

Article

Intelligent Evaluation of Chinese Hard-Pen Calligraphy Using a Siamese Transformer Network

Fei Yan ^{1,2}, Xueping Lan ¹, Hua Zhang ^{1,*} and Linjing Li ¹ ¹ School of Information Engineering, Southwest University of Science and Technology, Mianyang 621010, China; yanfei72021@163.com (F.Y.); lanxuepsc@163.com (X.L.); lilinjing2018@163.com (L.L.)² Mianyang Polytechnic, Mianyang 621000, China

* Correspondence: zzhh839@163.com

Abstract: The essence of Chinese calligraphy inheritance resides in calligraphy education. However, it encounters challenges such as a scarcity of calligraphy instructors, time-consuming and inefficient manual assessment methods, and inconsistent evaluation criteria. In response to these challenges, this paper introduces a deep learning-based automatic calligraphy evaluation model. Initially, hard-pen handwriting samples from 100 volunteers were collected and preprocessed to create a dataset consisting of 4800 samples, along with the corresponding label files for hard-pen calligraphy evaluation. Subsequently, YOLOv5 was utilized for region detection and character recognition on the evaluation samples to obtain the corresponding standard samples. Lastly, a Siamese metric model, with VGG16 as the primary feature extraction submodule, was developed for hard-pen calligraphy evaluation. To improve feature extraction and propagation, a transformer structure was introduced to extract global information from both the evaluated and standard samples, thereby optimizing the evaluation results. Experimental results demonstrate that the proposed model achieves a precision of 0.75, recall of 0.833, and mAP of 0.990 on the hard-pen calligraphy evaluation dataset, effectively realizing automatic calligraphy evaluation. This model presents a novel approach for intelligently assessing hard-pen calligraphy.

Keywords: calligraphy evaluation; deep learning; Siamese network; attention mechanism; transformer



Citation: Yan, F.; Lan, X.; Zhang, H.; Li, L. Intelligent Evaluation of Chinese Hard-Pen Calligraphy Using a Siamese Transformer Network. *Appl. Sci.* **2024**, *14*, 2051. <https://doi.org/10.3390/app14052051>

Academic Editor: Alexander Barkalov

Received: 3 January 2024

Revised: 10 February 2024

Accepted: 27 February 2024

Published: 29 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Learning calligraphy encapsulates the finest understanding and continuation of traditional Chinese culture. However, the resources for calligraphy education remain relatively limited, and the pressing issue of an insufficient number of specialized calligraphy instructors is conspicuous. Calligraphy evaluation holds the potential to spark enthusiasm for calligraphy learning, yet it faces challenges such as a scarcity of renowned mentors, low evaluation efficiency, and disparities in evaluation criteria. Influenced by the rapid advancement of computer and information technology, traditional calligraphy is becoming digitized in terms of creative tools, artistic effects, writing techniques, and creative methodologies, culminating in the emergence of the interdisciplinary domain of digital calligraphy research. Digital calligraphy research primarily comprises three facets: (1) digitized modeling of calligraphy tools; (2) analysis and processing of calligraphy images; and (3) representation and synthesis of calligraphic characters. These three directions within digital calligraphy research focus on a shared concern: the computational aesthetics and evaluation of calligraphy. Intelligent aesthetic evaluation poses a challenging problem within the field of artificial intelligence. Current efforts in aesthetic evaluation primarily focus on aesthetic ratings of photographic images [1], aesthetic evaluation of painted artwork images [2], aesthetic evaluation of videos [3], and aesthetic evaluation of web page layout design [4]. However, there is a relative scarcity of research on computational aesthetics and evaluation of calligraphy. The existing related research predominantly branches into two

directions, calligraphy evaluation based on feature engineering and calligraphy evaluation based on deep learning.

The application of feature engineering in calligraphy evaluation commonly integrates manually extracted features with classifiers. Common artificial features, based on traditional visual theories, include SIFT [5], HOG [6], and Gabor [7], among others. These features, extracted from images, subsequently undergo common machine learning methods for classification, such as SVM [8], RF [9], KNN [10], Xgboost [11], and LightGBM [12]. Han et al. [13] introduced an interactive calligraphy guidance system, leveraging image-processing techniques to extract quantifiable features such as the center, size, and projection of each handwritten character. These features are then subjected to fuzzy inference to render evaluations of the handwritten characters. Gao et al. [14] designed eight directional features to implement Chinese handwritten quality evaluations by analyzing confidence in online Chinese handwritten recognition. Li et al. [15] represented the topology of Chinese characters using WF histograms and employed these topological features as inputs for calligraphy evaluation via an Adaboost ensemble composed of support vector regression (SVR). Drawing inspiration from classical calligraphy principles, Sun et al. [16] proposed 22 global shape features. Incorporating a 10-dimensional feature vector for sparse coding to capture component layout information, these features are concatenated and input into an artificial neural network for calligraphy evaluation. Wang et al. [17] introduced a hierarchical evaluation approach that encompasses evaluations of entire characters and individual strokes. This approach computes comprehensive weighted similarities and transforms them into final evaluation scores. In subsequent research efforts, the team employed a skeleton-based iterative nearest point algorithm to synthesize overall and stroke-level similarities, yielding ultimate comprehensive evaluation scores [18]. Zhou et al. [19] devised three feature criteria based on Chinese calligraphy theory for extracting features of Chinese characters from calligraphy teaching materials. Subsequently, they employed a possibility–probability distribution method to implement calligraphy evaluation. Addressing challenges inherent in resistive touch screen handwriting, Xing [20] introduced a method for fuzzy evaluation of Chinese character strokes, which entails the construction of membership templates, the selection of fuzzy subsets, and the generation of template parameters. While engineering-designed artificial features have demonstrated favorable performance across a range of evaluation tasks, the method is constrained by limitations in feature quantity and representation scope. This often leads to the omission of nuanced evaluation information in calligraphy. Furthermore, the intricate nature of manual feature design results in substantial errors and limited applicability, thus constraining the effectiveness of evaluation models.

Significant progress has been achieved in image aesthetic quality evaluation through deep learning methods. For instance, in the context of aesthetic quality evaluation focused on landscape photographs in the AVA dataset, Zhang et al. [21] devised a multi-modal self-cooperative attention network (MSCAN). Meanwhile, Li et al. [22] utilized a deep Siamese network to introduce a multi-task deep learning framework for personalized and general image aesthetic evaluation. Due to the lack of public datasets for calligraphy evaluation, there is less research on calligraphy evaluation based on deep learning than in image aesthetic quality evaluation. In the field of calligraphy evaluation, Qiang et al. [23] proposed a method for categorizing the aesthetic quality of hard-pen calligraphy works by students using a CNN-based approach in 2019, achieving commendable accuracy. However, a continuous and detailed aesthetic scoring system, along with the interpretability of such scores, remains lacking. In 2022, Sun et al. introduced a framework for calligraphy aesthetic evaluation through the utilization of a Siamese network architecture coupled with transfer learning, primarily focusing on brush calligraphy evaluation [24]. In these deep learning-based calligraphy evaluation methodologies, evaluation features are mainly extracted from calligraphy images, overlooking the influence of brushstroke dynamics during calligraphy creation. Xu et al. [25] employed a TLD tracking algorithm to extract brush movement trajectories from video streams and proposed the MCNN-LSTM model to de-

rive scores of handwriting quality by comparing them with reference templates. Similarly, Wang et al. [26] collected sequential data of brushstroke creation using a nine-axis sensor, and proposed a combination of long short-term memory networks with K-nearest neighbor algorithms for calligraphy evaluation through the analysis of writing motion data. Deep learning methods have the potential to enhance the efficiency, robustness, and reliability of automated calligraphy evaluation. However, it is important to note that most existing evaluation research focuses on brush calligraphy, with limited exploration into the evaluation of hard-pen calligraphy.

With the ongoing development of our era, the hard-pen has gradually evolved into the primary writing instrument for modern work and life, and hard-pen calligraphy has emerged as a branch of traditional calligraphy. Distinguished from brush calligraphy by its writing tool, hard-pen calligraphy exhibits distinct aesthetic characteristics in its creations. It readily highlights the beauty of character form and structure but exhibits relative limitations in stroke formation and expressive penmanship. Capitalizing on this attribute, this paper primarily integrates Chinese character recognition and similarity calculation methods, applying deep learning models to automate hard-pen calligraphy evaluation. This endeavor seeks to achieve an automated evaluation of hard-pen calligraphy, as illustrated in the overall structure of the evaluation model depicted in Figure 1.

