International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 4, Number 1, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v4n1.17
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Image Privacy Item Recognition Based on Hybrid Model of Hierarchical Feature Recognition and ViT

Chengyuan Liu *

School of Computing and Data Science, Xiamen University, Selangor, Malaysia
*Corresponding Author: l2698136050@163.com

ABSTRACT

With the development of artificial intelligence technology, privacy risk detection has become particularly important in scenarios such as intelligent monitoring and identity authentication. However, existing technologies have shortcomings in complex scenarios and global feature processing, resulting in low detection accuracy in some cases. This paper proposes a hybrid model that combines CNN, OCNN and Transformer models to extract features and achieves higher detection accuracy. This method innovatively combines the advantages of different feature extraction methods and improves the ability to identify privacy risks. Experimental results show that the proposed method outperforms existing technologies on multiple test sets, not only improving detection accuracy but also reducing false alarm rates.

KEYWORDS

Deep convolutional neural network; Privacy protection; Hierarchical feature extraction; Vision Transformer; Secure image processing

1. INTRODUCTION

1.1. Research Background

Mobile devices have transformed photo sharing on social networks. Although sites like Facebook let users set viewing permissions, these sites rarely analyze photo content pre-upload, and deleting uploaded photos permanently is nearly impossible. Consequently, photos may expose sensitive information such as home location, contact details, bank accounts, and family members before users realize [1].

Fig. 1 and Fig. 2, downloaded from Google, illustrate privacy risks. Figure 1 shows an ID card revealing personal details like name, age, address, and birth date, potentially aiding cybercriminals. Fig. 2 depicts a license plate, which could lead to the owner's information being accessed through insurance companies and sold, resulting in frequent sales or fraud calls.

In recent years, photo privacy leaks have become a significant issue, often occurring without the victim's knowledge. A news release by Guangzhou Daily Ocean Network on May 13, 2024, reported that loan sales staff hired part-time workers to photograph license plates and phone numbers for credit assessments, violating car owners' privacy [2]. Once an ID card is leaked, it can quickly lead to fraud, usury, and other illegal activities [3]. Therefore, photo privacy detection, which identifies privacy risks like license plates and ID cards through image and video analysis, is crucial [1].

Figure 1. Example of leaked ID card

Figure 2. Example of leaked license plate

1.2. Research Motivation

Research on image privacy object recognition has gained attention recently, focusing mainly on CNN and enhanced deep learning models like OCNN for ID card and license plate recognition. These methods improve accuracy and robustness through multi-level feature extraction [1].

Recent advancements, particularly the Vision Transformer (ViT) model, have enhanced image recognition using self-attention and global feature extraction, especially in complex, multi-object scenarios [4]. Hybrid models combining CNN and ViT are being explored to further boost accuracy and efficiency, but haven't yet been applied to image privacy recognition, where CNN remains standard [5].

Despite progress, current models struggle with complex scenes, occlusions, and blurred images. Data scarcity also hinders model generalization and robustness. This article proposes a hybrid model combining OCNN, CNN, and ViT to address these issues by leveraging their strengths.

Specifically, I utilized OCNN for hierarchical feature recognition to process 3D data and extract spatial features, CNN for multi-level convolution to extract local features, and ViT to divide images into blocks and use self-attention for global feature extraction. Finally, these features are fused together

