



ARTICLE

Multi-Stage-Based Siamese Neural Network for Seal Image Recognition

Jianfeng Lu^{1,2}, Xiangye Huang¹, Caijin Li¹, Renlin Xin¹, Shanqing Zhang^{1,2} and Mahmoud Emam^{1,2,3,*}

¹School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, 310018, China

²Shangyu Institute of Science and Engineering, Hangzhou Dianzi University, Shaoxing, 312300, China

³Faculty of Artificial Intelligence, Menoufia University, Shebin El-Koom, 32511, Egypt

*Corresponding Author: Mahmoud Emam. Email: memam@ai.menofia.edu.eg

Received: 04 September 2024 Accepted: 23 November 2024 Published: 17 December 2024

ABSTRACT

Seal authentication is an important task for verifying the authenticity of stamped seals used in various domains to protect legal documents from tampering and counterfeiting. Stamped seal inspection is commonly audited manually to ensure document authenticity. However, manual assessment of seal images is tedious and labor-intensive due to human errors, inconsistent placement, and completeness of the seal. Traditional image recognition systems are inadequate enough to identify seal types accurately, necessitating a neural network-based method for seal image recognition. However, neural network-based classification algorithms, such as Residual Networks (ResNet) and Visual Geometry Group with 16 layers (VGG16) yield suboptimal recognition rates on stamp datasets. Additionally, the fixed training data categories make handling new categories to be a challenging task. This paper proposes a multi-stage seal recognition algorithm based on Siamese network to overcome these limitations. Firstly, the seal image is pre-processed by applying an image rotation correction module based on Histogram of Oriented Gradients (HOG). Secondly, the similarity between input seal image pairs is measured by utilizing a similarity comparison module based on the Siamese network. Finally, we compare the results with the pre-stored standard seal template images in the database to obtain the seal type. To evaluate the performance of the proposed method, we further create a new seal image dataset that contains two subsets with 210,000 valid labeled pairs in total. The proposed work has a practical significance in industries where automatic seal authentication is essential as in legal, financial, and governmental sectors, where automatic seal recognition can enhance document security and streamline validation processes. Furthermore, the experimental results show that the proposed multi-stage method for seal image recognition outperforms state-of-the-art methods on the two established datasets.

KEYWORDS

Seal recognition; seal authentication; document tampering; siamese network; spatial transformer network; similarity comparison network

1 Introduction

Currently, enterprises and institutions need to process a large volume of sealed documents as part of their daily business operations. Before archiving, these sealed documents must first be recognized and verified. Manual inspection of seals is commonly performed to ensure the authenticity of the



documents, but this process is labor-intensive and prone to errors such as inconsistent placement and incomplete seals. Moreover, seal recognition presents unique challenges due to the similar appearance of different seals, including identical external shapes and central patterns [1]. Usually, seal images possess some features such as incomplete ink, angular rotation, varying types, and subtle differences in shape. Additionally, seals have a straightforward structure and relatively minimal differences in texture and color features between categories.

In the literature, some traditional feature-based seal recognition algorithms have been proposed [2–6]. These approaches heavily rely on manually designed features, which may not be robust enough to capture the subtle distinctions between seal categories. Ueda [2] introduced a statistical decision approach that used local and global features of seals for template matching. Their method registered the seals by changing their orientation and position and then extracted several features, including the coordinates of the outer boundary points, stroke direction, character line width, and mean-line width. Unfortunately, this method is vulnerable to noise attacks. Another classification method based on the matching decision of the seal skeleton was proposed by Fan et al. [3]. They extracted the refined strokes of the seal characters according to the relative stability of the seal stroke topological structure. However, this method is also sensitive to noise interference. Furthermore, Lee et al. [4] introduced a stroke feature map that combines the relationship graph and geometric position, but this technique fails when the center skeletons of two seals are alike but their stroke width is different. Moreover, Haruki et al. [5] suggested a circular seal edge feature extraction algorithm that reduced the impact of oblique stamped seals. In contrast, Horiuchi [6] segmented seal images into small regions with equal intervals and applied enhanced cyclic convolution to determine the registration rotation angle based on pixel features in these regions. However, this algorithm is sensitive to the inclination of the stamped seal, making it less effective in real-world applications.

Recent advancements in Convolutional Neural Networks (CNNs) for digital image recognition have introduced more sophisticated feature extraction and classification methods for seal recognition tasks [7,8]. CNN-based methods which are validated on generic datasets, often employ classical models like VGG16 [9] and ResNet [10] being the classical models. For instance, Yaya et al. [11] utilized ResNet-based transfer learning to transform seal text recognition into an image classification problem involving 2602 classes. Furthermore, Wang et al. [12] employed a Siamese network with multi-task learning to solve the similarity measurement problem by learning a similarity metric from the data. Additionally, the insights from handwritten character recognition can be beneficial for seal image recognition tasks. Al Hamad et al. [13] introduced a novel Multi-Layer Perceptron (MLP) based feature extraction technique that significantly improved Arabic handwriting accuracy by analyzing block density and location. Furthermore, Ou et al. [14] explored the use of Histogram of Oriented Gradients (HOG) features with a Support Vector Machine (SVM) model to recognize partially blurred Qin seal script characters. Additionally, Al Hamad et al. [15] proposed an enhanced segmentation method that detects ligatures in Arabic scripts, addressing overlapping issues using pixel density analysis. These methods highlight the potential of advanced feature extraction and segmentation techniques for improving complex image recognition tasks. Generally, seal image recognition task can be considered as a classical problem in computer vision field and can be solved by exploring the long-range dependencies in high-quality features via high-level semantic features-based methods [16], representation-learning based methods [17], metric-learning-based methods [18]. However, the ability of these methods is limited to discriminate features for recognition tasks [19].

Despite these advancements, seal recognition still faces significant challenges. The existing methods often struggle with noise sensitivity, rotation variance, and the presence of incomplete or blurred seal patterns, that result in inconsistent classification performance. Deep learning-based approaches,

while promising, typically require large labeled datasets, which may not be feasible for all seal types due to the difficulty in data collection and labeling. Furthermore, the lack of generalization to new unseen seal categories remains a major limitation. Therefore, there is an urgent need for more effective algorithms that can address all these challenges and improve the robustness and accuracy of the existing seal recognition systems. In this paper, we propose a multi-stage seal recognition algorithm based on Siamese networks. The proposed algorithm consists of two main modules: a rotation correction module based on gradient histogram and a seal similarity comparison module. The main contributions of this paper are summarized as follows:

- We introduce a multi-stage seal recognition algorithm based on a Siamese network (twin network), to overcome the limitations of the existing state-of-the-arts (SOTA) methods.
- We present a novel rotation correction module based on Histogram of Oriented Gradients (HOG) to overcome the inconsistency generated from varying angles in the seal dataset. The proposed module first aligns a pair of seal images by using their respective gradient directions as references. Then, the proposed algorithm adaptively rotates the seal by calculating the gradient magnitude and direction. Therefore, the recognition errors caused by the inconsistent angle variations between the two seal images are reduced.
- To address the problem of handling new classes of data with fixed training data categories, a Seal Similarity Network (SS-Net) based on the Siamese network is employed. The network architecture adopts a dual-path structure to solve the problem of unclassified seal images. Furthermore, the Spatial Transformer Network (STN) is used to identify the similarity between the input and the standard seal by performing an affine transformation of the input seal image.
- A new seal image dataset with two subsets (namely, SEAL48_R45 and SEAL48_R90 datasets) is established. We apply automated enhancement preprocessing methods to expand the dataset and increase the training and diversity of data and hence improve the generalization ability of the network.
- Extensive experiments are conducted on the established seal recognition dataset, and the experimental results verify the superior performance of the proposed method.