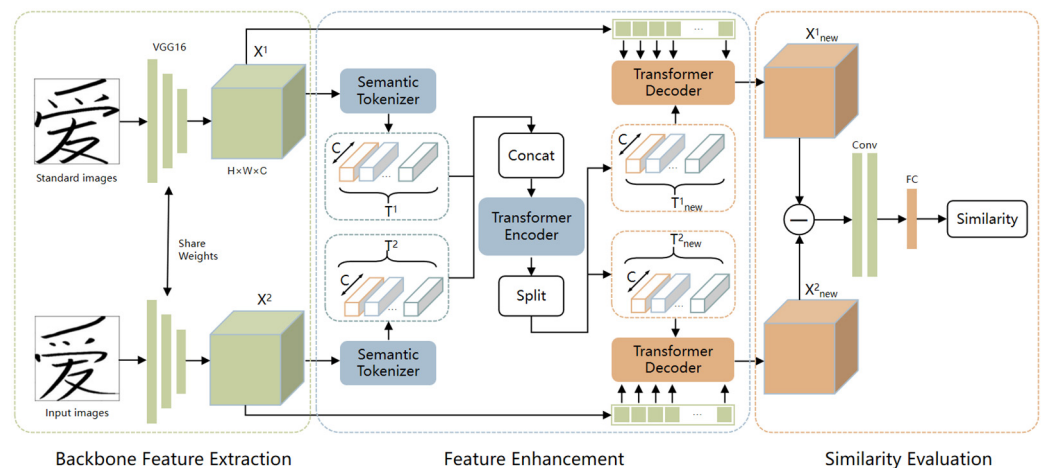


Figure 1. The overall structure of the hard-pen calligraphy evaluation model.

The main contributions of this paper are as follows:

- (1) Establishment of a dataset specifically designed for evaluating hard-pen Chinese character writing. Key features of the dataset include: 100 volunteers transcribing six randomly selected Chinese characters to reduce random errors; collection of handwriting images from volunteers with different styles to ensure sample diversity; application of appropriate preprocessing techniques to enhance image quality; and annotation of bounding boxes to enrich the dataset. The dataset caters to the experimental requirements of hard-pen calligraphy evaluation models.
- (2) Utilization of YOLOv5 with a streamlined CSPDarknet53 as the backbone feature extraction network enabled precise detection and recognition of Chinese characters, aligning each character with its corresponding standard template. This robust foundation significantly enhances the subsequent evaluation processes.
- (3) Adoption of a Siamese network architecture as the central framework, complemented by an improved VGG16 as the primary feature extraction network. The extracted feature maps are then fed into a Transformer structure to capture interrelationships, thereby enhancing the model's feature extraction capabilities. The incorporation of similarity metrics enables the realization of hard-pen calligraphy evaluation.

2. Proposed Dataset

Currently, deep learning-based network models rely primarily on data-driven approaches. Existing publicly available Chinese character datasets are mainly used for offline or online handwritten Chinese character recognition tasks. The HCL2000 dataset [27], an offline, handwritten simplified Chinese character dataset, encompasses 3755 frequently used primary-level characters. It contains handwriting samples contributed by 1000 participants from various age groups, genders, educational backgrounds, and occupations. Another dataset, HCD [28], is also an offline primary-level Chinese character dataset. It is categorized into ten subsets based on the quality of handwriting and contains samples written by 1999 participants, doubling the size of HCL2000 dataset. The CASIA-HWD dataset [29] includes samples of handwritten Chinese characters and texts, comprising 7185 Chinese characters and 171 English letters, digits, and symbols, gathered from 1020 individuals. This dataset is applicable for both offline and online handwritten Chinese character recognition tasks. However, there is currently a lack of publicly available datasets specifically designed for evaluating hard-pen writing, necessitating the creation of such a dataset. To facilitate network training and enhance model accuracy, annotation of the dataset is essential. Moreover, to improve the generalization capability of detection and recognition models, data augmentation is required.

2.1. Data Acquisition and Preprocessing

A total of six Chinese characters were randomly selected from the first-level Chinese character database, and 100 individuals were invited to write these chosen characters. The chosen characters were “爱” (love), “你” (you), “中” (middle), “华” (China), “天” (sky), and “下” (under), with each character garnering 100 writing samples. To better suit mobile application scenarios, images were captured using smartphones, resulting in a total of 600 photographs saved in JPEG format. During the capture process, the resolution of the sample images initially proved insufficient, resulting in unclear outlines of the character strokes. This adversely affected the quality of the features extracted by the evaluation model, consequently leading to a decline in model evaluation performance. Conversely, excessively high resolution not only failed to significantly improve model evaluation performance but also incurred substantial computational costs. Following testing, the optimal resolution for the captured sample images was determined to be 880×1230 . To simulate real-world application scenarios, the lighting conditions during the sample capture process were not standardized. However, variations in lighting conditions could impact the display of Chinese characters and background in the sample images. Thus, preprocessing of the collected sample images was deemed necessary to ensure effective model evaluation.

Preprocessing operations serve dual objectives. Firstly, they facilitate data augmentation, thereby creating a dataset with diverse characteristics to simulate real-world influences. This process ensures sample diversity, ultimately enhancing the model's generalization capabilities. Various types of noise, including Random noise, Gaussian noise, and Salt and Pepper noise, are intentionally introduced to the collected dataset to achieve this augmentation. Secondly, preprocessing aims to mitigate adverse effects stemming from environmental conditions, thereby improving the accuracy of model training. To facilitate subsequent recognition and processing, a thresholding method is employed to binarize Chinese character images, effectively removing interference caused by factors such as lighting variations and shadows. The thresholding techniques utilized encompass single thresholding, adaptive thresholding, and OTSU thresholding [30], as depicted in Figure 2. Among them, OTSU thresholding yields suboptimal results, primarily due to the presence of large shadow regions, which significantly impact character evaluation. Conversely, both single thresholding and adaptive thresholding demonstrate satisfactory performance, with single thresholding exhibiting notable speed advantages. To optimize preprocessing speed, the single thresholding method is selected to binarize sample images, with a segmentation threshold set at 50. Furthermore, erosion and dilation operations are conducted to eliminate residual noise post thresholding and to mitigate minor noise. To address com-

putational errors resulting from variations in character sizes, the dataset undergoes cropping to remove excess background and standardizing the images to a uniform size. The outcomes of the sample preprocessing procedures are depicted in Figure 3.

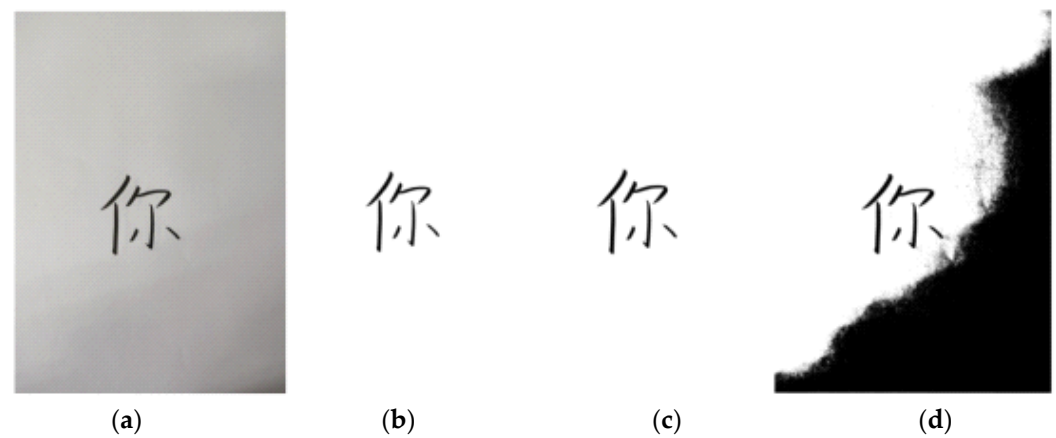


Figure 2. Comparison of preprocessing effects on Chinese character images using different thresholding methods. (a) Original image; (b) Results of single thresholding; (c) Results of adaptive thresholding; (d) Results of OTSU thresholding.