2. PROPOSED SOLUTION

2.1. Specific Structure

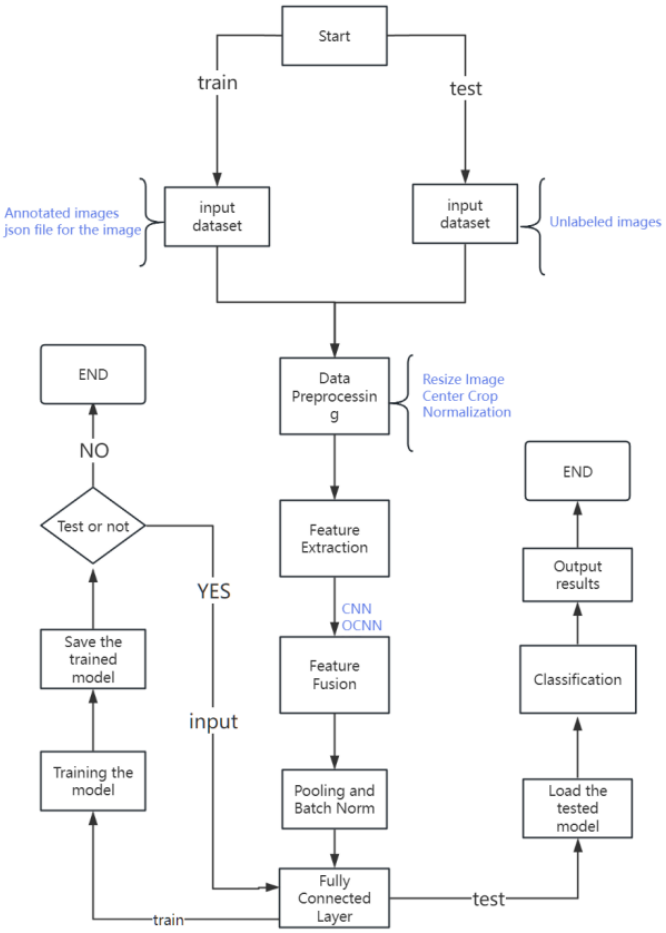

Figure 3. Model running process

2.2. Convolutional Feature Extraction

We aim to extract visual features from images using a pre-trained ResNet-50 model. This model reduces parameters and computational complexity while maintaining high accuracy [8].

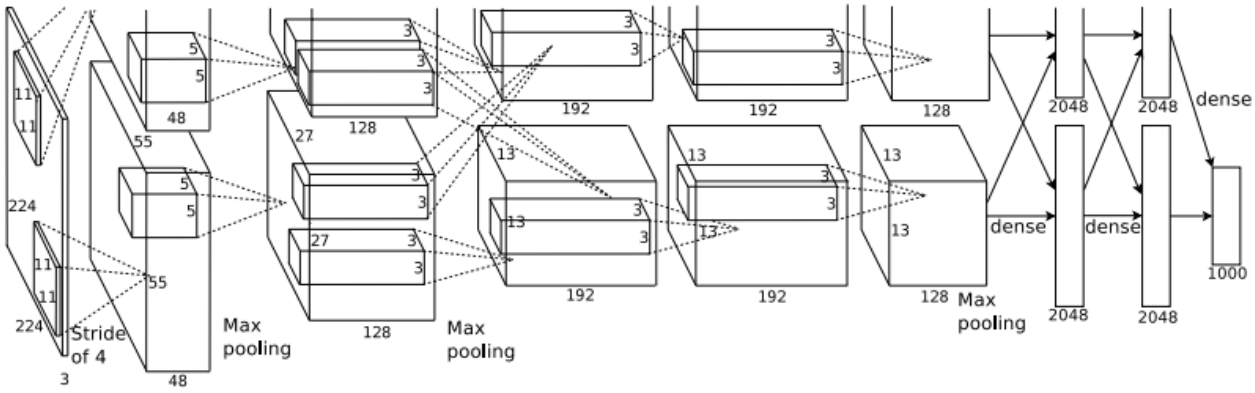

Figure 4. The architecture of our CNN

2.2.1. Data preprocessing

Before the image is input into the ResNet-50 model, a series of preprocessing steps are required for the image. Specifically, there are three steps: 1. Resize the image to 256×256 pixels. 2. Center cropping, cropping the center part of the image so that it is 224×224 pixels in size. 3.Normalization and other operations, use the mean and standard deviation of the ImageNet dataset to normalize the image, reduce the difference between different images, and make the model more stable for training and prediction [7].

Table 1. Architectures for ImageNet

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	[3×3, 64; 3×3, 64] ×2	[3×3, 64; 3×3, 64] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3
conv3_x	28×28	[3×3, 128; 3×3, 128] ×2	[3×3, 128; 3×3, 128] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×8
conv4_x	14×14	[3×3, 256; 3×3, 256] ×2	[3×3, 256; 3×3, 256] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×23	[1×1, 256; 3×3, 256; 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512; 3×3, 512] ×2	[3×3, 512; 3×3, 512] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

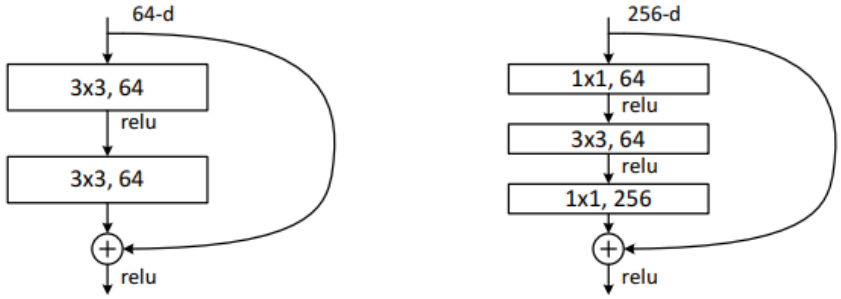

Figure 5. A deeper residual function F for ImageNet. Left: a building block (on 56×56 feature maps) as in Fig. 3 for ResNet34. Right: a "bottleneck" building block for ResNet-50/101/152.