The remainder of this paper is organized as follows: [Section 2](#) presents the related works. [Section 3](#) provides an analysis of the proposed seal recognition algorithm. The experimental results and performance evaluation comparisons with some SOTA methods are presented in [Section 4](#). Finally, [Section 5](#) concludes the main contributions of this paper.

2 Basic Principles

2.1 Coarse and Fine-Grained Classification

Deep learning-based image classification algorithms can be classified into two main categories: coarse-grained classification and fine-grained classification. Coarse-grained classification involves classifying data into large categories such as animals with respect to dogs, cats, birds, and so on. Such type of classification tasks focus on identifying global features of the image such as shape, color, and texture, rather than local features as in VGG, ResNet, and similar classifiers. Furthermore, the benchmark evaluation of these models is primarily performed using popular datasets such as CIFAR-10 and CIFAR-100. On the other hand, fine-grained classification is intended to classify images into subcategories such as seal types, dog breeds, flower species, etc. This classification approach is primarily driven by identifying the local features of the image. Fine-grained classification methods can be broadly categorized into two types: strong supervision-based and weak supervision-based

approaches. The strong supervision-based fine-grained classification schemes such as Part-based R-CNN [20], Pose Normalized CNN [21], Mask-CNN [22] employ a large amount of manual effort to annotate the key regions, which makes it more expensive and complex in practical scenarios. In contrast, weak supervision-based fine-grained classification methods have been extensively researched due to their benefits in terms of ease of use and low cost.

Furthermore, weak supervision-based classification methods only need image category labels to achieve high-precision classification, unlike strong supervision-based classification methods that require additional annotation information such as bounding boxes. Classic weak supervision-based fine-grained classification methods include Recurrent Attention Convolutional Neural Network (RA-CNN) [23], Multi-attention CNN (MA-CNN) [24], Bilinear CNN (B-CNN) [25], Multi-Resolution CNN (MR-CNN) [26], etc. These algorithms can automatically identify and recognize the most discriminative features in an image without any additional information, usually by increasing the dimension of the features [27].

2.2 Siamese Network

Siamese neural network (SNN) or simply Siamese network [28] is a class of artificial neural network that have two or more subnetworks with the same network configurations. They are widely employed in various applications of image recognition [29,30], due to their ability to learn similarities between input data effectively, such as speech recognition, text similarity matching, image retrieval, face verification, and other applications [31–33]. These SNN networks are usually used for signature verification on American checks to determine whether the customer's signatures match the bank's reserved signature. Siamese networks are conjoined neural network models where the configurations in the network are shared. The commonly used Siamese network model includes two parallel networks that separately map input data to low-dimensional vectors. Then, a similarity measurement module takes the two vectors as input and calculates their similarity. Consequently, classification or regression tasks can be then performed based on the obtained similarity value. Fig. 1 shows a simple binary classification Siamese network with logistic prediction P .

2.3 Spatial Transformer Network

Convolutional Neural Network (CNN) can learn to be translation and rotation invariance through the convolutional and max pooling layers. However, Spatial Transformer Networks (STN) [34] explicitly endow the network with rotation invariance by performing affine transformations on the feature maps including translation, rotation, and scaling. Overall, STN have demonstrated substantial improvements in various real-world applications [35,36]. Hamdi et al. [35] extend STN and introduced an end-to-end multi-view transformation network for shape recognition. Their method improved the accuracy of learning spatial transformations for the input without using any additional supervision or altering the learning procedure. Additionally, Sinclair et al. [36] utilized STN to enhance segmentation and registration for 2D and 3D medical imaging. Their approach effectively reduced noise in the target images, improved pixel-wise predictions, and outperformed traditional models in this domain. These applications highlight the practical benefits of incorporating STNs in various domains to achieve more robust and adaptable models.

As shown in Fig. 2, STN consists of three main parts: a localization network, a grid generator, and a sampler. This differentiable module transforms the input feature map to an output feature map through a 2D affine transformation in a forward pass.

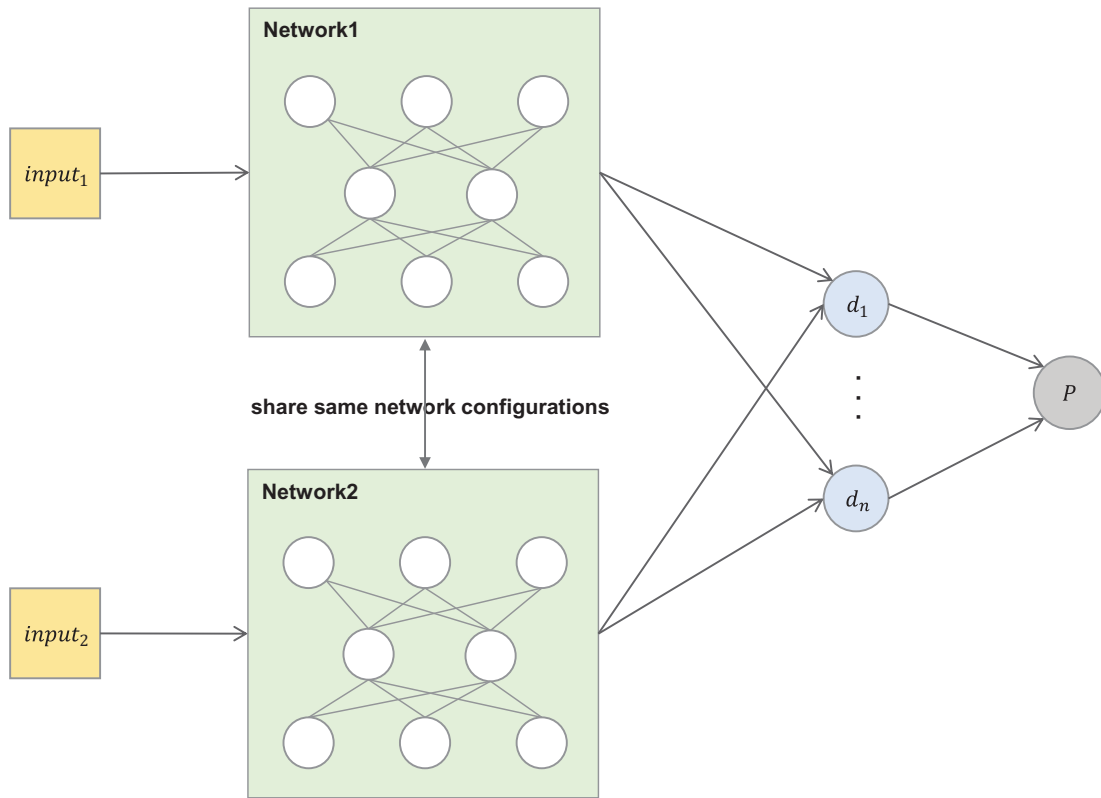


Figure 1: Simple structure for siamese network

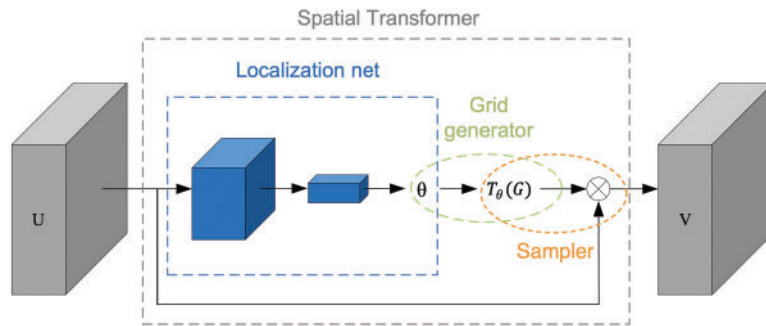


Figure 2: Spatial transformer network

The localization network takes in a feature map $U \in \mathbb{R}^{W \times H \times C}$ and outputs the parameter θ that is then applied to the input feature map as a transformation. In 2D affine transformations, θ is a six-parameter matrix. The parameter θ is determined by the fully connected layer function f_{loc} in the localization network that maps to an affine matrix.