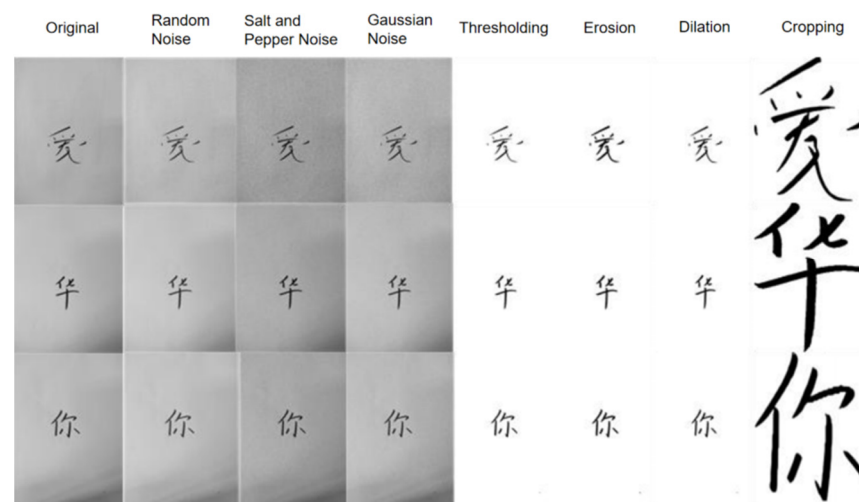


Figure 3. Sample preprocessing rendering.

2.2. Data Annotation

In accordance with the training requirements of the Chinese character detection and recognition model, the Labelling software (version 1.8.1) was employed to perform category bounding box annotation on the sample images. The chosen format for annotation boxes was YOLO-style, involving the utilization of rectangular boxes to encompass the target objects. After selecting these boxes, the specific target category was assigned, leading to the creation of a corresponding label file. These label files followed a text document format, containing relevant information such as category indices, central coordinates, and relative dimensions of the annotated boxes. Figure 4 visually depicts the process of data annotation. The Chinese character samples were organized in the “images” directory, with the corresponding label files stored in the “labels” directory. This arrangement resulted in a total of 4800 samples paired with 4800 corresponding labels. Additionally, to facilitate the training of the Siamese network within the calligraphy evaluation model, distinct categories of Chinese characters were segregated into individual folders. During the training process, a random subset of 20 samples from each character category was selected from the collected dataset.

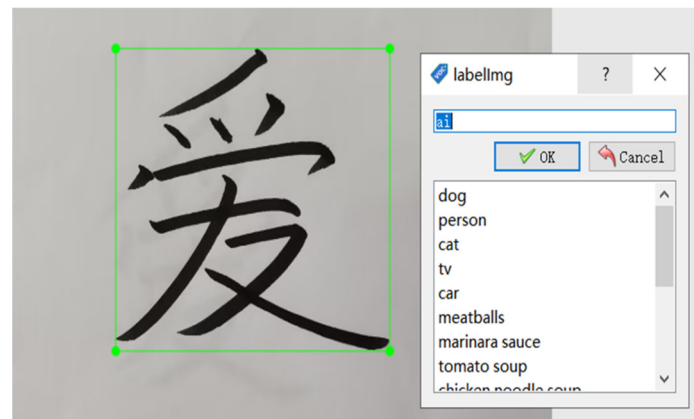


Figure 4. Data annotations.

3. Proposed Methodology

This paper introduces an intelligent calligraphy evaluation approach based on deep learning, with the overall process outlined in Figure 5. First, the evaluation dataset is established, which involves data collection and preprocessing. Subsequently, the YOLOv5 object detection network is employed to detect and recognize the input evaluation samples, yielding standardized reference samples for the evaluation candidates. Next, a Siamese model is constructed, with VGG16 serving as the backbone network. By applying upsampling operations, feature extraction is enhanced. Additionally, the transformer structure is utilized to capture contextual information from the evaluation samples, thereby enhancing both feature extraction and propagation. This comprehensive process is aimed at optimizing the evaluation outcomes of the model.

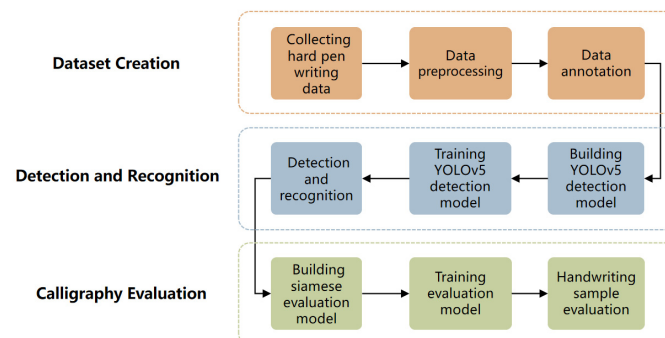


Figure 5. Overall flowchart of calligraphy evaluation.

3.1. Detection and Recognition Network

In this study, YOLOv5 is employed as the detection and recognition network to perform region detection and recognition on the target evaluation samples, thereby obtaining standardized samples for evaluation. YOLOv5 is a deep learning algorithm in the field of object detection [31], which has demonstrated significant improvements in both detection speed and recognition accuracy compared to YOLOv4. The model primarily consists of three components: backbone, neck, and head, as illustrated in Figure 6. In the backbone, key structures such as the Conv module, C3 module, and Spatial Pyramid Pooling (SPP) module are employed to process the input image through a series of convolutional and pooling layers, progressively extracting high-level feature representations from the image. The neck utilizes techniques such as convolutions at different scales, upsampling, and downsampling from the Feature Pyramid Network (FPN) to fuse feature maps from different levels, resulting in multi-scale feature representations for more comprehensive and enriched feature expression. These fused feature maps are then forwarded to the prediction layer. The head, responsible for object detection within the feature pyramid, comprises convolutional layers, pooling layers, and fully connected layers. In the YOLOv5

model, the detection head module is primarily responsible for multi-scale object detection on the feature maps extracted by the backbone network, ultimately achieving handwritten Chinese character detection and character recognition tasks.

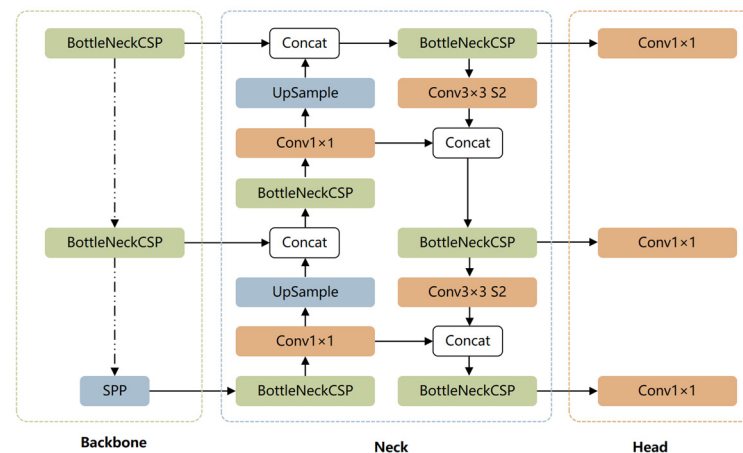


Figure 6. YOLOv5 network architecture.

3.2. Siamese Evaluation Network

The calligraphy evaluation network is built upon the Siamese network architecture [32], which incorporates an enhanced VGG16 [33] as the backbone feature extraction network. The feature maps extracted by VGG16 are then input into a Transformer structure to capture inter-relationships, thereby enhancing the model's generalization capacity. The Siamese network comprises two input channels, with identical weights and parameters across all structures within these channels. After extracting features from two input samples, two feature vectors are obtained for each sample. The similarity between the two samples is evaluated by measuring the distance between their respective feature vectors. The Siamese network can be employed to quantify the dissimilarity between two Chinese characters, making it a suitable choice as the foundational network framework for the evaluation model.

