A Comparison for Patch-level Classification of Deep Learning Methods on Transparent Images: from Convolutional Neural Networks to Visual Transformers

Hechen Yang[a], Chen Li[a,*], Peng Zhao[a], Ao Chen[a], Xin Zhao[a] and Marcin Grzegorzek[b]

[a] *Microscopic Image and Medical Image Analysis Group, MBIE College, Northeastern University, 110169, Shenyang, PR China*
[b] *Institute of Medical Informatics, University of Luebeck, Luebeck, Germany*

ARTICLE INFO

Keywords:
patch level
image classification
Transparent Images
deep learning
Convolutional Neural Network
visual transformer

ABSTRACT

Nowadays, analysis of transparent images in the field of computer vision has gradually become a hot spot. In this paper, we compare the classification performance of different deep learning for the problem that transparent images are difficult to analyze. We crop the transparent images into 8×8 and 224×224 pixels patches in the same proportion, and then divide the two different pixels patches into foreground and background according to groundtruch. We also use 4 types of convolutional neural networks and a novel ViT network model to compare the foreground and background classification experiments. We conclude that ViT performs the worst in classifying 8×8 pixels patches, but it outperforms most convolutional neural networks in classifying 224×224.

1. Introduction

With the advent of the era of science and technology, the application of transparent images has become more and more widely used in various fields around humans, such as the segmentation of renal transparent cancer cell nuclei in medicine [1]. The shape and location information of the cell nucleus is of great significance for the classification and diagnosis of benign and malignant renal cancer. Another example is to identify the number of transparent microorganisms in environment, so as to judge the degree of environmental pollution [2]. In recent years, the detection of transparent objects in images is also a hot spot in vision research. It is not an easy task to detect whether there are transparent objects or translucent objects in images [3]. Because the transparent target area to be observed is generally very small or very thin, the colors and contrast of foreground and background are similar, and only the residual edge part leads to low resolution of foreground or background, which largely depends on its background and lighting conditions. Therefore, there is an urgent need for some effective methods to identify transparent or translucent images.

In recent years, computer vision has good performance in computer vision acquisition [4], contour tracking [5], edge detection [6], face recognition [7], fingerprint recognition [8], automatic driving [9] and medical image analysis [10]. We considering the excellent performance of computer vision in image analysis, such as high speed, high accuracy, low consumption, high degree of quantification, strong objectivity [11], therefore computer vision can make up the shortcomings of traditional morphological methods. It brings new opportunities to transparent image analysis. Especially, when an image is transparent and short of visual information, we usually need to crop it into patches to discover more visual details to recover the lost information. Hence, research work on patch-level is significant for transparent image analysis, such as patch-level image segmentation and classification tasks.

In recent years, deep learning is the most efficient method in the field of machine vision, such as the popular *Convolutional Neural Network* (CNN) Xception [12], VGG-16 [13], Resnet50 [14], Inception-V3 [15], MobileNet [16], NasNet [17], and novel *Visual Transformers* (VTs) [18]. CNNs slowly expand the receptive field until it covers the whole image by accumulating convolution layers, so CNNs complete the extraction of graphics from local to global information. In contrast, transformers can obtain global information from the beginning, so they are more difficult to learn, but their ability to learn long-term dependence is stronger [18]. Hence, CNNs and Transformers have advantages and disadvantages in dealing with visual information. Therefore, this paper compares the patch-level classification performance of transparent images with different CNN and VT methods, where it aims to discover the adaptability of different deep learning models on this research domain.

This paper uses EMDS5 as an example of transparent images. First, the transparent images are divided into training, validation, and test sets according to a ratio of 2:2:4. The workflow of patch-level image classification is shown in Fig. 1, where (a) is the training set, including original images and ground truth (GT) images with multi-scale settings. (b) is the training process of deep learning models where several typical deep learning methods are selected and trained. (c) is the test set. (d) is the patch-level classification prediction result.

The structure of this paper is as follows: In Section 2, related work about deep learning in the classification of transparent images is introduced. In Section 3, comparative experiments about transparent images classification on multi-scale patches with deep learning methods are carried out. In Section 4, the conclusion and future work about transparent

[*] Corresponding author
lichen201096@hotmail.com (C. Li)