Then, the grid generator outputs a parameterized sampling grid for the output feature map V . The grid is first generated based on the input feature map, followed by the application of the $T_\theta(G)$ transformation obtained in the previous step to obtain the corresponding position of the grid in the output feature map.

Finally, the sampler can use a differentiable sampling algorithm, usually bilinear interpolation, to retrieve the values from the input feature map that corresponds to the output feature map. Both the sampling and interpolation must be differentiable to ensure that the gradient updates propagate to both the input feature map and the sampled grid coordinates during the backpropagation process. The coordinates can provide partial derivatives regarding the affine transformation parameters, which optimize the affine transformation matrix [37,38].

3 Proposed Method

This paper presents a seal recognition algorithm that employs a Siamese network, and the specific processes of the proposed algorithm are shown in Fig. 3. The algorithm is composed of two modules, with the output from the first module serving as input to the next one. The first module is a rotation correction module that reduces the angle differences between two seal images. To be precise, the rotation correction module is based on a Histogram of Oriented Gradients (HOG). The second module adopts a Siamese network as the central model structure and introduces a seal similarity comparison network SS-Net (Seal Similarity Network). The network improves the accuracy of seal recognition by incorporating STN to obtain affine transformations that are more conducive to seal image similarity comparison. Finally, the algorithm combines the two modules and matches the recognized seal with the standard seal templates to complete the seal recognition task. The proposed algorithm utilizes a small amount of labeled seal data for training and can recognize both categories; the known labeled classes and new classes that are not included in the training set.

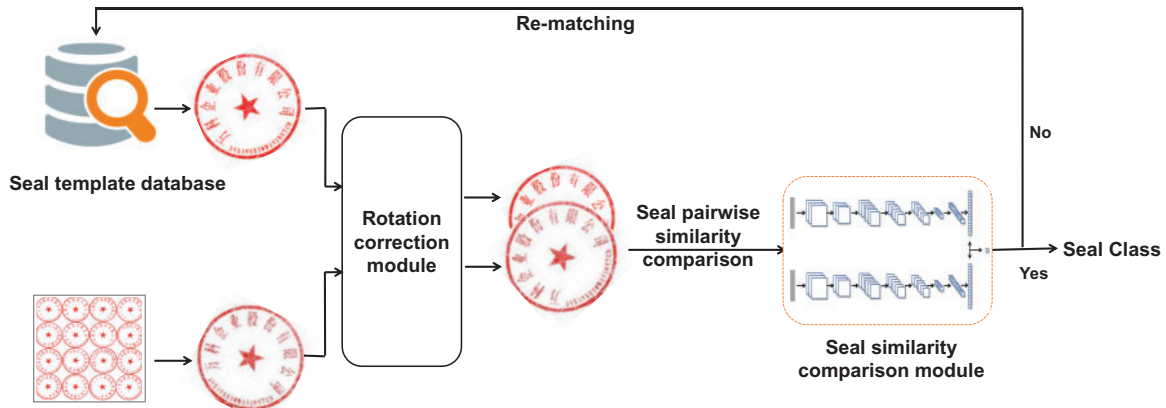


Figure 3: The flow of the proposed multi-stage seal recognition algorithm based on the siamese network

3.1 Rotation Correction Module Based on HOG

HOG is a well-known feature descriptor in computer vision, which is designed to record image gradient magnitude and orientation and it is widely used to develop image feature extraction algorithms for tasks such as object detection and human pose recognition.

We adopt a technique to map the seal image into b bins of gradient histogram to adjust and standardize the seal angle. The horizontal axis of the histogram consists of b units, dividing the angle range into b parts. Each bin corresponds to the angle range size R , which is related to the image gradient direction. The vertical axis of the histogram represents the cumulative sum of the image gradient magnitudes within that bin. This methodology enables the extraction of the main gradient

direction of the seal image and the calculation of the offset for angle correction. Algorithm 1 shows the specific steps of the proposed rotation correction module based on HOG, which can be summarized as follows:

1. The input seal image undergoes grayscale conversion and filtering processing to obtain the grayscale image of the seal to be corrected.
2. The grayscale seal image is then converted into a gradient histogram to determine the value of x_m .
3. The angle of rotation A is determined by $A = x_m \times R$, followed by rotating the corrected seal image I .

Algorithm 1: Rotation correction module based on histogram of oriented gradients

input: The seal to be corrected denoted by I ; Number of bins denoted by b , angle range of each bin denoted by R

Output: The corrected image of the seal, denoted by I_r

1 $R \leftarrow 360/b$

2 $H_I = HogHist(GrayAndBlur(I))$

3 $x_m = argmax(H_I)$

4 $A = x_m \times R$

5 $I_r = rotate(I, A)$

6 **return** I_r

7 /*The function $HogHist()$ represents a mapping function for gradient histogram*/

8 /*The function $GrayAndBlur()$ is a grayscale filtering function.*/

9 /*The function $argmax()$ returns the index of the highest peak of the gradient histogram.*/

10 /*The function $rotate(I, a)$ represents the image rotation function, where i represents the input image and a represents the rotation angle.*/

11 /* H_I : gradient histogram; x_m : index of the highest peak; A : deviation angle*/

In this module, a pair of seal images are entered into it. These images are then rotated to correct their orientation, and to align the two seal images with their respective primary gradient direction as shown in Fig. 4. In experiments, the value of b is set to 60 in this module. It is worth mentioning that various images of the same seal type usually have a comparable primary gradient direction after filtering. Consequently, normalizing these images by rotating them towards roughly the same orientation can avoid reducing the recognition accuracy of the seal images caused by a significant difference in angles.

3.2 Similarity Comparison Module

The proposed seal similarity comparison network SS-Net (Seal Similarity Network) is a twin network structure with shared weights in the left-right path, as shown in Fig. 5. As the network's input size is predefined, it is necessary to adjust the single-channel seal data graph of varying sizes to a uniform size of 105×105 before using it in situations where the seal's dimensions are non-uniform.