3.3. Backbone Feature Extraction Network

The calligraphy evaluation model adopts the Siamese network as the overarching structural framework, utilizing VGG16 as the backbone feature extraction network within the Siamese structure. Input images undergo convolutional processing by VGG16 to derive effective features. Images of size $256 \times 256 \times 3$ are fed into the VGG16 network for hierarchical feature extraction. Through multiple convolutional and pooling operations, and by employing the ReLU activation function after each convolution operation to strengthen the inter-layer connections, the image features are extracted, ultimately resulting in a feature matrix with a size of $4 \times 4 \times 512$. Stacks of smaller convolutional kernels are a key element of VGG16 [34]. To address the issue of diminished model accuracy due to the small feature maps resulting from multiple convolution operations, an enhancement is made to the VGG16 network. A combination of upsampling and pointwise convolution is applied to the vector processed by VGG16 to augment the feature map size and alter the feature map dimensions. The improved structure is illustrated in Figure 7. Upsampling employs bilinear interpolation [35], utilizing interpolation algorithms to calculate new values inserted between the existing matrix elements.

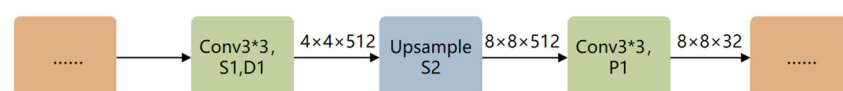


Figure 7. VGG network enhancement module.

3.4. Enhanced Feature Extraction Network

After image feature extraction by the backbone network, considering global image information can enhance the model's performance and processing efficiency. Therefore, in this study, the Transformer structure [36] is incorporated into the model framework. Since the output of the main feature extraction network consists of two feature maps, and the Transformer operates on tokens, it is necessary to transform the feature maps into token structures with high-dimensional semantic information. To represent the varying parts between the two input images using a single token, the feature maps obtained from VGG16 network processing are converted into tokens using the Semantic Tokenizer, which extracts compact semantic tokens from each temporal feature map. The processing procedure of the Semantic Tokenizer is illustrated in Figure 8. Initially, the obtained feature maps are divided into multiple submaps, each of which is mapped to a token. To incorporate spatial information, spatial attention is employed to compress the channels and extract spatial information, resulting in compact tokens with enriched semantic information.

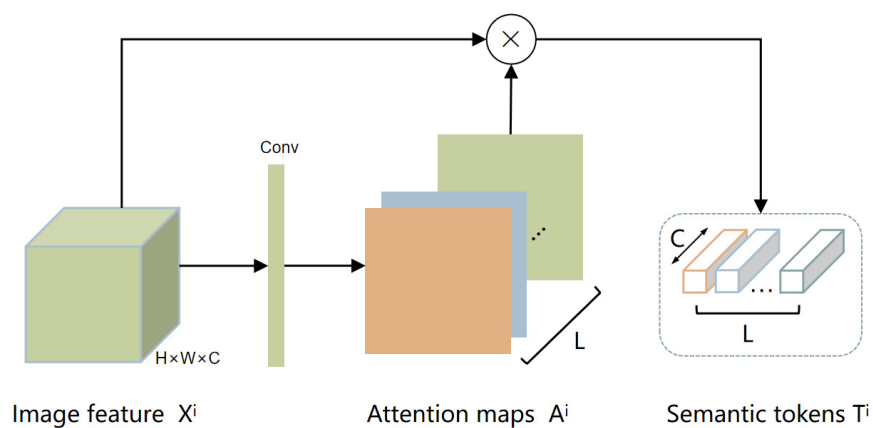


Figure 8. Flowchart of Semantic Tokenizer Processing.

To obtain concise tokens, a set of spatial attention mappings is learned using the Tokenizer, aiming to concentrate the spatial feature mapping into a set of tokens. Let X^1 and $X^2 \in \mathbb{R}^{HW \times C}$ represent the input dual-temporal feature maps, where H , W , and C are the height, width, and channel dimensions of the feature maps, respectively. Let T^1 and $T^2 \in \mathbb{R}^{L \times C}$ denote two sets of tokens, with L being the size of the token vocabulary set. The feature map of size $C \times W \times H$ is compressed into a $1 \times W \times H$ feature map, allowing the aggregation of average and maximum information along the dimension of the feature map. This aggregated information serves as the convolutional layer for extracting attention information, enabling the extraction of image attention information. This process yields L semantic groups, each representing a unique semantic concept. Subsequently, a softmax operation is applied to the HW dimensions of each semantic group, ensuring non-negative attention weights and thus obtaining spatial attention mappings. Finally, the attention mappings are employed to compute the weighted average of pixels in X^i , resulting in a compact vocabulary set of size L , referred to as semantic tokens T^i .

3.5. Encoder

As shown in Figure 9, the encoding structure of the Transformer consists of multiple layers, each composed of multi-head self-attention mechanisms and feedforward neural networks. The input to this structure must be a sequence in time. Each time step of the sequence is processed by the CNN Encoder and then enriched with positional encoding. Every time step of the sequence is inputted into the multi-layer Transformer encoder. The self-attention mechanism captures the interdependencies between the components of the input sequence by computing self-attention for each time step. In the feedforward neural network, the model transforms each time step through two linear transformations and activation functions. After processing through the encoding structure, the sequence acquires

its feature representation, which is then utilized by the decoder to generate the target image sequence.

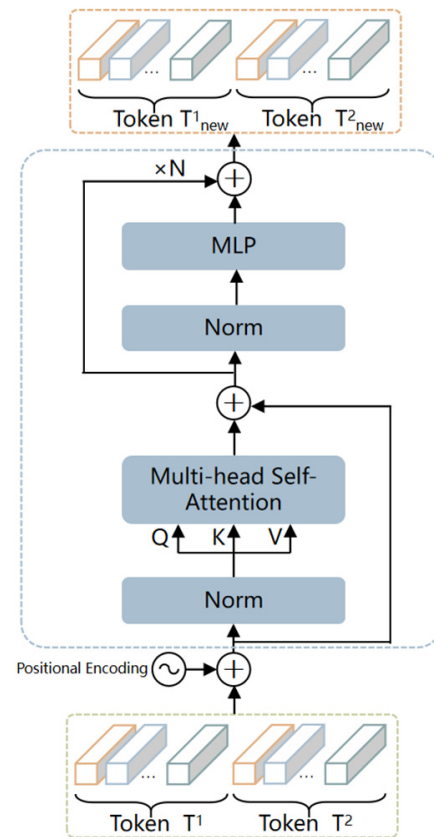


Figure 9. Encoder structure diagram.

The processed tokens T^i ($i = 1, 2$) from the Semantic Tokenizer module are concatenated, yielding the input tokens represented as $X = \{x_1, x_2, \dots, x_{2L}\}$, where x_t denotes the feature token at time step t . Positional Encoding incorporates a d -dimensional vector into each time step t to enable the model to discern the distances between various time steps. This vector is defined by Equation (1):

$$p_{i,j} = \begin{cases} \sin\left(i/10,000^{2j/d}\right), & \text{if } j \text{ is even} \\ \cos\left(i/10,000^{2(j-1)/d}\right), & \text{if } j \text{ is odd} \end{cases} \quad (1)$$

where i represents the position in the sequence, j denotes the dimension in the vector, and d is the total dimension of the vector. Subsequently, x_t is added to p_t , resulting in z_t . The matrix $Z \in \mathbb{R}^{2L \times C}$ is then subject to normalization.

To enhance the processing of multi-modal data in this study, we employ a novel normalization method called “Norm”. Norm consists of two main components: Instance Normalization (IN) and Layer Normalization (LN). IN is designed for normalizing individual modalities and primarily operates on the channel dimension of a single modality’s data. On the other hand, LN is devised for normalizing multi-modal data, where it conducts normalization on the feature vectors at each position of each modality within every sample. The normalization formula for IN is given in Equation (2) [37], wherein μ signifies the mean located along the channel dimension, σ indicates the standard deviation along the

same dimension, and γ and β are, in turn, recognized as the learnable scaling and shifting factors, respectively, within that dimension:

$$IN(z) = \gamma \frac{z - \mu}{\sigma} + \beta \quad (2)$$

The normalization formula of LN is shown in Equation (3) [38]:

$$LN(z) = \gamma \frac{z - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta \quad (3)$$

The input vectors utilizing this normalization technique comprise multiple batches, wherein ε denotes a minute constant (typically chosen as 10^{-5}). Following the application of the normalization procedure, these vectors are projected into three distinct vector spaces: Query, Key, and Value. These projections are carried out through the utilization of three parameter matrices W_Q , W_K , and W_V , each belonging to the domain $R^{d \times L}$, as shown in Equation (4):

$$Q = ZW_Q \quad K = ZW_K \quad V = ZW_V \quad (4)$$

The corresponding attention scores are computed through the dot product operation between Q and K as follows:

$$Attention(q_i, K, V) = \sum_{i=1}^T \frac{\exp(q_i k_i^T)}{\sum_{j=1}^T \exp(q_i k_j^T)} v_i \quad (5)$$

Here, the $\exp(\cdot)$ function computes the inner product between the query vector and the key vector. When calculating attention scores, the process first involves determining weights based on the similarity between the query and key vectors. Subsequently, these weights are utilized for element-wise multiplication and summation across all values. Multiple attention heads are concurrently connected to create a multi-head attention mechanism. The output, enriched with positional encodings, is concatenated with the result of the multi-head attention mechanism processing. Following normalization, this combined result is then passed through a multi-layer perceptron (MLP) in the feedforward neural network.