2.2.2. Feature extraction process

We use the pre-trained ResNet-50 to extract low and mid-level visual features. The model consists of multiple residual blocks, each containing three layers of convolution: 1x1, 3x3, and 1x1. Outputs from these layers are added to the input through skip connections, forming residual units [8].

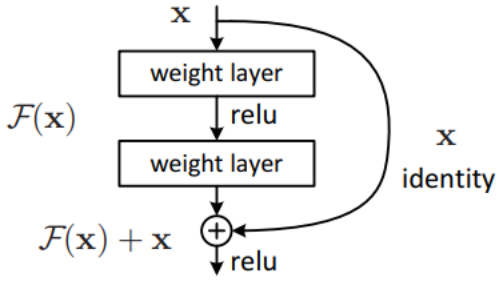

Figure 6. Residual learning: a building block

2.2.3. Pooling layers

Use a pooling layer to perform dimensionality reduction and feature selection on the output of the convolutional layer. We use the following formula to represent this pooling operation: $y_{pool} = max(y)$.

Among them, y_{pool} is the feature map after pooling, and we use the maximum pooling operation. The pooling operation reduces the size of the feature map by selecting the maximum value within a local area while retaining salient features [8].

2.2.4. Batch Normalization

Batch normalization reduces internal covariate shift, speeds up training, and improves model stability by standardizing feature maps [17]. We use μ to present the mean, σ to present the standard deviation, $\epsilon$to present a small constant used to prevent division by zero errors. The whole process can be expressed mathematically as follows: $BN(y) = \frac{y-\mu}{\sqrt{\sigma^2+\epsilon}}$.

2.2.5. Fully connected layers

Convolutional features are flattened and passed through fully connected layers for further feature extraction via linear transformation. The feature dimension after flattening is [batch size, 2048 * 7 * 7]. The output dimension of the fully connected layer after calculating based on the weights and biases is [batch size, 2048], [7-8, 17].

2.3. OCNN

OCNN is used for object feature extraction, focusing on identifying specific objects in images. It processes high-level semantic features and is pre-trained on large datasets to recognize objects like faces and vehicles, providing richer semantic information and enhancing feature representation for privacy risk monitoring tasks.

The steps for feature extraction using OCNN are similar to those in the convolutional feature extraction pipeline and are not detailed here [7].

2.4. Vision Transformer Feature Extraction

The Vision Transformer (ViT) model extracts high-level features by dividing the image into fixed-size blocks and processing them using the Transformer architecture. ViT captures global information and long-distance dependencies, suitable for tasks like image classification and object detection [12-13].

2.4.1. Data preprocessing

Data also needs to be preprocessed, including steps such as resizing the image, center cropping, and normalization [12-13].

2.4.2. Feature extraction process

(1) Image segmentation: First, the image is divided into fixed-size blocks (16x16 pixels), flattened into feature vectors to form the input sequence.

(2) Input Embedding: Map the features to an embedding space and add position encoding to preserve sequence information [12-13].

(3) Multi-head self-attention mechanism: Next, using the multi-head self-attention mechanism, ViT is able to capture the complex relationships between image patches.

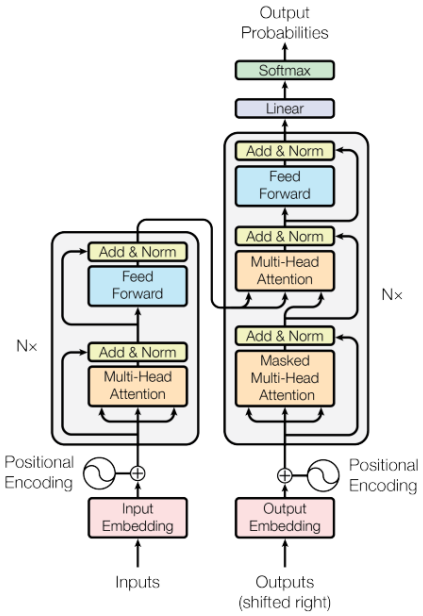

Figure 7. The Transformer - model architecture.