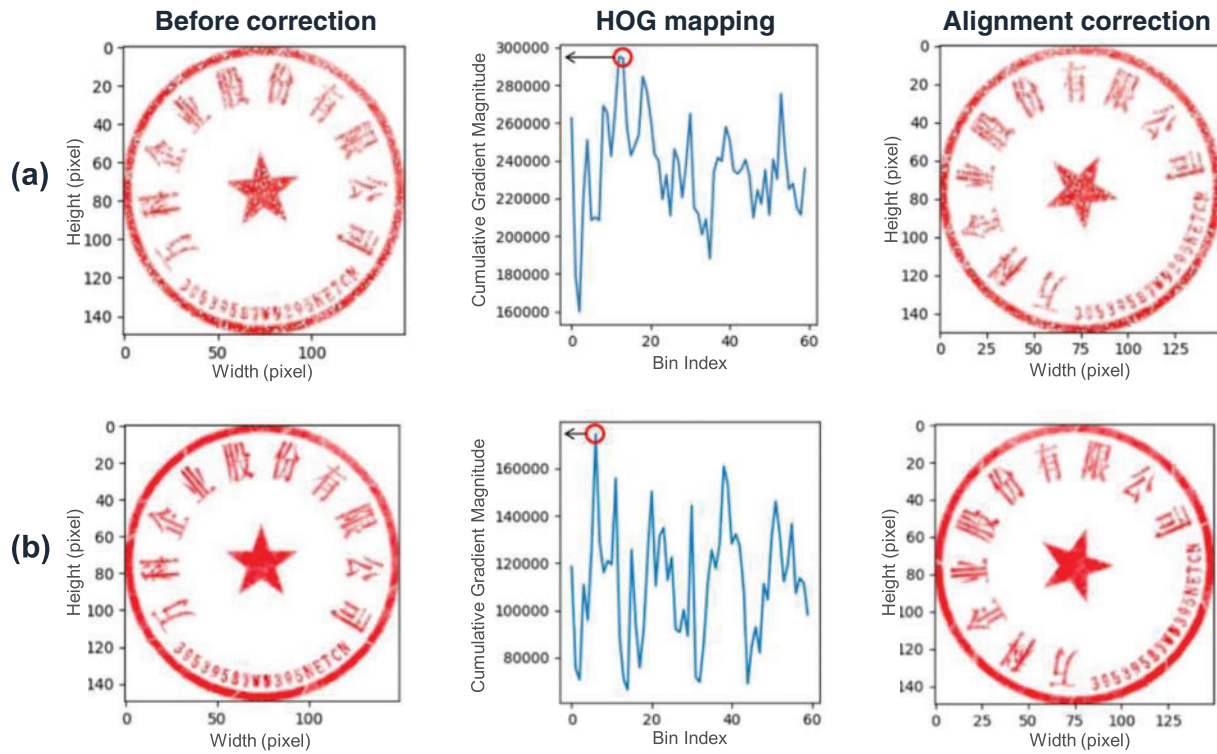


Figure 4: Schematic diagram of the rotation correction module based on HOG. (a) The input seal image to be recognized, (b) The output seal image template of the same category

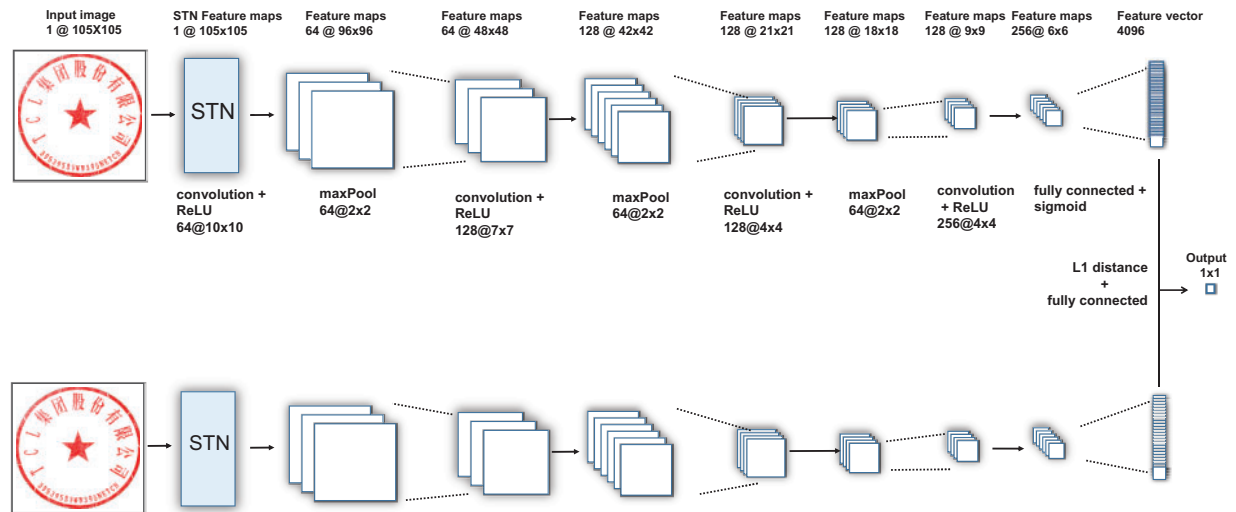


Figure 5: The proposed SS-Net network architecture

The network structure of the proposed similarity comparison model contains a sequence of convolutional layers with two identical path structures. Each layer uses a variable-sized, single-channel convolution kernel, with a number of convolution kernels being a multiple of 16 and a stride of 1. The

ReLU activation function is employed to map the output features. Furthermore, the network uses a maximum pooling layer of size 2×2 and a stride of 2. The last convolutional layer's unit is mapped to a 4096-dimensional vector through a fully connected layer. Since this model adopts a dual-route structure, two 4096-dimensional vectors are generated. The L1 distance between the two vectors is calculated and inputted into the final Sigmoid activation function. The model outputs either 0 or 1, to directly determine whether the input seal pair belongs to the same class or not.

Based on the model of the twin network, we employ STN at the outset of the two network inputs, as shown in Fig. 5. The STN network orients the input seal image to fit the standard seal shape by performing an affine transformation, which helps in the classification. The unclassified images are compared with each template in the database using the Database Matching (DM) algorithm, which assists in identifying the seal category.

The STN has an input and output size of 105×105 . The detailed parameters of one of the networks are shown in Tables 1 and 2.

Table 1: Detailed parameters of STN blocks

Block	Layer	Output	Parameters
STN Block 1	Conv2d	[8, 96, 96]	808
	MaxPool2d	[8, 48, 48]	
	ReLU	[8, 48, 48]	
STN Block 2	Conv2d	[10, 42, 42]	3930
	MaxPool2d	[10, 21, 21]	
	ReLU	[10, 21, 21]	
STN Block 3	Linear	[32]	141152
	ReLU	[32]	
STN Block 4	Linear	[6]	198

Table 2: Detailed parameters of siamese blocks

Block	Layer	Output	Parameters
Siamese Block 1	Conv2d	[64, 96, 96]	6464
	ReLU	[64, 96, 96]	
Siamese Block 2	MaxPool2d	[64, 48, 48]	401536
	Conv2d	[128, 42, 42]	
	ReLU	[128, 42, 42]	
Siamese Block 3	MaxPool2d	[128, 48, 48]	0
Siamese Block 4	Conv2d	[128, 18, 18]	262272
	ReLU	[128, 18, 18]	
Siamese Block 5	MaxPool2d	[128, 9, 9]	0

(Continued)

Table 2 (continued)

Block	Layer	Output	Parameters
Siamese Block 6	Conv2d	[128, 18, 18]	524544
	ReLU	[128, 18, 18]	
	Linear	[4096]	
Siamese Block 7	Sigmoid	[4096]	37752832
	Linear	[1]	
	Sigmoid	[1]	

The SS-Net is trained using a binary cross-entropy loss function, which can be defined in Eq. (1).

$$Loss = -\frac{1}{n} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \quad (1)$$

The binary labels $y_i = 0, 1$ indicate that image pairs of the same category are labeled with 1, and those with different categories are labeled with 0. The probability $p(y_i)$ represents the membership of the network output to the label y_i .

3.3 Network Training and Seal Recognition

To train the proposed seal similarity comparison network, we randomly select a pair of seal images from both categories (same and different categories). Then, the orientation of the seal image pairs is aligned using the proposed rotation correction module, as shown in Fig. 6. Furthermore, some pre-processing operations are applied such as grayscale conversion and filtering, to enhance the quality of the training set. Moreover, to improve the recognition performance of the proposed model, the rotation correction module is employed to adaptively correct the orientation of the seal images and eliminate the impact caused by the inclination and distortion of the images.