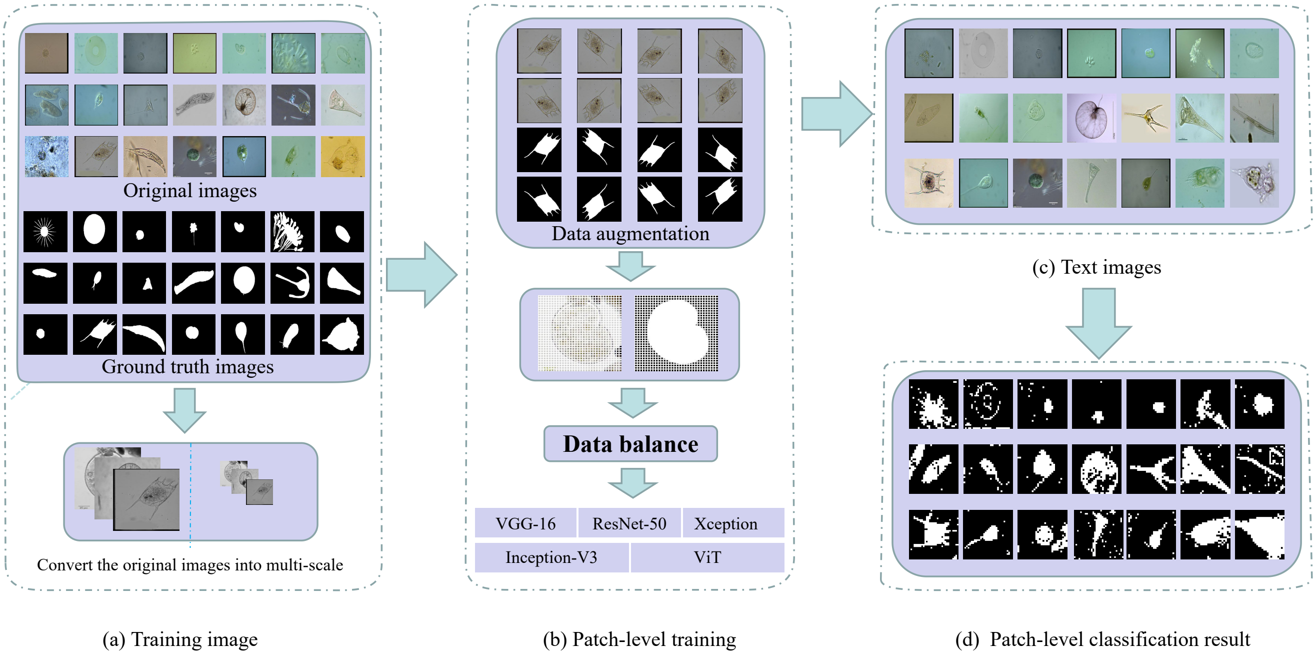


Figure 1: Workflow of patch-level classification in transparent images (using environmental microorganism EMDS-5 images as examples).

images are summarized in patch-level classification of deep learning methods results, and future development of deep learning methods for analyzing transparent images.

2. Related Work

This section introduces some common analysis methods, application scenarios and research purposes of transparent images. The advantages and disadvantages of some popular deep learning methods are also discussed.

2.1. Introduction to Transparent Image Analysis

Object analysis is one of the important branches in the field of robot vision, especially the analysis of transparent images of objects (transparent images) is challenging [19]. In traditional machine analysis methods, the flexibility of transparent image features obtained by integrating multi-class algorithms is poor, and the analysis performance is difficult to improve. For example, home robots can't see things at all when they are detecting some transparent glassware. The ClearGrasp machine learning algorithm performs well in analysing transparent objects [20]. It can estimate high-precision data of transparent objects from RGB-D transparent images, thereby improving the accuracy of detecting transparent objects.

As an important technical means for analysing objects, photoelectric sensors are widely used in the fields of industrial automation, mechanization and intelligence. It uses the properties of light to detect the position and change of the object, but when detecting transparent color objects, the light beam of the traditional diffuse reflection photoelectric sensor penetrate the transparent material, causing the sensor to fail. Diffuse reflection photoelectric sensor adopts a phase-locked loop narrowband filter frequency selection technology, which improves the sensitivity to self-returning light and stability of detecting transparent objects [21].

There are many transparent objects in the industrial field. Such as transparent plastics, transparent colloids, and liquid drops. These transparent objects bring a lot of uncertainty to products. If factories want to have high-quality products, sometimes it is very important to analysis these transparent objects and control shapes of the transparent objects. However, it is a difficult problem to segmentation the shape of transparent objects through morphological methods. For instance, Hata et al. used a genetic algorithm to segmentation the transparent paste drop shape in the industry and obtained good performance [22].

The segmentation of transparent objects is very useful in computer vision applications. However, the foreground of a transparent image is usually similar to its background environment, which leads to the general image segmentation methods in dealing with transparent images in general. The light field image segmentation method can accurately and automatically segment transparent images with a small depth of field difference and improve the accuracy of the segmentation and it has a small amount of calculation [23]. Hence, it is widely used in the segmentation of transparent images.