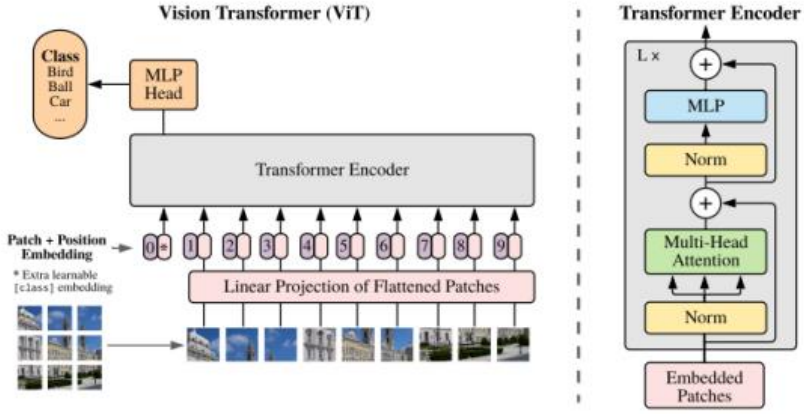

Figure 8. Model overview.

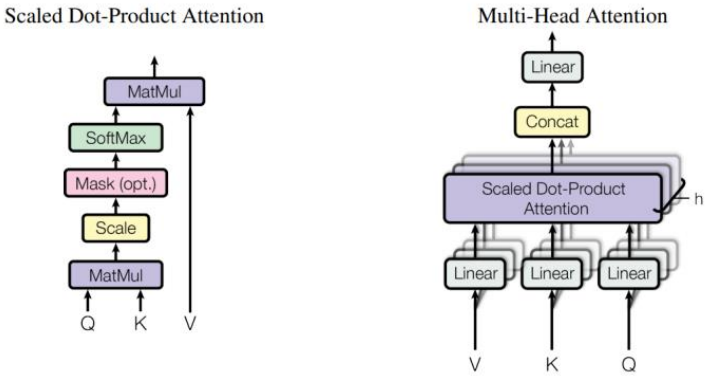

Figure 9. (left) Scaled Dot-Product Attention. (right) Multi-Head Attention consists of several attention layers running in parallel.

In Figure 9, Q is the query matrix, K is the key matrix, and V is the value matrix, d_k which is the dimension of the key vector. By calculating the dot product of the query, key, and value and applying

the softmax function, the model is able to assign a weight to each image patch, which can capture the relationship and dependency between different patches [12-13].

$$\text{Attention}(\text{Q}, \text{K}, \text{V}) = \text{softmax}(\frac{\text{QK}^{\text{T}}}{\sqrt{\text{d}_{\text{k}}}})\text{V} \quad (1)$$

(4) Feedforward Neural Network:

Each Transformer layer includes a feed-forward neural network for further feature extraction. It consists of two fully connected layers and a ReLU activation function:

$$FFN(x) = ReLU(W_1 x + b_1)W_2 + b_2 \quad (2)$$

Where W_1 and W_2 are weight matrices, and b_1 and b_2 are bias terms. Each layer's parameters differ, enhancing feature representation through nonlinear transformations [12-13].

(5) Layer Normalization and Residual Connections:

Layer normalization and residual connections are crucial for improving model stability and performance. Residual connections involve performing layer normalization first and then adding input features to features after sub-layer processing.

We use x and $SubLayer(x)$ to represent input features and features after sub-layer processing, such as multi-head attention or feedforward neural network. The formula is:

$$y = x + SubLayer(x) \quad (3)$$

The final output is a high-level feature with dimensions [batch size, 2048] [12-13].

2.5. Feature Fusion

We use $CNNFeature, OCNNFeature$ and $TransformerFeature$ to represent the features extracted from CNN, OCNN and Transformer respectively. Through feature fusion, feature representations at different levels are integrated to improve the quality and robustness of the overall feature expression.

$$FusedFeature = Concat(CNNFeature, OCNNFeature, TransformerFeature)$$

The dimension after feature concatenation will be [batch size, 3 * 2048], and then further processing will be performed [12].