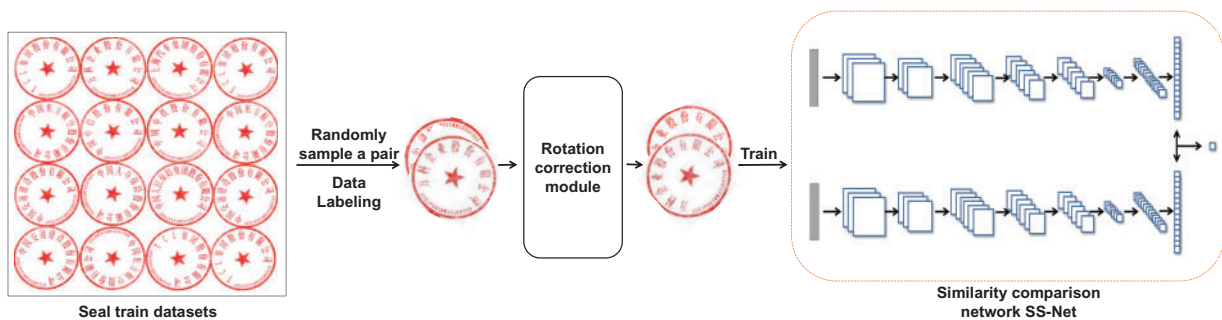


Figure 6: Training process of the proposed seal similarity comparison network SS-Net

We obtain 210,000 valid labeled pairs of seal images for seal recognition dataset, after multiple rounds of random sampling. These samples cover a wide range of different seal types that ensure the representativeness and diversity of the dataset. The similarity comparison network is used for the final seal recognition task after completing the successful training process.

The trained network is intrinsically a similarity comparison algorithm, thus only able to distinguish whether two seals belong to the same category. To address this limitation, we present a multi-stage seal recognition algorithm for precise identification. Our proposed approach integrates the rotation correction module and the similarity comparison module with the database matching technique (DM). We outline the algorithm's specific steps in Algorithm 2.

Algorithm 2: Recognizing the type of seal in C classes

input: the seal image to be recognized, x , and a standard template database of C classes,

$$T = t_1, t_2, \dots, t_C$$

output: the type of seal to be recognized, C_x

1 $x_r \leftarrow RC(x)$

2 **for** $i \leftarrow 1$ **to** C **do**

3 $tr_i \leftarrow RC(t_i)$

4 **if** $Net(x_r, tr_i) == 1$ **then**

5 $C_x \leftarrow i$

6 **end**

7 **end**

8 **return** C_x

9 /*The function $RC()$ represents the rotation correction module based on gradient histogram.*/

10 /*The function $Net()$ represents the similarity comparison module.*/

4 Experimental Results and Analysis

4.1 Dataset Preparation

In this paper, we establish a new seal image dataset that covers different types of seal images with white background, as shown in Fig. 7. Seals have detailed specification standards based on various unit categories, including external shapes, internal patterns, diameter sizes, and font types. Insufficient data can prevent models from learning sophisticated feature representations. Furthermore, it can lead to overfitting in seal recognition tasks. To enhance the generalization capability of the similarity-matching network, a large and diverse dataset of seal samples from different unit categories should be obtained for the training process. Additionally, we expand the dataset by applying automated enhancement preprocessing methods, to improve the generalization ability of the network. The variations found in imprints of the same seal are often the result of the contact between the seal and the stamped object, due to some factors such as force, angle, position, wear, and contamination of the seal surface. In fact, these differences can surpass the variations found in imprints from divergent seal types. To overcome this challenge, this paper compiles the most representative seal templates from the database for the recognition task of particular seal types.

Real-world conditions in seal collection and management introduce unique challenges, such as defective images, incomplete ink prints, and inconsistent angles in recognition tasks. As a result, this paper focuses on three types of attacks on the original seal images for data augmentation: noise attack, smearing attack, and rotation attack, as shown in Fig. 8. The noise attack aims to emulate situations where the seal's ink print is incomplete during stamping. The smearing attack replicates scenarios where background factors, like text writing lines, overlap with obscure seal images, consequently inducing partial incompleteness in the foreground and background segmentation of the seal. Lastly, the rotation attack simulates the conditions where the collected seal image is misaligned.



Figure 7: Some standard seal images from the dataset (Given the sensitivity of the data, the fingerprint information presented in this figure has undergone de-identification and it is only for academic research purposes)

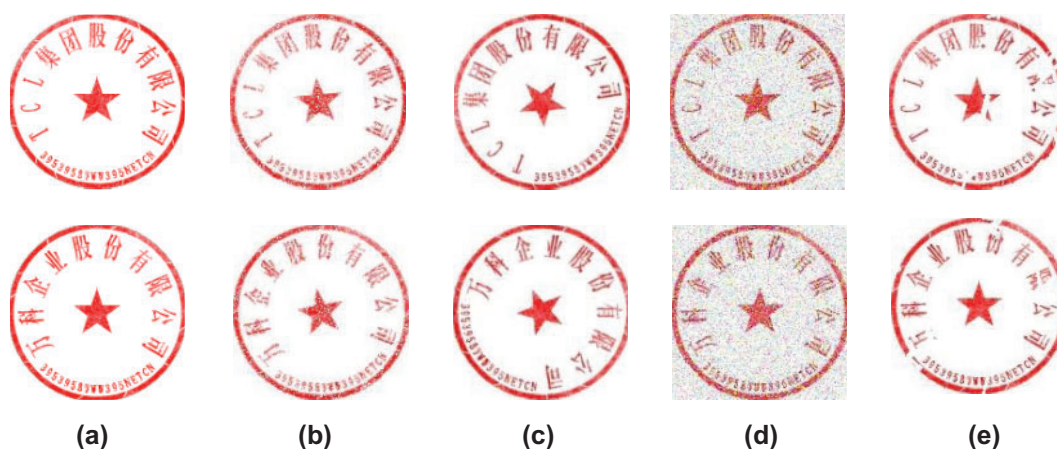


Figure 8: Dataset augmentation. (a) Two original seal images from the dataset, (b) corresponding seal images after salt-and-pepper noise attack, (c) corresponding seal images after rotation attack, (d) corresponding seal images after Gaussian noise attack, (e) corresponding seal images after smear attack (Given the sensitivity of the data, the fingerprint information presented in this figure has undergone de-identification and it is only for academic research purposes)

Following the completion of data augmentation, we establish two datasets for seal classification, namely: SEAL48_R45 and SEAL48_R90. The details about these datasets are listed in [Table 3](#). The SEAL48_R45 dataset consists of 48 classes of seal images, with 100 seal samples in each class. Each seal image is randomly rotated clockwise and counter-clockwise with a small angle from 0° to 45° . Similarly, SEAL48_R90 dataset contains 48 classes of seal images, with each class containing 100 samples. Each seal image is then randomly rotated clockwise and counter-clockwise with a small rotation angle from 45° to 90° . After multiple rounds of random sampling, we obtain a total of 210,000 valid labeled pairs of seal images, for the seal recognition dataset.