The role of the MLP is to perform a non-linear transformation on the input attention vectors, thereby enhancing the model's representational capacity. Within this study, this module also facilitates the transition between the self-attention module and the cross-modal fusion module. In other words, it carries out the following operation on the outcome derived from the multi-head attention mechanism:

$$MLP(z) = ReLU(zW_1 + b_1)W_2 + b_2 \quad (6)$$

Among these, $W_1 \in R^{C \times 2C}$ and $W_2 \in R^{2C \times C}$ stand as two parameter matrices, while $b_1 \in R^{C \times 2C}$ and $b_2 \in R^C$ represent two bias vectors. Following the transformation by the MLP , a more enriched and intricate representation is obtained, leading to the enhancement of model's performance. After the processing by multiple Encoder modules, a novel set of feature markers, L_{new}^i (where $i = 1, 2$), is derived.

3.6. Decoder

The architecture of the Decoder section, as depicted in Figure 10, is primarily responsible for generating the output sequence, thereby mapping the token sequence back into the pixel-level space. Specifically, the Decoder initially receives a temporally informed feature as input and then proceeds to generate the output sequence progressively, utilizing a self-attention mechanism. In this study, the decoding segment incorporates a Bitemporal Attention mechanism, extending the attention mechanism across two time steps and enabling the model to simultaneously account for relationships between the two temporal instances.

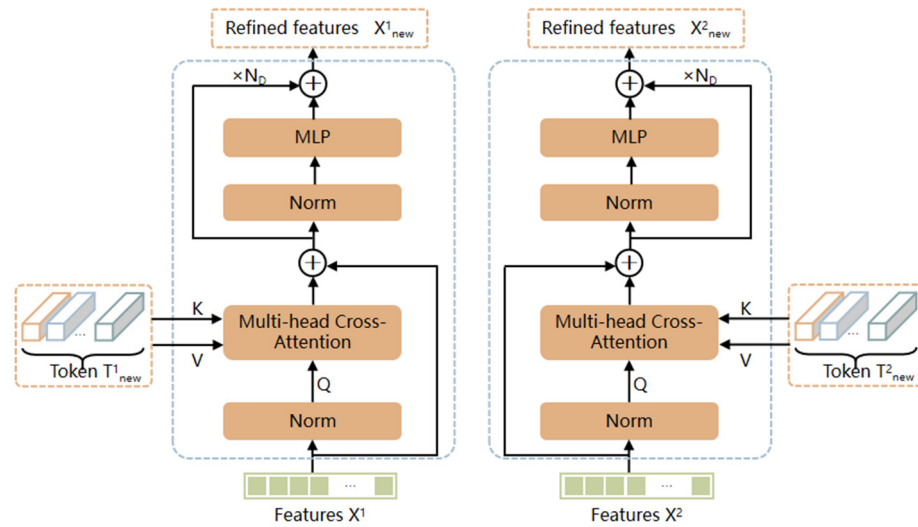


Figure 10. Decoder structure diagram.

In the Decoder section, given a sequence of features X^i , the Decoder leverages the relationship between each pixel and T_{new}^i to derive refined features X_{new}^i . The initial step involves employing the Self-Attention mechanism to learn representations for each temporal instance. Its calculation is illustrated in Equations (7) and (8):

$$MultiHead(X^{i,(l-1)}, T_{new}^i) = \text{concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (7)$$

$$\text{where } \text{head}_i = \text{Attention}(X^{i,(l-1)}W_i^Q, T_{new}^iW_i^K, T_{new}^iW_i^V) \quad (8)$$

In these two expressions above, the parameter head_i denotes the output of the i -th attention head, while the parameter matrices for the i -th attention head are represented by W_i^Q , W_i^K , W_i^V , and the output matrix is denoted as W^O . Subsequently, these are fed into the Bitemporal Attention mechanism to incorporate information across different time steps. Its expression is shown in Equation (9).

$$BitempAttention(X_1, X_2) = \text{soft max}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (9)$$

Following the processing through the attention mechanism, the data are subsequently directed into a feedforward neural network for nonlinear processing, ultimately resulting in the refined features X_{new}^i .

3.7. The Prediction Network

In total, two novel pixel-level feature maps are derived through the initial two sections, while the prediction network obtains a new differential prediction map through a sequence of operations. The absolute value of the result after subtracting the two feature maps, as illustrated in Equation (10), corresponds to the distinctive segment between the two feature maps:

$$D = |X^{1*} - X^{2*}| \quad (10)$$

The results are input into the change classifier, which consists of two 3×3 convolutional layers along with BN normalization operations. After the softmax operations, the image dimensions are transformed to $2 \times 64 \times 64$. Finally, the ultimate similarity between the two Chinese character handwriting samples is inferred from the output of the fully connected layer.

Calculating the disparity between predicted outcomes and actual labels is achieved using the binary cross-entropy loss formula, expressed mathematically as follows:

$$L = \frac{1}{N} \sum_{i=1}^N \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right] \quad (11)$$

where, N represents the total count of samples employed in training. The actual label for the i -th sample is denoted by y_i , while an alternative y_i is used to indicate the predictive outcome of the i -th sample. Within this loss function, when $y_i = 1$, the focus of the loss function is specifically on the portion where the model predicts a positive class. Conversely, when $y_i = 0$, the loss calculation solely takes into account the segment pertaining to the negative class outcomes. Throughout the training process, optimization of the model's evaluation of Chinese character handwriting samples is achieved by progressively reducing the loss function.

4. Results and Discussion

4.1. Chinese Character Detection and Recognition Experiments Analysis

The Chinese character detection and recognition model was implemented using Python version 3.7 and PyTorch version 1.5.1. Libraries such as Numpy, Torchvision, and OpenCV-Python were installed to facilitate the detection and recognition of handwritten Chinese character samples. Following multiple rounds of testing, the model was trained using a learning rate of 10^{-4} and a batch size of two.

The Chinese character detection and recognition model was implemented using the YOLOv5 detection network. Alternatively, the usage of an image classification network can also achieve satisfactory Chinese character recognition results. For comparative experiments, the well-established and computationally efficient GoogLeNet network from the realm of classification networks was selected for evaluation. As depicted in Figure 11, the GoogLeNet network exhibited an optimal training set loss of 0.366, with the loss converging around 0.40 after approximately 20 epochs. By the time the model reached six epochs, the recognition accuracy on the validation set had reached one.

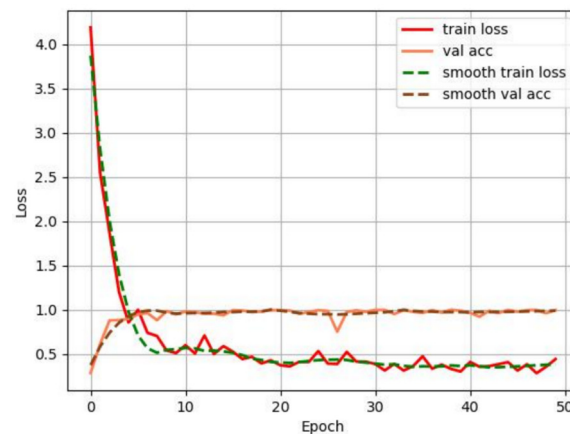


Figure 11. GoogLeNet training process.