2.6. Prediction Output Module

After feature extraction and fusion, the next step is to use a classifier to detect privacy risks in the image. The fully connected layer classifier, a supervised learning method, is suitable for complex tasks like image classification, object detection, speech recognition, and natural language processing. It handles large-scale, high-dimensional data and learns complex patterns [15].

2.6.1. Data Input

The fused features are used as input data, and the label (privacy risk or no privacy risk) is used as the target output. The feature dimension of the input data is [batch size, 3 * 2048].

2.6.2. Training Model

When training the model, the fully connected layer mainly updates its weights and bias parameters through the back-propagation algorithm and gradient descent optimization method.

(1) Forward Propagation

During forward propagation, data flows from the input layer to the output layer, passing through fully connected layers and non-linear activation functions.

We use x to represent input data and pass it to the fully connected layer. The linear transformation is calculated as $z = W \cdot x + b$, where W is the weight matrix and b is the bias vector. Then, perform a non-linear transformation on z to obtain the activation value a, expressed as $a = \sigma(z)$, where σ is the activation function (e.g., ReLU, Sigmoid). Finally, repeat until the data reaches the output layer. In the output layer, the Softmax activation function converts the output value into a probability distribution: $y = Softmax(z) = \frac{e^{z_i}}{\sum_j e^{z_j}}$.

(2) Compute Loss

Obtaining the output of the model through forward propagation$y$, we use the loss function to calculate the error between the predicted value and the true value to guide the update of the model parameters. For classification tasks, the loss function we use is Cross-Entropy Loss, which performs well in multi-classification tasks.

$$L = -\sum_i t_i log\ (y_i) \tag{4}$$

Where t is the one-hot encoding of the true label and y is the predicted probability of the model.

(3) Backpropagation and gradient updates

The back-propagation algorithm is used to calculate the gradient of the loss function with respect to each parameter (weight and bias), and these gradients are used to update the model parameters through the optimizer.

We used the Adam optimizer, which is an adaptive learning rate optimization method based on first- and second-order moment estimates.

Parameters set: learning rate: 0.0001, exponential decay rate of first-order moment β_1: 0.9. Exponential decay rate of ε second-order moment β_2: 0.999, decimal value to prevent division by zero error: 10^{-8} [21].

(4) Repeat Iteration

The above steps will be repeated in multiple iterations until the loss function of the model on the training set converges to a lower value or reaches the preset number of training times. Due to equipment limitations, the iteration value used in our training is only 30 [19-20].

2.6.3. Prediction and Classification

hybrid model will output the classification results to determine whether the image has privacy risks after predicting the new image features. The output layer will give the predicted probability of each category, and select the category with the largest probability and higher than 0.8 as the final classification result. At the same time, if the probability is lower than 0.8, it will be judged as no privacy leakage risk. (The manually set threshold is 0.8)

3. EXPERIMENTAL SETUP

In order to verify the superiority of the hybrid model (hierarchical feature recognition and Vision Transformer) in private item recognition, we designed and conducted the following experiments.

3.1. Dataset Preparation

Due to the particularity and difficulty of obtaining private data, this experiment uses a self-made dataset. The dataset includes:

3.1.1. Training set

Pictures containing private items (Chinese ID cards and blue license plates): 100 pictures in total, including pictures of multiple cars and pictures from multiple perspectives as complex scene pictures.

A json file that annotates images containing private items.

Images containing non-private items (e.g. business cards, cards, images of holding these items): 50 in total.

3.1.2. Test set

Mix private objects and secure images, including images from various angles, occlusions, and blurs to simulate the complexity of real scenes.

3.1.3. Experimental model

Hierarchical feature recognition model: Based on the model proposed by **[1]**, it is used to extract image features at multiple levels. However, since I failed to obtain the author's code and experimental results, I reproduced his code.

ViT model: I reproduced the ViT-based model.

Hybrid model (CNN + OCNN + ViT): The ViT model is integrated on the basis of the hierarchical feature recognition model to enhance the global feature extraction capability.

3.2. Experimental Procedures

3.2.1. Data preprocessing

All images were uniformly resized and standardized. At the same time, data enhancement was performed on images containing private items.

3.2.2. Model training

The above three models were trained on the same dataset, and the Adam optimizer and cross entropy loss function were used. However, due to the lack of professional experimental equipment, the number of training iterations was only 30, and the learning rate was 0.0001.

