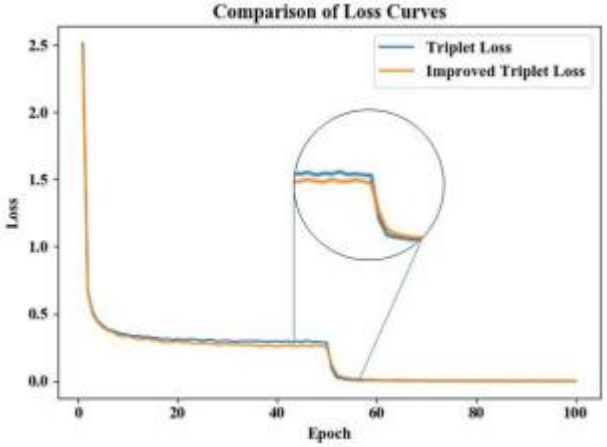

Figure 9 Training loss curve comparison

4.5 Clustering results and analysis of terracotta warrior

The facial features of 80 figurines were randomly selected from the terracotta dataset, and the 128-dimensional features of each figurine's face were extracted. Two clustering methods were used, one was clustered by the Agglomerative clustering algorithm, and the other was clustered by the K-means clustering method in SPSS software, and the clustering results of the two were compared and analyzed to verify the feasibility of the proposed method. The comparison of clustering results is shown in Table 8 below.

Table 8 Comparative results of terracotta warrior facial clustering

SPSSClustering results		The results of the clustering of the method in this paper	
Cluster1	11	Cluster1	11
Cluster2	11	Cluster2	11
Cluster3	8	Cluster3	8
Cluster4	12	Cluster4	12
Cluster5	7	Cluster5	7
Cluster6	9	Cluster6	8
Cluster7	7	Cluster7	8
Cluster8	15	Cluster8	15

From the above clustering results, it can be observed that the improved method aligns well with the overall results of the SPSS clustering method. The main differences are evident in clusters 6 and 7, particularly in G10-32.

5. CONCLUSIONS

In addressing the challenge of small sample sizes in facial recognition of Terracotta Warriors using deep learning, we proposed an improved SqueezeNet model for Terracotta Warrior facial feature extraction. This model effectively compressed the parameters involved in the Terracotta Warrior facial feature extraction process, leading to an enhanced recognition accuracy. Experimental results demonstrate that the proposed improved

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-2024-205-2024 | © Author(s) 2024. CC BY 4.0 License.

SqueezeNet model reduced the model size by 5.2M while improving recognition accuracy by 4.1%. Compared to the baseline models, this method achieved more precise recognition results on small sample data.By combining automated recognition of Terracotta Warrior facial features with Agglomerative clustering, we achieved high-accuracy clustering of Terracotta Warrior facial features. In future work, we plan to compare this method with graph convolutional neural network clustering methods that perform better on large-scale facial datasets, aiming to find the most suitable approach for Terracotta Warrior facial classification. Additionally, with the incorporation of facial recognition technology, this method can be extended to devices such as FPGA, enabling a portable Terracotta Warrior facial recognition system.

REFERENCES

Alhichri H, Bazi Y, Alajlan N, et al. Helping the visually impaired see via image multi-labeling based on SqueezeNet CNN[J]. Applied Sciences, 2019, 9(21): 4656.

Ahonen T, Hadid A, Pietikainen M. Face description with local binary patterns: Application to face recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2006, 28(12): 2037-2041.

Abiram R N, Vincent P. Identity preserving multi-pose facial expression recognition using fine tuned VGG on the latent space vector of generative adversarial network[J]. Math. Biosci. Eng, 2021, 18(4): 3699-3717.

Belhumeur P N, Hespanha J P, Kriegman D J. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection[J]. IEEE Transactions on pattern analysis and machine intelligence, 1997, 19(7): 711-720.

Ester M, Kriegel H P, Sander J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise[C]//kdd. 1996, 96(34): 226-231.

Fei Yicheng. Significance of the Discovery of the Terracotta Warriors for the Study of Ancient Military and Sculptural Art[J]. Comparative Cultural Innovation Research, 2017, 1(28): 20-21.

Gowda K C, Krishna G. Agglomerative clustering using the concept of mutual nearest neighbourhood[J]. Pattern recognition, 1978, 10(2): 105-112.

Hu Yungang, Wang Jingyang, Lan Dexing, et al. Study on Feature Extraction and Statistical Analysis of Facial Features of Terracotta Warriors based on Point Cloud Data[J]. Conservation of Cultural Heritage and Archaeological Science, 2022, 34(01): 109-117.

Hartigan J A, Wong M A. Algorithm AS 136: A k-means clustering algorithm[J]. Journal of the royal statistical society. series c (applied statistics), 1979, 28(1): 100-108.

Huo Aiqing, Zhang Wenle, Li Haoping. Traffic Sign Recognition Based on SqueezeNet Model with Deep Residual Network and GRU[J]. Computer Engineering and Science, 2020, 42(11): 2030-2036.

Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700-4708.

Mansanet J, Albiol A, Paredes R. Local deep neural networks for gender recognition[J]. Pattern Recognition Letters, 2016, 70: 80-86.

Schroff F, Kalenichenko D, Philbin J. Facenet: A unified embedding for face recognition and clustering[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 815-823.

Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.

Qin H, Yan J, Li X, et al. Joint training of cascaded CNN for face detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 3456-3465.

Turk M, Pentland A. Eigenfaces for recognition[J]. Journal of cognitive neuroscience, 1991, 3(1): 71-86.

Tammina S. Transfer learning using vgg-16 with deep convolutional neural network for classifying images[J]. International Journal of Scientific and Research Publications (IJSRP), 2019, 9(10): 143-150.

Taigman Y, Yang M, Ranzato M A, et al. Deepface: Closing the gap to human-level performance in face verification[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 1701-1708.

Yan Zifeng. Design of Face Recognition Access Control System Based on Deep Learning[D]. Xi'an: Xi'an Shiyou University, 2022.

Yan Z, Zhang H, Piramuthu R, et al. HD-CNN: hierarchical deep convolutional neural networks for large scale visual recognition[C]//Proceedings of the IEEE international conference on computer vision. 2015: 2740-2748.

Zhao Zhen, Xiao Weiguo, Xia Juxian, et al. Brief Report on the Excavation of Pit No. 1 of the Terracotta Warriors and Horses Accompanying the Mausoleum of the First Emperor of Qin from 2009 to 2011[J]. Cultural Relics, 2015, (09): 4-38+2+1.

Zhang Liangliang. Age Estimation Algorithm for Facial Images Based on Deep Learning[D]. Anhui: Anhui University of Engineering, 2020.

Zhang K, Zhang Z, Li Z, et al. Joint face detection and alignment using multitask cascaded convolutional networks[J]. IEEE signal processing letters, 2016, 23(10): 1499-1503.

Zhu C, Zheng Y, Luu K, et al. Cms-rcnn: contextual multi-scale region-based cnn for unconstrained face detection[J]. Deep learning for biometrics, 2017: 57-79.