The correct segmentation of zebrafish in biology has greatly promoted the development of life sciences. However, the zebrafish's transparency makes the edges blurred in the segmentation. Mean shift algorithm can enhance the color representation in the image and improve the discrimination of the specimen against the background [24]. This method improves the efficiency and accuracy of zebrafish specimen segmentation.

Visual object classification is very important for robotics and computer vision applications. Commonly used statistical classification methods such as bag-of-features [25] are often applied to image classification. The principle is to extract local features of the image for classification. However, these methods cannot be applied to the classification of transparent images, because transparent images largely depend on the background. Foreground transparent objects do not have their own complete characteristics, and it is difficult to accurately classify them. The more popular method is the light field distortion feature [26], which can describe transparent objects without knowing the texture of the scene, thus improving the accuracy of classifying transparent images.

2.2. Deep Learning

Simonyan et al. propose the VGG series of deep learning network models (VGG-Net), of which VGG-16 is the most representative [27]. VGG-Net can imitate a larger receptive field by using multiple 3×3 filters, which enhances nonlinear mapping, reduces parameters and improves the network to be more judgmental. Meanwhile, VGG-16 continue to deepen the depth of the previous VGG-Net, with 13 convolutional layers and 3 fully connected layers. With the continuous increase of convolution kernel and convolution layer, the nonlinear ability of the model is stronger. VGG-16 can better learn the features in images and achieve good performance in the analysis of images classification, segmentation and detection. Simonyan proves that as the depth of the network increases, it promotes the accuracy of image analysis [27]. But this increase in depth is not without limit. Excessively increasing the depth of the network will lead to network degradation problems. Therefore, the optimal network depth of VGG-Net is set to 16-19 layers. Moreover, VGG-16 has three fully connected layers, which causes more memory to be occupied, too long training time and difficulty in tuning parameters.

He et al. propose the ResNet series of networks and add a residual structure in networks to solve the problem of network degradation [28]. The ResNet model introduces a jumpy connection method "shortcut connection". This connection method allows the residual structure to skip some levels that have not be fully trained in the feature extraction process, and increases the model's utilization of feature information during the training process. As the most classical model in the ResNet series, ResNet50 has a 50-layer network structure. This model adopts the highway network structure, which makes the network have strong expression capabilities and the ability to acquire more advanced features. Therefore, it is widely used in the field of image analysis. However, the network model is too deep and complicate, so how to judge which layers in the deep network have not be fully trained, and then optimize the network is a difficult problem.

Szegedy et al. propose the GoogLeNet network model, which has the advantage of reducing the complexity of the network on the basis of ResNet. They first proposed Inception-v1, whose network is 22 layers deep and consists of multiple Inception structures cascade as basic modules. Each Inception module consists of a 1×1, 3×3, 5×5 convolution kernel and a 3×3 maximum pooling, which is similar to the idea of multi-scale and increases the adaptability of the network to different scales [29]. With the continuous improvement of the inception module, the inception-v2 network uses two 3×3 convolutions instead of 5×5 convolutions and increases the BN method, which reduces the amount of calculation and speeds up the training time [30]. The Inception-v3 network introduces the idea of decomposing convolution, splitting a larger two-dimensional convolution into two smaller one-dimensional convolutions, further reducing the amount of calculation [31]. At the same time, Inception-v3 optimizes the Inception module, embeds the branch in the branch and improves the accuracy of the model.

Xception is another improvement after Inception-v3 [32]. It mainly uses depthwise separable convolution to replace the convolution operation in Inception-v3. The Xception model uses deep separable convolution to increase the width of the network, which not only improves the accuracy of classification but also improves the network's ability to learn subtle features. Meanwhile, Xception adds a residual mechanism similar to ResNet to significantly improve the speed of convergence during training and the accuracy of the model. However, Xception is relatively fragmented in the calculation process, which results in a slower iteration speed during training.

Transformer is a deep neural network based on the self-attention mechanism, which enables the model to be trained in parallel and can obtain the global information of the training data. Due to its computational efficiency and scalability, it is widely used in the field of Natural Language Processing. Recently, Dosovitskiy et al. proposed the Vision Transformer (ViT) model and found that it performs very well on image classification tasks [33]. In the first step of training, the ViT model divides pictures into fixed-size image patches and uses its linear sequence as the input of the transformer model. In the second step, position embeddings are added to the embeddings patches to retain the position information, and then the image features are extracted through the multi-head attention mechanism. Finally, the classification model is trained. ViT breaks through the limitation that RNNs model cannot be calculated in parallel and self-attention can produce a more interpretable model. ViT can be suitable for solving image processing tasks, but experiments have proved that large data samples are needed to improve the training effect.