In contrast, the training process of the YOLOv5 model, as depicted in Figure 12, illustrates the behavior of various loss components over training epochs. The “train/box_loss” gradually approached 0.005 as the epoch value neared 100, indicating the loss in predicting bounding boxes. Examining the “train/obj_loss”, which represents the loss related to the presence of target objects, the optimal value achieved was 0.003. Similarly, the “train/cls_loss”, indicating the loss pertaining to prediction results, reached its best value of 0.001. With an increase in the number of training epochs, precision, recall, and mAP values escalated, eventually reaching a maximum value of 1.

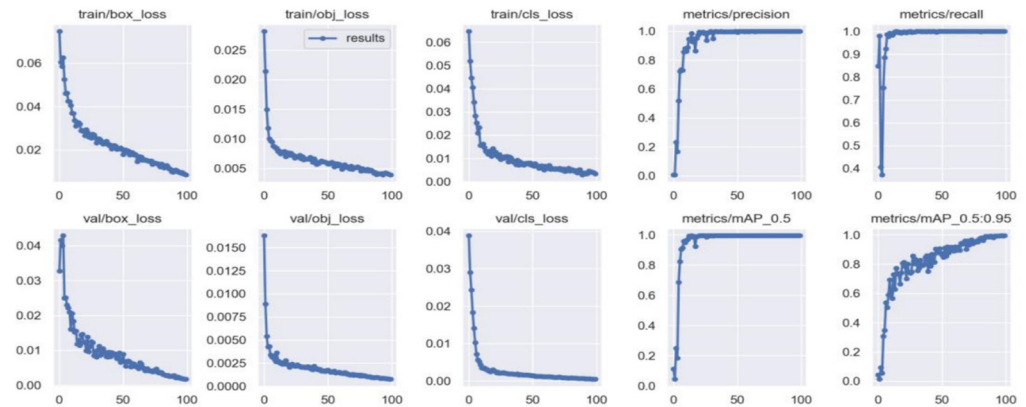


Figure 12. YOLOv5 training process.

By observing Table 1, it is evident that the YOLOv5 model outperforms the GoogLeNet network in terms of precision, recall, and mAP metrics. Testing with the same sample as illustrated in Figure 13 clearly indicates the superior accuracy of the YOLOv5 model. Therefore, it can be concluded that the performance of the YOLOv5 model surpasses that of the GoogLeNet network in this task.

Table 1. Performance comparison of different models.

Model	mPrecision (%)	mRecall (%)	mAP50 (%)
GoogLeNet	99.84	99.90	99.70
YOLOv5	99.99	100	99.95

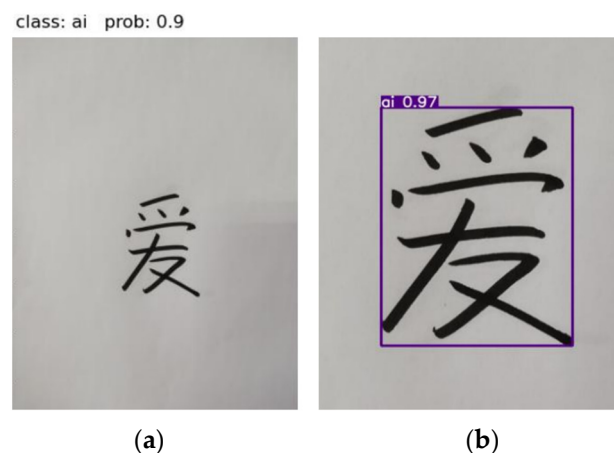


Figure 13. Comparison of test results. (a) GoogLeNet test results; (b) YOLOv5 test results.

The Chinese character detection and recognition network, based on YOLOv5, has achieved the task of detecting and recognizing Chinese characters. By utilizing the lightweight CSP-Darknet53 as the main feature extraction network, YOLOv5 reduces computational complexity, conserves storage space, and enhances inference speed. In comparison to the GoogLeNet network, this approach has resulted in a 0.10% improvement in precision, a 0.10% increase in recall, and a 0.25% enhancement in mAP metrics. These advancements signify the effectiveness of this method for Chinese character writing detection and recognition.

4.2. Analysis of Calligraphy Evaluation Experiments

The calligraphy evaluation aims to predict the similarity between two images using a Siamese network framework. VGG16 is employed as the backbone feature extraction network, while a Transformer serves as the enhanced feature extraction network to accomplish feature extraction. Then, the similarity between the written samples and standard

samples is calculated. The experimental environment is built on PyTorch and utilizes libraries such as NumPy, os, cv2, and PIL. The SGD optimizer is selected with a learning rate of 0.01, a momentum value of 0.9, and a weight decay of 0.0005. The batch size is set to two. The training process's loss variation is illustrated in Figure 14, and training concludes after 100 epochs with the minimum achieved loss of 0.178. The loss gradually decreases around the 80th epoch, converging to a steady state, reaching an optimal value.

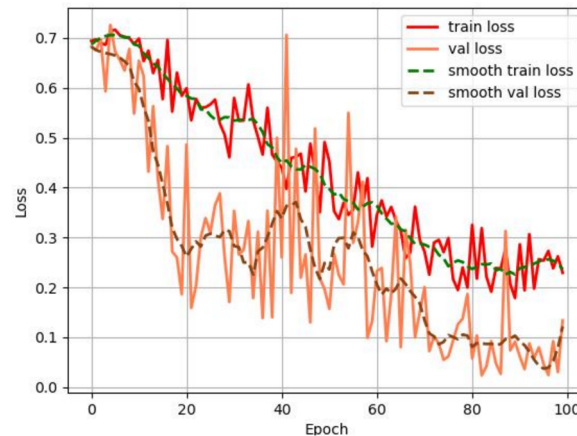


Figure 14. Calligraphy evaluation model training process loss changes.

To identify better-performing evaluation models, it is necessary to execute operations such as swapping backbone feature network models and removing network models to compare the resulting differences. The comparison involves ResNet50 and VGG16 as choices for the backbone feature extraction network. ResNet and VGG16 both belong to the category of deep convolutional neural networks (CNNs), yet they exhibit certain architectural distinctions. ResNet addresses the vanishing gradient problem through residual connections, enabling cross-layer information propagation that sustains performance in deep networks [39]. ResNet is notably deeper than VGG16, capable of surpassing 100 layers. Furthermore, to further assess the influence of the backbone feature extraction network on the calligraphy evaluation model, we exclude the VGG16 network. Instead, the transformer structure is directly employed for feature extraction, known as the ChangeFormer network—a transformer-based Siamese network. The core concept of the ChangeFormer involves employing self-attention mechanisms to establish global dependencies within input sequences. It also concurrently employs variable-length sliding windows to capture local dependencies. Similarly, to verify the role of the enhanced feature extraction network in the calligraphy evaluation model, the transformer structure is removed. This involves utilizing solely the backbone feature network for feature extraction, omitting the enhanced feature extraction network and capturing more image-feature information. Consequently, the context and positional dependencies between input sequences are disregarded. This approach aims to observe the network training outcomes under such conditions.

The variations in training loss for the calligraphy evaluation models are depicted in Figure 15. When comparing Figure 15a with Figure 14, it becomes apparent that the rate of loss decline is higher when VGG16 is utilized as the backbone feature extraction network. The minimum loss value achieved is 0.178 and is reached around the 70th epoch, with relatively minor fluctuations thereafter, indicating near-convergence to an optimal state. Conversely, employing ResNet50 as the backbone feature extraction network results in a more gradual descent of loss, with a minimum loss value of 0.385. By the 70th epoch, a similar steady state is achieved. From this, it can be deduced that VGG16 outperforms ResNet50 as the backbone network for calligraphy evaluation. Comparing Figure 15b with Figure 14, when the backbone feature extraction network is omitted and only the transformer structure is employed for feature extraction, the loss undergoes marginal fluctuations around 0.69. Notably, the ChangeFormer performs better when the epoch is less than 10. Observing Figure 15c and its consistency with Figure 14 in terms of loss trends, the network's

minimum loss reaches 0.319. Around the 80th epoch, a relatively slight variation in loss is observed, indicating a proximity to optimal convergence. This network exhibits relatively minor differences in performance compared to the network selections of this study and demonstrates an improved effectiveness compared to the model using ResNet50 as the backbone network. Finally, when comparing Figure 15d with Figure 14, it shows the loss trend for the Siamese network based on ResNet50. However, the minimum loss for this network only reaches 0.331. By the 80th epoch, the loss variation stabilizes, approaching optimal convergence.