Table 3: An overview about the two established datasets used

Dataset	Number of classes	Number of seal images in each class	Rotation angle
SEAL48_R45	48	100	(0°: 45°)
SEAL48_R90	48	100	(45°: 90°)

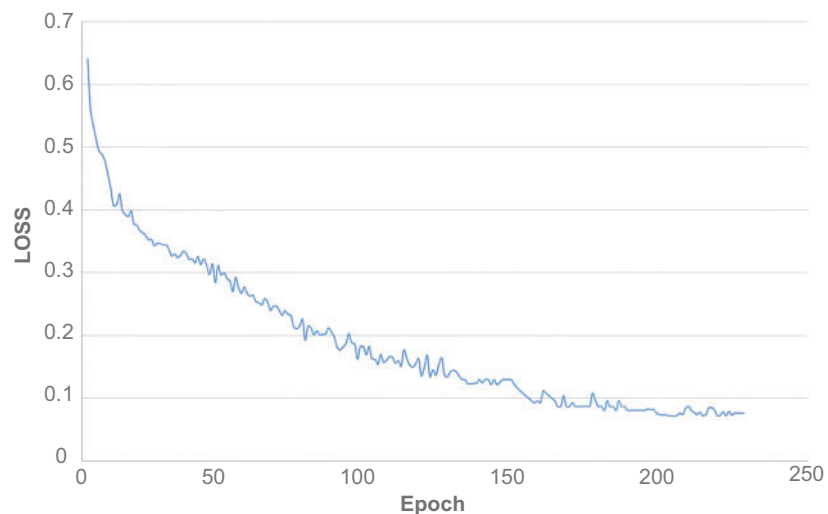
4.2 Implementation Details

In the experiments, an Intel(R) Core(TM) i7-10700 CPU @ 2.90 GHz and a GeForce RTX 3090Ti GPU are used. The operating system employed is Linux 5.4.0-42-generic. For training the seal similarity comparison network SS-Net, we set the batch size to 128 and the learning rate to 0.001. The number of iterations is set to a total of 4000, while the ADAM optimization algorithm is utilized for training.

To demonstrate the feasibility and effectiveness of the proposed approach for seal recognition, we create a new seal image dataset based on 48 original seal templates. We obtain 200 sets of testing data with tagged examples of 9600 by randomly augmenting and sampling the test set. Each set includes one pair of identical seal images and 47 pairs of distinct images. Furthermore, we design experiments that involve the following: (1) comparing the proposed method with the traditional 1-Nearest Neighbor clustering algorithm [39]. (2) comparing the proposed approach with non-similarity comparison classification methods such as VGG16 [9], ResNet [10], and B-CNN algorithm [25].

4.3 SS-Net Network Training

This section aims to examine the training process of the SS-Net similarity comparison network. As shown in Fig. 9, the training of the proposed SS-Net network presents a smooth loss curve, which gradually converges from the 200th epoch. The observed behavior indicates that the training process of the SS-Net network is relatively simple and converges rapidly, which makes it effortless to train a similarity comparison network for actual seal recognition scenarios.

**Figure 9:** Training loss of SS-Net network

4.4 Comparative Experiment

To evaluate the performance and validate the effectiveness of the proposed algorithm, we conduct a set of experiments on the two established seal datasets; namely SEAL48_R45 and SEAL48_R90. Comparison analyses of various image classification models are performed which include: 1-Nearest Neighbor method [39], VGG16 [9] and ResNet50 [10], and B-CNN [25] for fine-grained classification. Additionally, we employ conventional similarity comparison metrics such as MSE, SSIM, and COS with database matching techniques to carry out seal recognition experiments. The experimental results are shown in Table 4. The results indicate that the proposed algorithm outperforms the other comparing models. The proposed algorithm achieves an accuracy of 92.54% on SEAL48_R45 dataset. Similarly, the proposed algorithm is more effective than the other comparing models on the SEAL48_R90 dataset.

Table 4: Comparison of accuracy results among different methods using the established datasets

Method	SEAL48_R45 (%)	SEAL48_R90 (%)	Parameters (M)
1-Nearest neighbor [39]	30.45	26.41	–
VGG16 (VGG_D) [9]	78.01	75.32	138
Resnet50 [10]	80.31	76.31	26
B-CNN (VGG_D, VGG_D) [25]	79.84	76.43	135
B-CNN (VGG_D, VGG_M) [25]	81.35	75.95	140
Ours	92.54	91.43	79

To provide more comprehensive information about the recognition system's performance, we present the confusion matrix on the two established seal datasets (SEAL48_R45 and SEAL48_R90), as shown in Fig. 10. The color in each cell indicates the probability of the actual class being predicted as the current class. Additionally, the results of the accuracy (ACC) for each class on SEAL48_R45 and SEAL48_R90 are calculated as shown in Fig. 11.

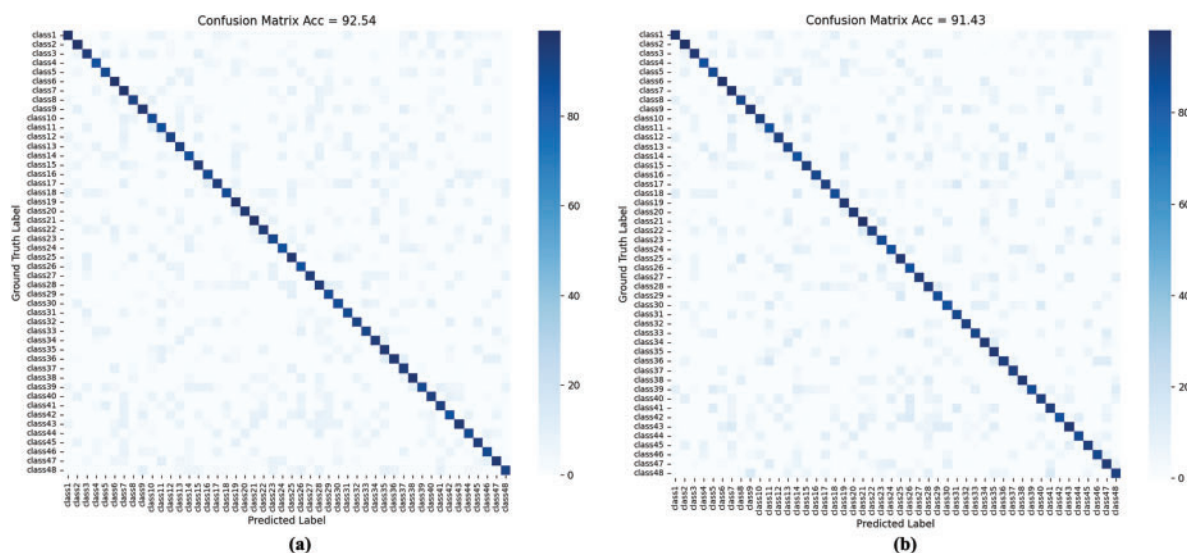


Figure 10: Confusion matrix results. (a) On SEAL48_R45 dataset, (b) on SEAL48_R90 dataset

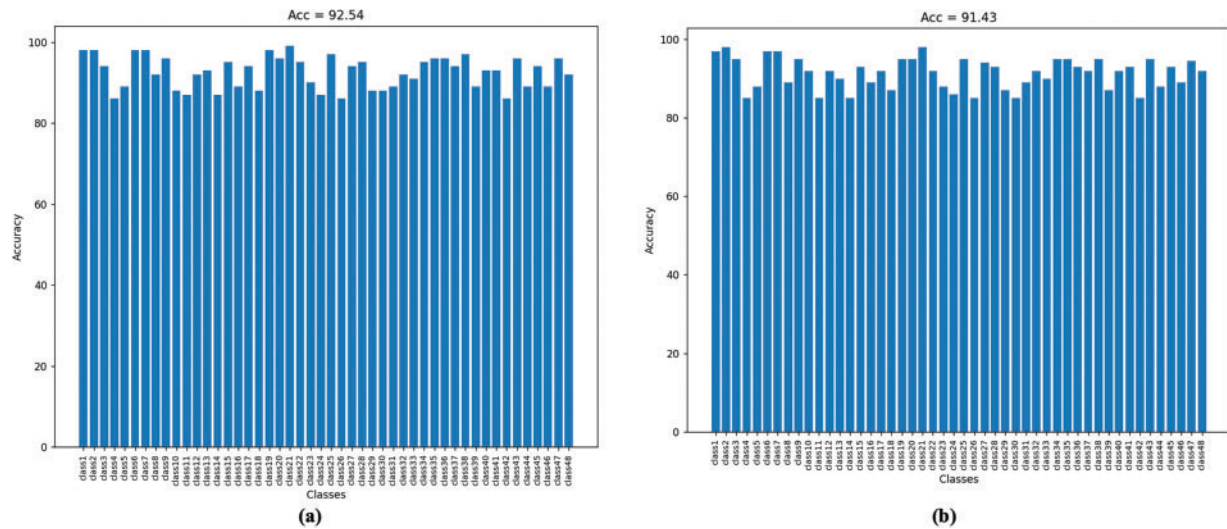


Figure 11: Results of accuracy (ACC) for each class. (a) On SEAL48_R45 dataset, (b) on SEAL48_R90 dataset