2.3. Summary

Transparent image analysis is used in various fields, but the foreground and background of transparent images are too similar to make analysis difficult. Compared with deep learning methods, the general traditional analysis methods

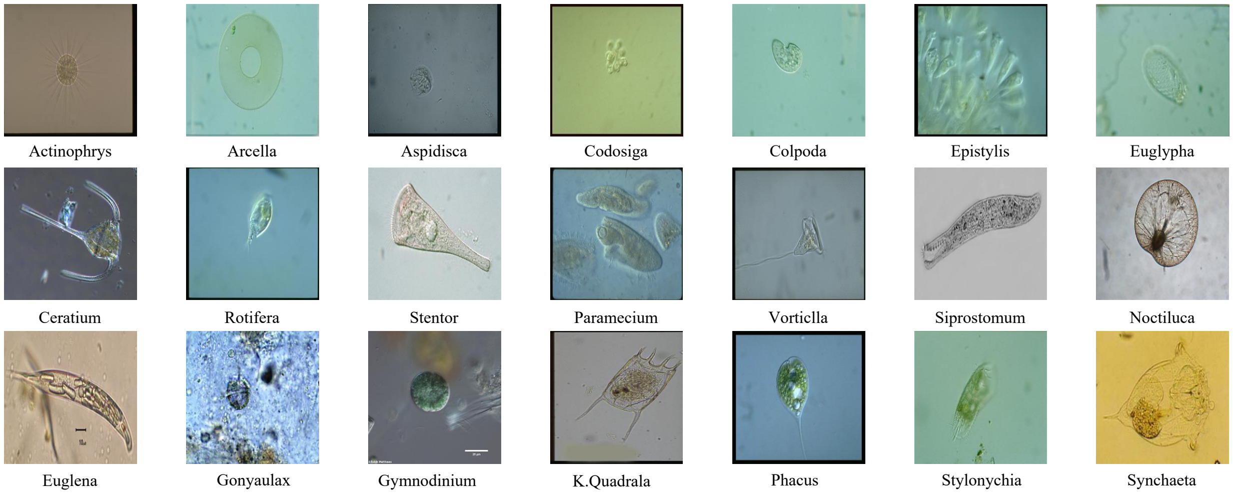


Figure 2: Environmental microorganism EMDS5 images.

are time-consuming, labor-intensive and costly. So this paper compares the performance of several classical deep learning networks for transparent image analysis.

3. Comparative Experiment

This section introduces the patch-level classification experiment process and classification results of transparent images under several deep learning networks.

3.1. Experiment Setting

3.1.1. Data Settings

In our work, we use Environmental Microorganism Data Set Fifth Version (EMDS-5) as transparent images for analysis [2]. It is a newly released version of the EMDS series, which contains 21 types of EMs, each of which contains 20 original microscopic images and their corresponding ground truth (GT) images (examples are shown in Fig.2 and Fig. 3). We randomly divide each category of EMDS-5 into training, validation, and test data sets at a ratio of 1:1:2. Therefore, we have 105 original images and their corresponding GT images for training and validation respectively, and 210 original images for testing as shown in Tab 1.

Table 1
EMDS-5 Experimental data.

	Training Set	Validation Set	Test Set
Actinophrys	5	5	10
Arcella	5	5	10
Aspidisca	5	5	10
Codosiga	5	5	10
Colpoda	5	5	10
Epistylis	5	5	10
Euglypha	5	5	10
Paramecium	5	5	10
Rotifera	5	5	10
Vorticlla	5	5	10
Noctiluca	5	5	10
Ceratium	5	5	10
Stentor	5	5	10
Siprostomum	5	5	10
K.Quadrala	5	5	10
Euglena	5	5	10
Gymnodinium	5	5	10
Gonyaulax	5	5	10
Phacus	5	5	10
Stylonychia	5	5	10
Synchaeta	5	5	10
total	105	105	210

3.1.2. Data Preprocessing

In the first step, we uniformly convert all images sizes to 224×224 pixels and 7168×7168 pixels to keep that each image is cropped into the same number of multi-scale patches. In the second step, we gray-scale EMDS-5 images to facilitate the calculation of gradients and feature extraction during training. In the third step, the training and validation images, and their corresponding GT images are cropped into patches (8×8 pixels and 224×224 pixels), where 105×1024=107520 patches are obtained. We divide these small patches into two categories according to the corresponding GT image small patches: foreground and background. The classification basis is that the target area is greater than 50%, which means there is foreground, otherwise it is background. In the fourth step, we find that the 224×224 pixels patches with foreground and background are 16630 and 90890, respectively. In order to avoid data imbalance during training, we rotate the training set image small patches by 0, 90, 180, 270 degrees and mirror them for data augmentation. Then we further obtain 16630×8=133040 patches, from which 90890 patches are randomly selected as the target patches in the training set. We expand the data of the 8×8 pixels patches according to the same process. An example of the augment data is shown in Tab 2.