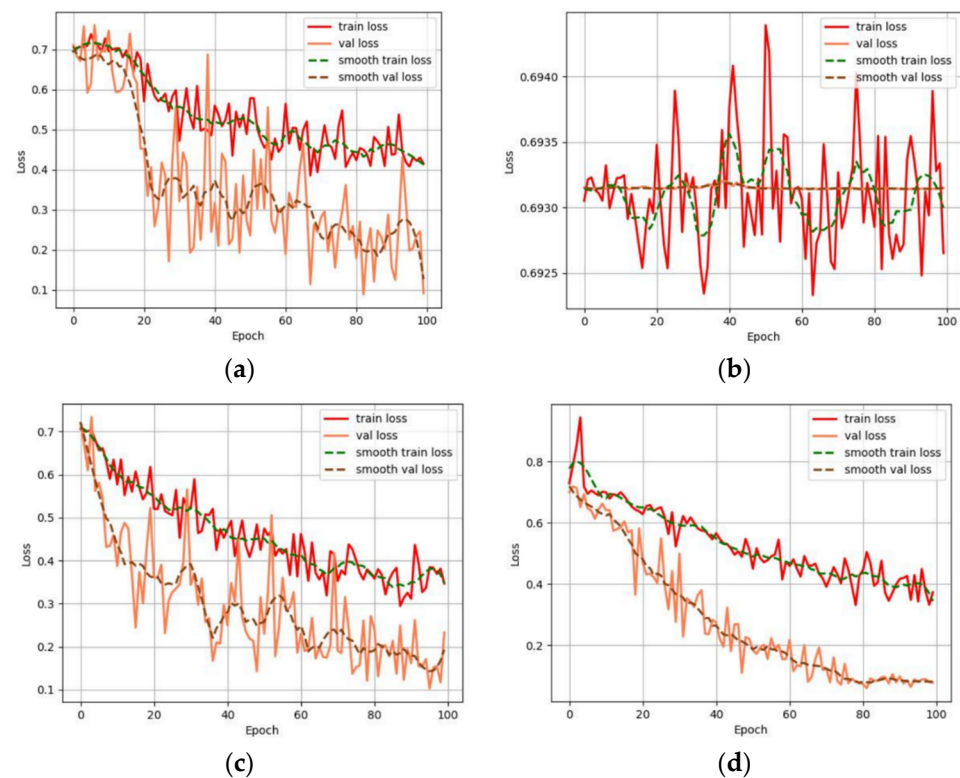
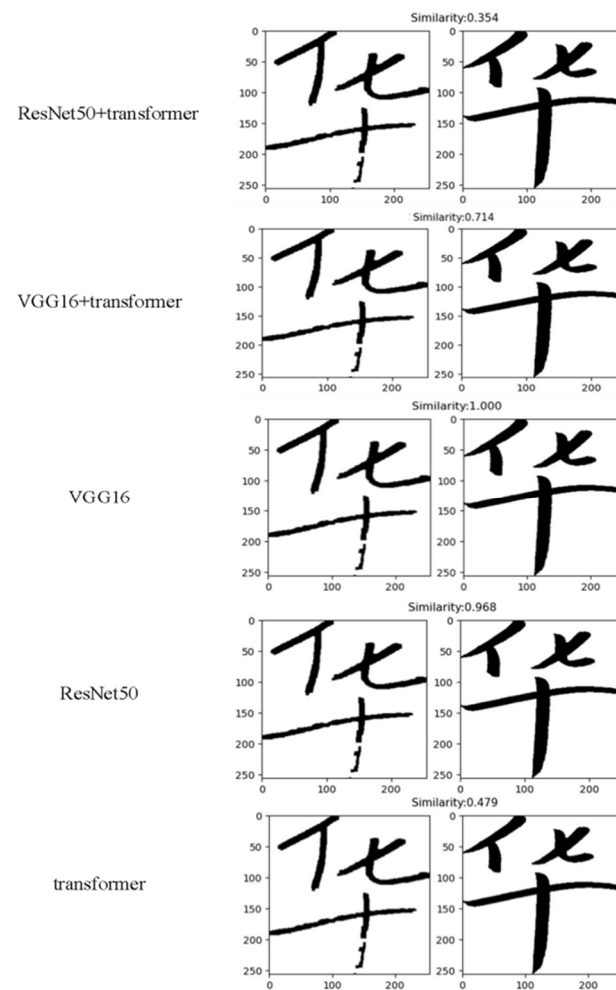


Figure 15. Calligraphy evaluation models training process loss variation. In the figure, these models are arranged in the following order: (a) ResNet50 + Transformer, (b) ChangeFormer, (c) VGG16, and (d) ResNet50.

The comparative experimental results are presented in Table 2. Without the inclusion of an enhanced feature extraction network, the Siamese network with ResNet50 as the backbone performs better. However, upon adding the transformer-enhanced feature extraction network, the network structure and parameters are finetuned. In this case, the Siamese network with VGG16 as the backbone outperforms the one with ResNet50 as the backbone. The results of the comparative experimental testing are illustrated in Figure 16, showing the effects of varying model structures on handwriting evaluations. The left side shows the handwriting sample under evaluation, while the right side shows the standard sample. Although there are slight differences in stroke forms and brushwork between the left handwriting sample and the standard one, the overall structural framework of the characters is quite similar to the standard sample. Given that hard-pen writing is relatively constrained in stroke forms and brushwork compared to brush writing, the structural framework is more critical in evaluating hard-pen calligraphy. From the results of the comparative experimental testing, it can be observed that the lowest score is 0.354 for ResNet50 + transformer, the highest score is 1 for VGG16, and the score for VGG16 + transformer is 0.714, which aligns better with the intuitive perception of the handwriting sample.

Table 2. The results of comparative experiments.

Model	Precision (%)	Recall (%)	mAP50 (%)
VGG16	57.86	71.80	99.90
ResNet50	60.20	71.80	99.60
VGG16 + transformer	75.00	83.33	99.90
ResNet50 + transformer	66.66	75.00	99.90
transformer	75.00	50.00	86.20

**Figure 16.** Comparison experiment test chart.

To further evaluate the model performance, we randomly selected 20 handwritten samples for each of the six Chinese characters from the dataset for manual assessment. A total of 10 evaluators participated, assigning scores based on the similarity between each handwritten sample and the standard sample of each Chinese character. For comparative analysis with evaluation models, scores were standardized from 0 to 1, rounded to two decimal places, resulting in 1200 manual scores. Handwritten samples were input into the evaluation models to obtain similarity scores. We employed Mean Absolute Error (MAE) and Pearson Correlation Coefficient (PCC) as evaluation metrics to compare the performance of the different structural models, as detailed in Table 3. Experimental results indicate that models incorporating a Transformer (VGG16 + Transformer and ResNet50 + Transformer) outperform others in terms of MAE and PCC. Particularly, VGG16 + Transformer demonstrated the lowest MAE (0.076) and the highest PCC (0.681), highlighting the positive impact of the Transformer, especially on the VGG16 model.

Table 3. The results of Mean Absolute Error and Pearson Correlation Coefficient.

Model	MAE	PCC
VGG16	0.090	0.610
ResNet50	0.087	0.624
VGG16 + transformer	0.076	0.681
ResNet50 + transformer	0.085	0.637
transformer	0.079	0.662

5. Conclusions

The proposed intelligent calligraphy evaluation model, based on deep learning with the Siamese structure as its fundamental framework, achieves the evaluation of hard-pen writing by calculating the similarity between the samples under evaluation and standard samples. The model utilizes an enhanced VGG16 as the backbone feature extraction network and introduces a Transformer structure to enhance the learning of interrelationships between feature maps, thereby improving the model's performance. Experimental results demonstrate that the Siamese structure is effective for the comparative evaluation of writing samples in hard-pen calligraphy. Additionally, the Transformer structure significantly enhances the evaluation model's performance.

Considering the challenges posed by the weak representation of stroke forms and brushwork in hard-pen writing, the model assesses the quality of hard-pen Chinese character writing based on the overall feature similarity. This approach provides a novel method for the automatic evaluation of hard-pen calligraphy. Specifically designed for hard-pen calligraphy assessment, the model acquires its feature extraction capability through learning from the hard-pen calligraphy evaluation dataset, demonstrating strong specificity and adaptability. However, the model is subject to constraints, including the size of the training dataset, which is relatively small, and the limited diversity of the evaluation features.