3.2.3. Model testing

The three models were tested using a test set that mixed private items and secure images, and their recognition accuracy was finally recorded.

3.3. Experimental Results

Table 2. Experimental Results

Model	Total images	ID Card accuracy:	License Plate accuracy	Overall accuracy
Hierarchical feature recognition model (CNN+OCNN)	222	43.29 %	64.74%	56.23 %
ViT Model	222	57.06%	71.31%	65.66 %
Hybrid Model	222	79 .17%	81.76%	80.73 %

3.3.1. Hierarchical feature recognition model

The recognition accuracy is low, especially when there is a lack of data sets and the ID card being tested is similar to other cards. The model also performs poorly in recognizing license plates when the ID card is partially occluded. In complex scenarios, such as when there are a large number of cars in the scene or when the car is at a tricky angle, the model does not recognize the license plate well.

3.3.2. ViT model

The accuracy has been improved and it can better capture the global features of the image, but it has shortcomings in processing details and local features.

3.3.3. Hybrid model

The accuracy is significantly improved compared to the hierarchical feature recognition model in the reference paper, and is also greatly improved compared to the popular ViT model in recent years [1]. It still remains stable even when the training set is insufficient. It can perform well in complex scenes (multiple cars, as shown in Figure 10, and multi-view pictures) and long-distance dependent feature extraction, and is also better than the hierarchical feature recognition model in simple scenes. This proves that the hybrid model is effective in Advantages in global feature integration.

Figure 10. Complex scene

3.4. Loss Analysis

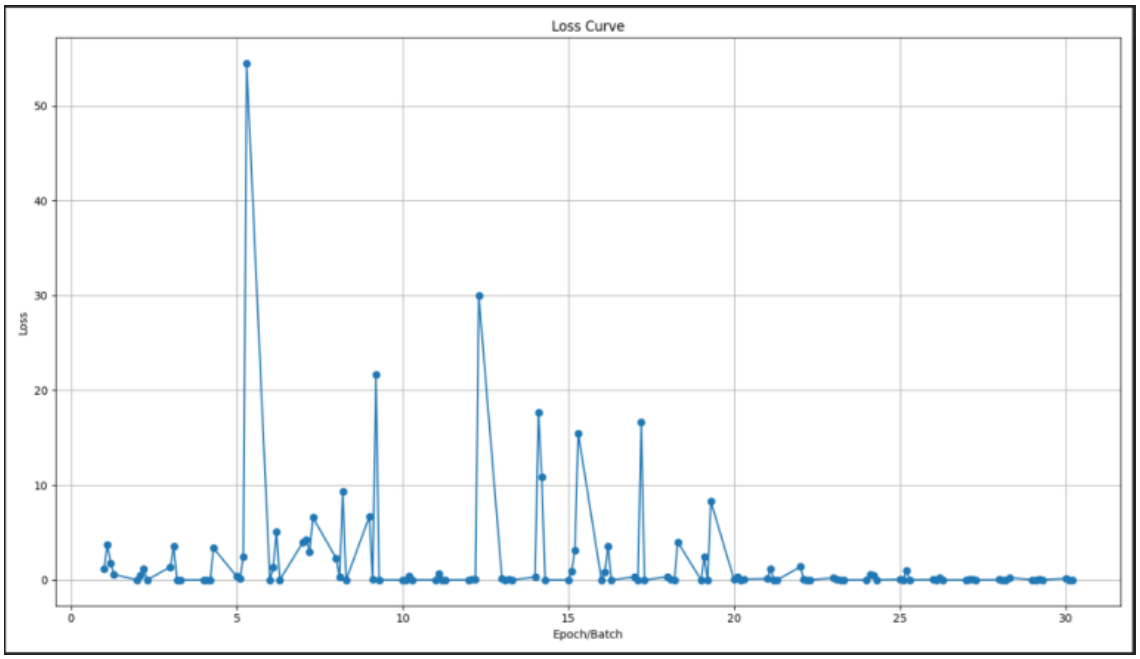

Figure 11. loss-epoch

In the hybrid model experiment, the iteration is 30, the batch size is 4, and the learning rate is 0.0001. The loss changes recorded during the training process are shown in Figure 8.

3.4.1. Initial and Mid-term Training Phase (Epoch 1-20)

During the first 20 epochs, the loss value shows significant fluctuations, indicating the model's poor adaptability to the training data. The fluctuations suggest that the model struggles with complex scenes and may not effectively learn from the data. Peaks around epochs 5, 10, and 15 highlight instability and difficulty in feature extraction during these phases.