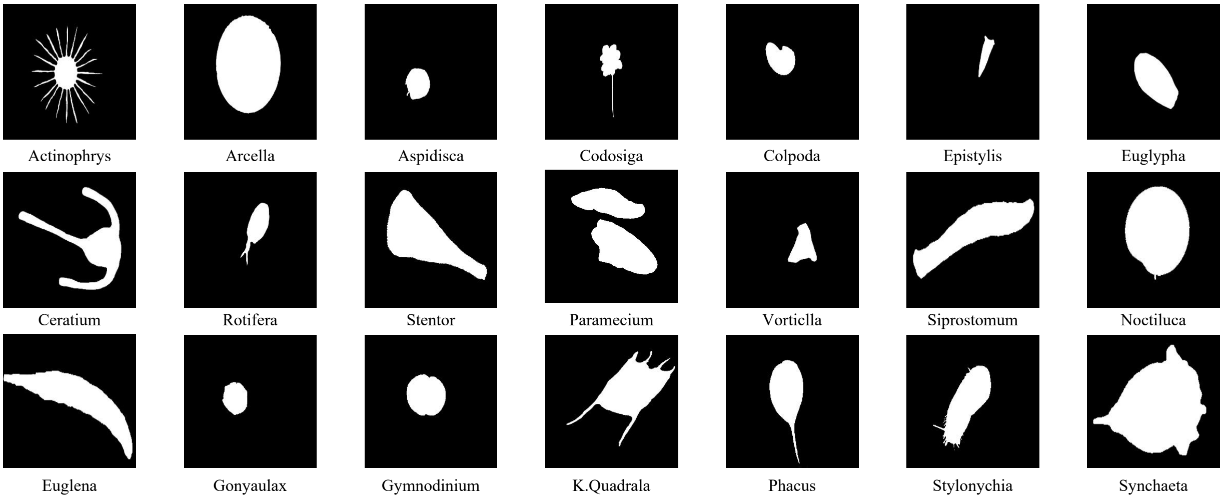


Figure 3: Environmental microorganism EMDS5-GT images.

Table 2
Data augmentation. FG (foreground) and BG (background)

Data Set	Training Set	Validation Set
8×8 pixels FG	16554	17356
8×8 pixels BG	90966	90164
Augmentation With FG	90966	\
8×8 Total	181932	107520
224×224 pixels FG	16630	17459
224×224 pixels BG	90890	90061
Augmentation With FG	90890	\
224×224 Total	181780	107520

3.1.3. *Experimental Environment*

Our classification comparison experiment is conducted on a local computer with Win10 Professional operating system, the computer runs 16 GB RAM i7-10700 CPU and 8 GB NVIDIA Quadro RTX 4000 GPU. The CNNs model we use in this paper is based on the Keras 2.3.1 framework using Tensorflow 2.0.0 as the backend; in the ViT model, we use the Pytorch 1.7.1 and Torchvision 8.0.2 operating environment.

3.1.4. *Hyper Parameters*

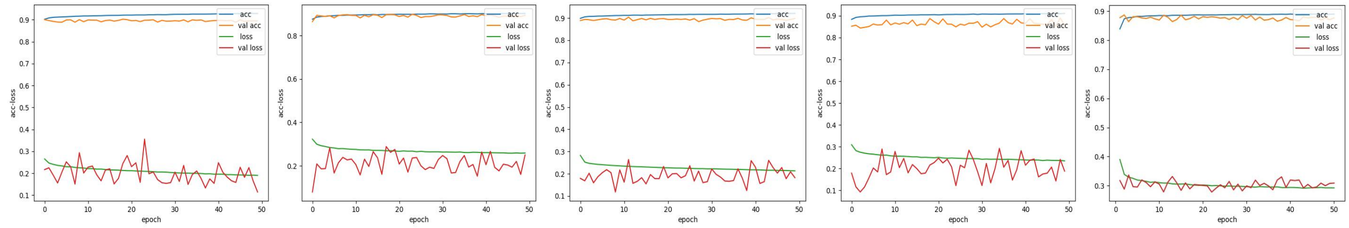
This experiment uses Adam optimizer, with 0.0002 learning rate and sets the batch size to 32 in our training process. In Fig. 4 and Fig. 5 we show the accuracy and loss curves of different deep learning models in this experiment. We find that the loss and accuracy curves of the training set are converging after training for 40 layers. Therefore, considering the computational performance of the workstation, we finally set 50 epochs for training.