Future research could address these limitations in two ways. Firstly, by expanding the scale of the handwriting evaluation dataset, increasing the number of Chinese characters covered by the samples, and incorporating standard writing samples from renowned hard-pen calligraphers. Secondly, by integrating dynamic data from the process of hard-pen calligraphy writing as a dimension of features into the similarity evaluation, enriching the features for hard-pen calligraphy evaluation, and better reflecting individual differences in the process of hard-pen writing.

Author Contributions: Conceptualization, H.Z.; methodology, X.L., L.L. and F.Y.; resources, H.Z.; data curation, X.L.; writing—original draft preparation, X.L. and F.Y.; writing—review and editing, F.Y. and L.L.; funding acquisition, F.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Mianyang Polytechnic Foundation for Science and Technology.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

Acknowledgments: The authors would like to acknowledge the editors and reviewers who provided valuable comments and suggestions that improved the quality of the manuscript.

Conflicts of Interest: The authors declare that they do not have any conflicts of interest. This research does not involve any human or animal participation. All authors have checked and agreed with the submission.

References

- Gao, F.; Li, Z.; Yu, J.; Yu, J.; Huang, Q.; Tian, Q. Style-adaptive photo aesthetic rating via convolutional neural networks and multi-task learning. *Neurocomputing* **2020**, *395*, 247–254. [\[CrossRef\]](#)
- Zhang, J.; Miao, Y.; Zhang, J.; Yu, J. Inkthetics: A comprehensive computational model for aesthetic evaluation of Chinese ink paintings. *IEEE Access* **2020**, *8*, 225857–225871. [\[CrossRef\]](#)
- Jiang, Y.; Liu, X. Image/video aesthetic quality management based on fusing 3D CNN and shallow features. *Future Gener. Comput. Syst.* **2021**, *118*, 118–123. [\[CrossRef\]](#)
- Wan, H.; Ji, W.; Wu, G.; Jia, X.; Wang, R. A Novel Webpage Layout Aesthetic Evaluation Model for Quantifying Webpage Layout Design. *Inf. Sci.* **2021**, *576*, 589–608. [\[CrossRef\]](#)
- Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [\[CrossRef\]](#)
- Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–26 June 2005; pp. 886–893.
- Daugman, J.G. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *J. Opt. Soc. Am. A* **1985**, *2*, 1160–1169. [\[CrossRef\]](#)
- Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [\[CrossRef\]](#)
- Breiman, L.J.M.I. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [\[CrossRef\]](#)
- Cover, T.; Hart, P. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **1967**, *13*, 21–27. [\[CrossRef\]](#)
- Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
- Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. In Proceedings of the Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
- Han, C.-C.; Chou, C.-H.; Wu, C.-S. An interactive grading and learning system for chinese calligraphy. *Mach. Vis. Appl.* **2008**, *19*, 43–55. [\[CrossRef\]](#)
- Gao, Y.; Jin, L.; Li, N. Chinese handwriting quality evaluation based on analysis of recognition confidence. In Proceedings of the 2011 IEEE International Conference on Information and Automation, Shenzhen, China, 6–8 June 2011; pp. 221–225.
- Li, W.; Song, Y.; Zhou, C. Computationally evaluating and synthesizing Chinese calligraphy. *Neurocomputing* **2014**, *135*, 299–305. [\[CrossRef\]](#)
- Sun, R.; Lian, Z.; Tang, Y.; Xiao, J. Aesthetic Visual Quality Evaluation of Chinese Handwritings. In Proceedings of the IJCAI 2015, Buenos Aires, Argentina, 28 July–1 August 2015; pp. 2510–2516.
- Wang, M.; Fu, Q.; Wu, Z.; Wang, X.; Zheng, X. A hierarchical evaluation approach of learning Chinese calligraphy. *J. Comput. Inf. Syst.* **2014**, *10*, 8093–8107.
- Wang, M.; Fu, Q.; Wang, X.; Wu, Z.; Zhou, M. Evaluation of Chinese Calligraphy by Using DBSC Vectorization and ICP Algorithm. *Math. Probl. Eng.* **2016**, *2016*, 4845092. [\[CrossRef\]](#)
- Zhou, D.; Ge, J.; Wu, R.; Chao, F.; Yang, L.; Zhou, C. A computational evaluation system of Chinese calligraphy via extended possibility-probability distribution method. In Proceedings of the 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Guilin, China, 29–31 July 2017; pp. 884–889.
- Xing, L. Design and Application of a Fuzzy Evaluation Algorithm for Stroke Force in Calligraphy Teaching. *Int. J. Emerg. Technol. Learn.* **2018**, *13*, 187–200. [\[CrossRef\]](#)
- Zhang, X.; Gao, X.; He, L.; Lu, W. MSCAN: Multimodal self-and-collaborative attention network for image aesthetic prediction tasks. *Neurocomputing* **2021**, *430*, 14–23. [\[CrossRef\]](#)
- Li, L.; Zhu, H.; Zhao, S.; Ding, G.; Lin, W. Personality-assisted multi-task learning for generic and personalized image aesthetics assessment. *IEEE Trans. Image Process.* **2020**, *29*, 3898–3910. [\[CrossRef\]](#)
- Qiang, X.; Wu, M.; Luo, L. Research on Hard-tipped Calligraphy Classification Based on Deep Learning Method. In Proceedings of the 2019 8th International Conference on Educational and Information Technology, Cambridge, UK, 2–4 March 2019; pp. 275–279.
- Sun, M.; Gong, X.; Nie, H.; Iqbal, M.M.; Xie, B. Siamese regression aesthetic fusion evaluation for Chinese calligraphic copy. *CAAI Trans. Intell. Technol.* **2022**, *8*, 1077–1086. [\[CrossRef\]](#)
- Xu, P.; Wang, L.; Guan, Z.; Zheng, X.; Chen, X.; Tang, Z.; Fang, D.; Gong, X.; Wang, Z. Evaluating brush movements for Chinese calligraphy: A computer vision based approach. In Proceedings of the 27th International Joint Conference on Artificial Intelligence IJCAI 2018, Stockholm, Sweden, 13–19 July 2018; pp. 1050–1056.
- Wang, Z.; Lv, R. Design of calligraphy aesthetic evaluation model based on deep learning and writing action. In Proceedings of the International Conference on Computing, Control and Industrial Engineering, Hangzhou, China, 16–17 October 2021; pp. 620–628.
- Zhang, H.; Guo, J.; Chen, G.; Li, C. HCL2000-A large-scale handwritten Chinese character database for handwritten character recognition. In Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, Wuhan, China, 27–28 April 2009; pp. 286–290.
- Fu, Q.; Ding, X.; Li, T.; Liu, C. An effective and practical classifier fusion strategy for improving handwritten character recognition. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Curitiba, Brazil, 23–26 September 2007; pp. 1038–1042.

29. Liu, C.-L.; Yin, F.; Wang, D.-H.; Wang, Q.-F. CASIA online and offline Chinese handwriting databases. In Proceedings of the 2011 International Conference on Document Analysis and Recognition, Beijing, China, 18–21 September 2011; pp. 37–41.
30. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. [[CrossRef](#)]
31. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
32. Bromley, J.; Guyon, I.; Lecun, Y.; Sckinger, E.; Shah, R. Signature Verification Using a Siamese Time Delay Neural Network. In Proceedings of the Advances in Neural Information Processing Systems 6, 7th NIPS Conference, Denver, CO, USA, 1993.
33. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
34. Sawarkar, A.D.; Shrimankar, D.D.; Ali, S.; Agrahari, A.; Singh, L. Bamboo Plant Classification Using Deep Transfer Learning with a Majority Multiclass Voting Algorithm. *Appl. Sci.* **2024**, *14*, 1023. [[CrossRef](#)]
35. Keys, R. Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust. Speech Signal Process.* **1981**, *29*, 1153–1160. [[CrossRef](#)]
36. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
37. Ulyanov, D.; Vedaldi, A.; Lempitsky, V. Instance normalization: The missing ingredient for fast stylization. *arXiv* **2016**, arXiv:1607.08022.
38. Ba, J.L.; Kiros, J.R.; Hinton, G.E. Layer normalization. *arXiv* **2016**, arXiv:1607.06450.
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2016, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.