Furthermore, it can be observed that as the rotation angle increases, the recognition difficulty also rises. However, the proposed algorithm effectively handles large-angle rotations. Traditional clustering-based methods, such as 1-Nearest Neighbor, exhibit very low recognition accuracy for seals and are nearly unusable. Compared to deep learning-based methods, the proposed algorithm achieves superior recognition accuracy, exceeding the SOTA methods by more than 10% on SEAL48_R45 and more than 15% on SEAL48_R90. Additionally, aside from ResNet50, our model has significantly fewer parameters than other deep learning-based models that highlights its efficiency and advantages in resource utilization. This reduction in parameter count not only facilitates faster training and inference times but also improves the model's applicability in real-world scenarios where computational resources may be limited. Thus, the proposed approach demonstrates a balance between performance and efficiency that make it a robust solution for seal recognition tasks under varying conditions.

4.5 Ablation Study

4.5.1 Ablation Experiment for Rotation Correction Module

By considering the different rotation angles of the seal images, we propose to use a rotation correction module based on HOG to correct and align the input seal images along their respective principal gradient directions. Therefore, to validate the feasibility of the algorithm, it is necessary to conduct ablation experiments.

This subsection aims to investigate the effectiveness of the rotation correction module based on HOG on the accuracy of the proposed seal recognition algorithm. In experiments, the baseline method is used where the seal pair is directly inputted into the seal similarity comparison module without passing through the rotation correction module based on HOG. The effect of the rotation correction module is evaluated based on the comparison between the proposed approach and the baseline. The comparative evaluation is carried out on SEAL48_R45 and SEAL48_R90 datasets and the experimental results are shown in Table 5, where RC stands for rotation correction. The experimental findings confirm that the rotation correction module based on HOG has a significant

effect on the recognition rate of the proposed algorithm. Specifically, it improves the accuracy rate by 2.29% and 1.93% on SEAL48_R45 and SEAL48_R90 datasets, respectively.

Table 5: The influence of the RC module on seal image recognition accuracy

Method	SEAL48_R45 (%)	SEAL48_R90 (%)
Baseline (without RC)	90.25	89.50
Baseline + RC	92.54	91.43

4.5.2 Ablation Experiment for Spatial Transformer Network Module

The STN module enables the network to learn spatial transformations, effectively enhancing the ability of the neural network to process spatial information in feature maps. In this experiment, we investigate the impact of the STN module on the accuracy of the proposed seal recognition algorithm. The experimental settings involve a comparison between the proposed approach and the baseline method where the STN module is removed from the seal similarity comparison network SS-Net. The experimental results of the comparative analysis between the proposed approach and baseline have been presented in Table 6. The experimental results indicate that the presence of the STN module has a positive impact on the recognition rate of the proposed algorithm. Compared with the Baseline, the proposed algorithm enhances the accuracy rate by 0.71% on SEAL48_R45 dataset. Similarly, it improves the accuracy rate by 1.89% on the SEAL48_R90 dataset.

Table 6: The impact of STN module on the recognition rate of seal images

Method	SEAL48_R45 (%)	SEAL48_R90 (%)
Baseline (without STN)	91.83	89.54
Baseline + STN	92.54	91.43

5 Conclusions and Future Work

This paper proposes a multi-stage seal recognition algorithm based on the Siamese network. The algorithm begins with an image rotation correction module utilizing gradient histogram analysis to correct and normalize seal images, effectively adapting to various seal orientations and reducing recognition errors. Following this, an improved Siamese network integrated with an STN module is applied for seal recognition, enabling the extraction of distinctive features from the seal data and accurately determining whether the input seals belong to the same category through similarity comparison. Experimental results show that the proposed method outperforms existing similarity comparison approaches in terms of accuracy and reliability, while also exhibiting superior generalization capabilities compared to traditional CNN-based recognition models. Furthermore, it demonstrates resilience to new seal types, allowing for one-time training and continuous usage. In the future work, we aim to develop a fully automatic approach from segmentation to recognition which will be able to automatically segment the seal region without any touched characters from the background and then identify the seal, through preprocessing steps such as color separation and morphological operations. Besides, we will explore using adaptive pooling techniques such as adaptive average pooling or adaptive max pooling after the convolutional layers, to better enhance

robustness for input size variations. Furthermore, as the database size increases, the algorithm's efficiency decreases significantly due to the computational demands for the matching step. To address this issue, we will try to explore more efficient matching algorithms such as locality-sensitive hashing (LSH) or approximate nearest neighbor (ANN) search, which can significantly reduce search time for large datasets. Moreover, we may involve using dimensionality reduction techniques such as principal component analysis (PCA) or deep feature embedding, to minimize computational costs during matching step. Additionally, potential adaptations of the algorithm for applications beyond seal recognition such as ancient character recognition, biometric identification, or industrial component inspection, could further expand its impact and utility.

Acknowledgement: The authors would like to thank all anonymous reviewers for their helpful comments and suggestions.

Funding Statement: This work was funded by the National Natural Science Foundation of China (Grant No. 62172132), Public Welfare Technology Research Project of Zhejiang Province (Grant No. LGF21F020014), and the Opening Project of Key Laboratory of Public Security Information Application Based on Big-Data Architecture, Ministry of Public Security of Zhejiang Police College (Grant No. 2021DSJSYS002).

Author Contributions: The authors confirm their contribution to the paper as follows: Jianfeng Lu: Conceptualization, Methodology, Supervision, Project administration, Funding acquisition, Validation, Writing—review & editing. Xiangye Huang: Conceptualization, Methodology, Software, Data curation, Validation, Resources, Writing—original draft, Writing—review & editing. Caijin Li: Software, Validation, Conceptualization, Resources, Writing—original draft, Writing—review & editing, Visualization. Renlin Xin: Validation, Software, Resources, Data curation, Writing—review & editing, Visualization. Shanqing Zhang: Conceptualization, Supervision, Resources, Formal analysis, Validation, Writing—review & editing, Visualization. Mahmoud Emam: Conceptualization, Methodology, Validation, Writing—original draft, Writing—review & editing, Formal analysis, Visualization. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The dataset generated and analysed during the current study are available online in the following link: https://drive.google.com/file/d/1bF4k37cy3bW11Pr9F0kWR4IKFzkg_smF/view?usp=drive_link (accessed on 22 November 2024).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Huang J, Liu Y, Huang Y, Chen S. Seal2Real: prompt prior learning on diffusion model for unsupervised document seal data generation and realisation. arXiv preprint arXiv:231000546. 2023.
2. Ueda K. Automatic verification of seal-impression pattern. In: Proc 7'th ICPR, 1984; vol. 2, p. 1019–21.
3. Fan TJ, Tsai WH. Automatic Chinese seal identification. *Comput Vis Graph Image Process*. 1984;25(3):311–30. doi:10.1016/0734-189X(84)90198-1.
4. Lee S, Kim JH. Unconstrained seal imprint verification using attributed stroke graph matching. *Pattern Recognit*. 1989;22(6):653–64. doi:10.1016/0031-3203(89)90002-2.