Table 3
Evaluation metrics for images classification.

Assessments	Formula
Acc	$\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$
Pre (P)	$\frac{\text{TP}}{\text{TP+FP}}$
Rec (R)	$\frac{\text{TP}}{\text{TP+FN}}$
Spe	$\frac{\text{TN}}{\text{TN+FP}}$
F1	$2 \times \frac{P \times R}{P+R}$

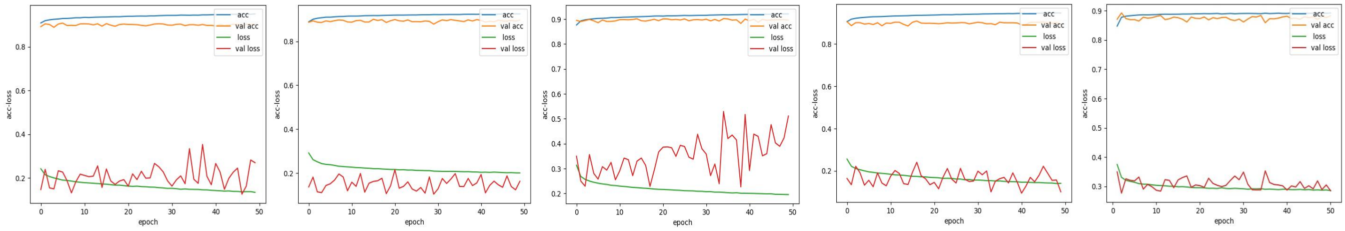
3.2. Evaluation Metrics

To compare the classification performance of different methods, we used the commonly used deep learning classification indicators Accuracy (Acc), Precision (Pre), Recall (Rec), Specificity (Spe), and F1-Score (F1) to evaluate the classification results. Acc reflects the ratio of correct classification samples to total samples. Pre reflects the proportion of correctly predict positive samples in the positive samples of model classification. Rec reflects the correct proportion of model classification in total positive samples Spe reflects the proportion of the model correctly classifying the negative samples in the total negative samples. F1 is a calculation result that comprehensively considers the Pre and Rec of the model. These evaluation indicators are defined in Tab 3. TP (True Positive), FN (False Negative), FP (False Positive), and TN (True Negative) are concept in the confusion matrix.



ResNet-50 Inception-V3 VGG-16 X-Inception ViT

Figure 4: Compare the results of the loss and accuracy curves of deep learning on the 8 × 8 pixels training and the validation sets.



ResNet-50 Inception-V3 VGG-16 X-Inception ViT

Figure 5: Compare the results of the loss and accuracy curves of deep learning on the 224 × 224 pixels training and the validation sets.

3.3. Comparative Experiment

3.3.1. Comparative Experiment of 8 × 8 Pixels Patches

Comparison on Training and Validation Sets: In order to compare the classification performance of CNNs and ViT models, we calculate Pre, Rec, Spe, F1, Acc, Max Acc. In Tab 4, we summarize the results of 8×8 pixels patches on validation set for each model. Overall, the Pre of the deep learning network in classifying the transparent image background is higher than the foreground. Besides, the ability of the five models to classify transparent images backgrounds is almost 97%, the highest is the VGG-16 value of 97.6%, and the lowest is the X-Inception and the ViT value of 96.7%. Meanwhile, the Pre rate of classification foreground VGG-16 is the best and the Pre rate is 63.1%. The Inception-V3 is the lowest 53.3%. For transparent images foreground classification, the highest Rec rate is the X-Inception value of 89.2%, and the lowest Vit value is 84.1%. For transparent images background classification, the highest Rec rate is the Vit value of 90.3% and the lowest is the X-Inception value of 85.0%. The Spe performance result of the classify background is opposite to the Rec performance result of the classify foreground. Among the five models, the highest Acc is ResNet50 with a value of 92.87%, and the lowest is ViT with a value of 89.26%.

Comparison on Test Set: In Tab 6 we summarize the results of these five network predictions. It can be seen that ResNet50 prediction Acc rate is the highest at 90.00%, X-Inception prediction Acc rate is the lowest at 85.85%. Furthermore , the lowest prediction Acc of the transparent foreground is the X-Inception value of 51.8%, and the highest is the ResNet50 value of 62.2%.

Table 4
A comparison of the classification results on validation set of 8 × 8 pixels patches. MAcc (Max Acc), FG (foreground) and BG (background) (In [%].)