3.4.2. Later training stage: Epoch 21-30

At this stage, despite the fluctuations, the loss value is significantly reduced overall, especially stable in the later training stage. In most batches, the loss value approaches 0, indicating that the model is gradually converging and the learning effect is significantly improved, the learning effect on training data is better.

Since the training sets are the same, the loss changes of the three models are similar, so we will not go into details here.

Through this experiment, the superiority of the hybrid model (CNN + OCNN + ViT) in the private item recognition task was verified. Despite the small training set and issues with data quality, the hybrid model still demonstrated higher recognition accuracy and broad applicability. This result shows that the hierarchical feature recognition model combined with ViT has great potential in handling complex image recognition tasks.

4. CONCLUSION

This paper proposes a hybrid model that combines hierarchical feature recognition (based on OCNN and CNN) and Vision Transformer for the recognition of private items (such as ID cards and license plates) in pictures. This paper proves the superiority of the hybrid model in terms of recognition accuracy and applicability through theoretical analysis and experimental verification. Specifically, the hybrid model significantly improves the recognition accuracy of private items in complex scenes by combining the local feature extraction capabilities of OCNN and CNN with the global feature processing capabilities of ViT.

The experimental results show that the hybrid model has a higher recognition accuracy than the separate hierarchical feature recognition model and ViT model on the same training and test sets. This result shows that the hybrid model has stronger adaptability and accuracy in dealing with a variety of complex scenarios and private item recognition tasks.

In the future, we can try to expand the data set to include more types of private items and data from different shooting angles to further verify the generalization ability of the model. However, due to the particularity and difficulty of obtaining private data, experiments should be conducted by official agencies such as the government. At the same time, we can try to apply the hybrid model to actual scenarios, such as WeChat Moments picture posting detection, TIKTOK video posting detection, etc., to test its performance in real environments. Finally, we can also consider adding a privacy protection mechanism during the model application process to ensure privacy security during data use and further enhance the social application value of the model.

REFERENCES

[1] Tran L, Kong D, Jin H, et al. Privacy-cnh: A framework to detect photo privacy with convolutional neural network using hierarchical features[C]//Proceedings of the AAAI conference on artificial intelligence. 2016, 30(1).

[2] He wen si. (2024, May 13). “Take the Front Photo + Move the Car Phone”, 5 Cents a! What Kind of Business Is This? Guangzhou Daily ocean net. https://news.dayoo.com/society/202405/13/140000_54667546.htm

[3] Ma jing zhen. (2022, April 7). “Take the Front Photo + Move the Car Phone”, 5 Cents a! What Kind of Business Is This? Xiao County People’s Government. https://www.ahxx.gov.cn/grassroots/259/156495891.html

[4] Hemalakshmi, G. R., Murugappan, M., Sikkandar, M. Y., Begum, S. S., & Prakash, N. B. (2024). Automated retinal disease classification using hybrid transformer model (SViT) using optical coherence tomography images. Neural Computing and Applications, 1-18.

[5] Barhoumi, Y., & Rasool, G. (2021). Scopeformer: n-CNN-ViT hybrid model for intracranial hemorrhage classification. arXiv preprint arXiv:2107.04575.

[6] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

[7] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25.

[8] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[9] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.

[10] Kohonen, T. (1990). The self-organizing map. Proceedings of the IEEE, 78(9), 1464-1480.

[11] Cottrell, M., & de Bodt, E. (1996, April). A Kohonen map representation to avoid misleading interpretations. In ESANN (Vol. 96, pp. 103-110).

[12] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

[13] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

[14] Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2018). Multimodal machine learning: A survey and taxonomy. IEEE transactions on pattern analysis and machine intelligence, 41(2), 423-443.

[15] Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020, November). A simple framework for contrastive learning of visual representations. In International conference on machine learning (pp. 1597-1607). PMLR.

[16] Le, Q., & Mikolov, T. (2014, June). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196). PMLR.

[17] Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning (pp. 448-456). pmlr.

[18] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. nature, 323(6088), 533-536.

[19] Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint arXiv:1412.7062.

[20] Yin, L., & Wang, Z. (2024). Bi-level binary coded fully connected classifier based on residual network 50 with bottom and deep level features for bearing fault diagnosis. Engineering Applications of Artificial Intelligence, 133, 108342.

[21] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.