5. Haruki R, Horiuchi T, Yamada H, Yamamoto K. Automatic seal verification using three-dimensional reference seals. In: Proceedings of 13th International Conference on Pattern Recognition, 1996; Vienna, Austria: IEEE; vol. 3, p. 199–203.
6. Horiuchi T. Automatic seal verification by evaluating positive cost. In: Proceedings of Sixth International Conference on Document Analysis and Recognition, 2001; Seattle, WA, USA: IEEE; p. 572–6.
7. Chauhan R, Ghanshala KK, Joshi R. Convolutional neural network (CNN) for image detection and recognition. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), 2018; Jalandhar, India: IEEE; p. 278–82.
8. Huang Z, Li K, Jiang Y, Jia Z, Lv L, Ma Y. Graph Relearn Network: reducing performance variance and improving prediction accuracy of graph neural networks. *Knowl-Based Syst.* 2024;301:112311. doi:10.1016/j.knosys.2024.112311.
9. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556. 2014.
10. 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; p. 770–8.
11. Yaya C, Quanxiang L, Kaili W, Yaohua Y. Historical Chinese seal text recognition based on ResNet and transfer learning. *J Comput Eng Appl.* 2022;58(10):125–39.
12. Wang Z, Lian J, Song C, Zheng W, Yue S, Ji S. CSRS: a Chinese seal recognition system with multi-task learning and automatic background generation. *IEEE Access.* 2019;7:96628–38. doi:10.1109/ACCESS.2019.2927396.
13. Al Hamad HA, Shehab M. Integrated multi-layer perceptron neural network and novel feature extraction for handwritten Arabic recognition. *Int J Data Netw Sci.* 2024;8(3):1501–16. doi:10.5267/j.ijdns.2024.3.015.
14. Ou Y, Zhou ZJ, Kang DW, Zhou P, Liu XW. Qin seal script character recognition with fuzzy and incomplete information. *Baghdad Sci J.* 2024;21(2 (SI)):0696–6.
15. Al Hamad HA, Shehab M. Improving the segmentation of Arabic handwriting using ligature detection technique. *Comput Mater Contin.* 2024;79(2):2015–34. doi:10.32604/cmc.2024.048527.
16. López-Cifuentes A, Escudero-Vinolo M, Bescós J, García-Martín Á. Semantic-aware scene recognition. *Pattern Recognit.* 2020;102:107256. doi:10.1016/j.patcog.2020.107256.
17. Luo T, Guan S, Yang R, Smith J. From detection to understanding: a survey on representation learning for human-object interaction. *Neurocomputing.* 2023;543:126243. doi:10.1016/j.neucom.2023.126243.
18. Li X, Yang X, Ma Z, Xue JH. Deep metric learning for few-shot image classification: a review of recent developments. *Pattern Recognit.* 2023;138:109381. doi:10.1016/j.patcog.2023.109381.
19. Huang B, Bai A, Wu Y, Yang C, Sun H. DB-EAC and LSTR: DBnet based seal text detection and lightweight seal text recognition. *PLoS One.* 2024;19(5):e0301862. doi:10.1371/journal.pone.0301862.
20. Zhang N, Donahue J, Girshick R, Darrell T. Part-based R-CNNs for fine-grained category detection. In: *Computer Vision–ECCV 2014*, 2014 Sep 6–12; Zurich, Switzerland: Springer; p. 834–49.
21. Branson S, Van Horn G, Belongie S, Perona P. Bird species categorization using pose normalized deep convolutional nets. arXiv preprint arXiv:14062952. 2014.
22. Wei XS, Xie CW, Wu J. Mask-CNN: localizing parts and selecting descriptors for fine-grained image recognition. arXiv preprint arXiv:160506878. 2016.
23. Fu J, Zheng H, Mei T. Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017; Honolulu, HI, USA; p. 4438–46.
24. Zheng H, Fu J, Mei T, Luo J. Learning multi-attention convolutional neural network for fine-grained image recognition. In: Proceedings of the IEEE International Conference on Computer Vision, 2017; Venice, Italy; p. 5209–17.

25. Lin TY, RoyChowdhury A, Maji S. Bilinear CNN models for fine-grained visual recognition. In: Proceedings of the IEEE International Conference on Computer Vision, 2015; Santiago, Chile; p. 1449–57.
26. Xu K, Lai R, Gu L, Li Y. Multiresolution discriminative mixup network for fine-grained visual categorization. *IEEE Trans Neur Networks and Learning Systems*. 2021;34(7):3488–500.
27. Chen C, Han D, Chang CC. MPCCT: multimodal vision-language learning paradigm with context-based compact Transformer. *Pattern Recognit*. 2024;147:110084. doi:10.1016/j.patcog.2023.110084.
28. Koch G, Zemel R, Salakhutdinov R. Siamese neural networks for one-shot image recognition. In: *ICML Deep Learning Workshop*, 2015; Lille, France; vol. 2.
29. Liu X, Gao W, Li R, Xiong Y, Tang X, Chen S. One shot ancient character recognition with siamese similarity network. *Sci Rep*. 2022;12(1):14820. doi:10.1038/s41598-022-18986-z.
30. Wu L, Wang Y, Gao J, Li X. Where-and-when to look: deep siamese attention networks for video-based person re-identification. *IEEE Trans Multimed*. 2018;21(6):1412–24.
31. Wang F, Cao P, Wang X, He B, Sun F. SiamADT: siamese attention and deformable features fusion network for visual object tracking. *Neur Process Lett*. 2023;55(6):7933–50. doi:10.1007/s11063-023-11290-5.
32. Lu J, Li S, Guo W, Zhao M, Yang J, Liu Y, et al. Siamese graph attention networks for robust visual object tracking. *Comput Vis Image Underst*. 2023;229:103634. doi:10.1016/j.cviu.2023.103634.
33. Tao X, Zhang D, Ma W, Hou Z, Lu Z, Adak C. Unsupervised anomaly detection for surface defects with dual-siamese network. *IEEE Trans Ind Inf*. 2022;18(11):7707–17. doi:10.1109/TII.2022.3142326.
34. Jaderberg M, Simonyan K, Zisserman A, Kavukcuoglu K. Spatial transformer networks. *Adv Neural Inf Process Syst*. 2015;28:1–9.
35. Hamdi A, Giancola S, Ghanem B. MVTN: multi-view transformation network for 3D shape recognition. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021; Montreal, QC, Canada; p. 1–11.
36. Sinclair M, Schuh A, Hahn K, Petersen K, Bai Y, Batten J, et al. Atlas-ISTN: joint segmentation, registration and atlas construction with image-and-spatial transformer networks. *Med Image Anal*. 2022;78:102383. doi:10.1016/j.media.2022.102383.
37. Liao L, Guo Z, Gao Q, Wang Y, Yu F, Zhao Q, et al. Color image recovery using generalized matrix completion over higher-order finite dimensional algebra. *Axioms*. 2023;12(10):1–23.
38. Xinde L, Dunkin F, Dezert J. Multi-source information fusion: progress and future. *Chin J Aeronaut*. 2023;37(7):24–58.
39. Wohlhart P, Kostinger M, Donoser M, Roth PM, Bischof H. Optimizing 1-nearest prototype classifiers. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013; Portland, OR, USA; p. 460–7.