Model	Class	Pre	Rec	Spe	F1	MAcc
ResNet50	FG	62.3	88.2	89.7	73.0	92.87
	BG	97.5	89.7	88.2	93.4	
Inception-V3	FG	61.8	88.6	89.5	72.8	90.24
	BG	97.6	89.5	88.6	93.4	
VGG-16	FG	63.1	88.6	90.0	73.7	92.09
	BG	97.6	90.0	88.6	93.6	
X-Inception	FG	53.3	89.2	85.0	66.7	91.10
	BG	96.7	85.0	89.2	90.9	
ViT	FG	62.4	84.1	90.3	71.6	89.26
	BG	96.7	90.3	84.1	93.4	

In order to more intuitively express the classification results of CNNs and ViT models for transparent image patches, we summarize the confusion matrices predicted by five models is shown in Fig. 7. We find that the ability of CNNs to classify foreground patches of transparent images is higher than that of ViT. Among them, the best CNNs model is Inception-V3, which correctly classified 29686 foreground patches, accounting for 91.50% of the total correct foreground patches. ViT correctly classified 27177 foreground patches, account-

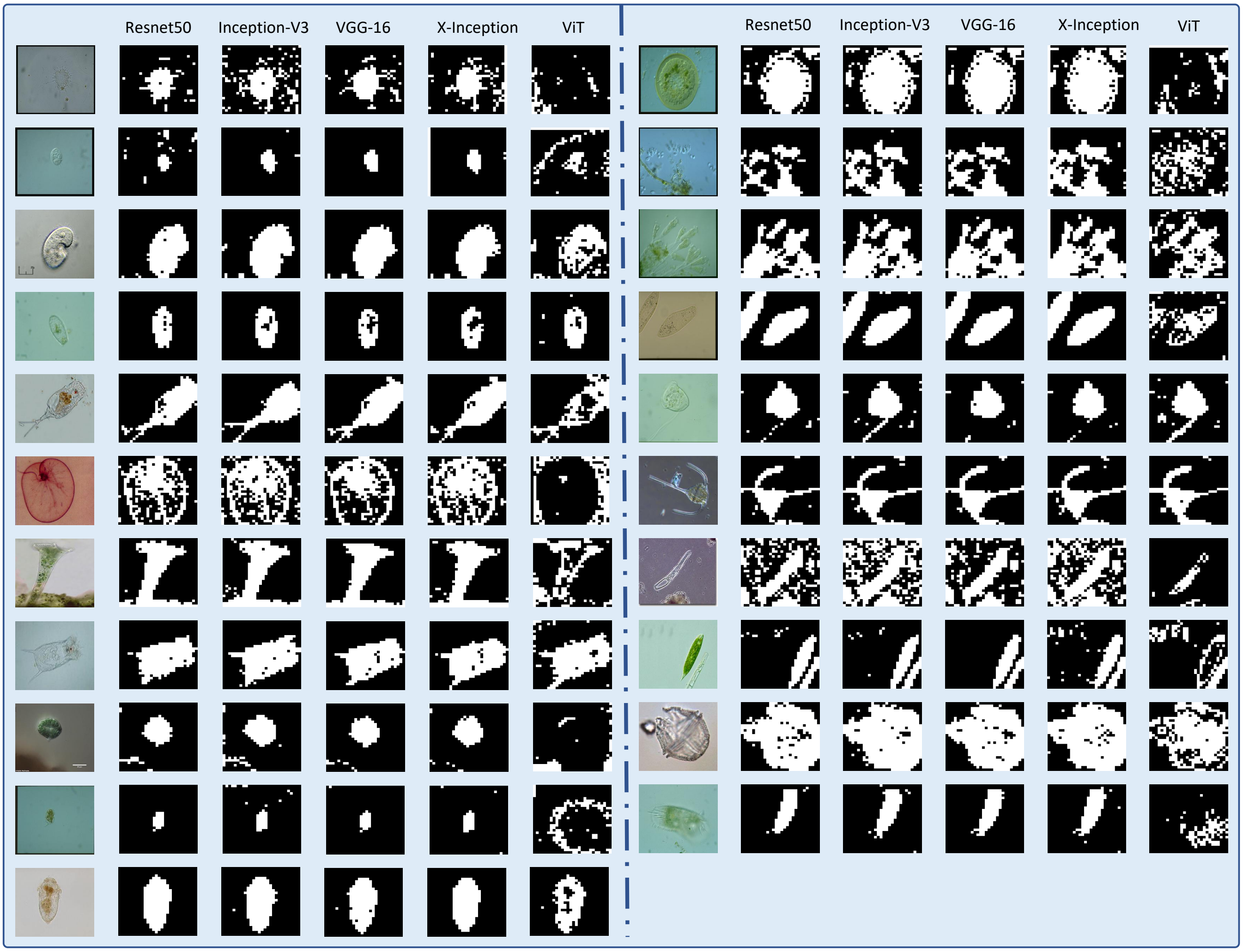


Figure 6: Reconstruction of 8 × 8 pixels transparent images classification results.

ing for 83.76% of the total correct foreground patches. In addition, the number of correctly classified backgrounds in ResNet50 is at most 165369, accounting for 90.57% of the total correct background patches, and the Pre of the classified background patches is 97.55%. Among the five models, ResNet50 has the highest prediction accuracy rate of 90.06% To better show the classification results, we reconstruct the transparent image after dicing in Fig. 6.

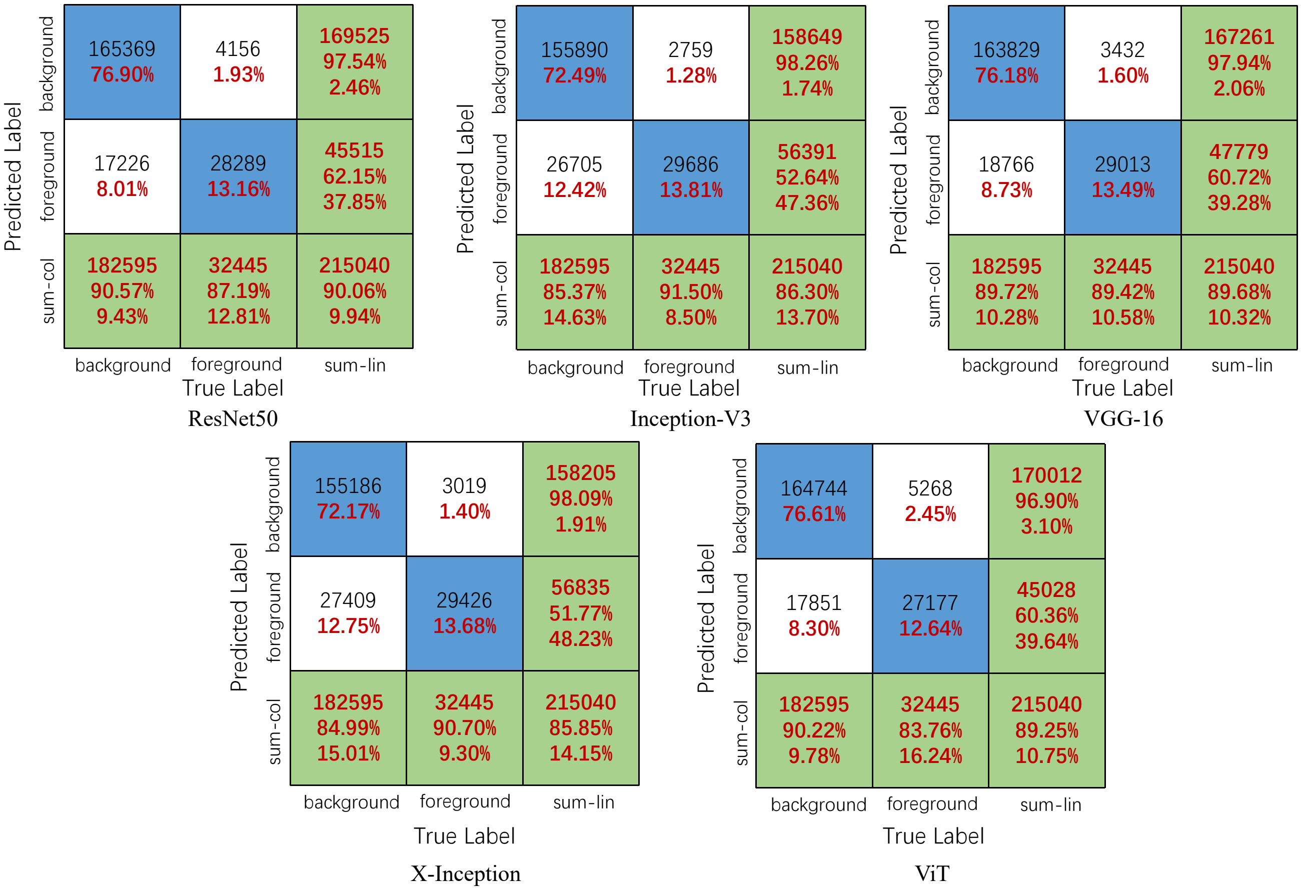
In Tab 5 we calculate the model training and prediction time and the size of the model during the experiment. From the perspective of model training time, the ViT model is much lower than CNNs models, where the ViT training time is 12418 seconds, and the X-Inception training time is the longest 45897 seconds. From the perspective of the size of the model, the minimum size of the ViT model is 31.2M, and the maximum size of the ResNet50 model is 114M. We calculate the time of the five prediction models. The fastest prediction time of VGG-16 is 757 seconds and the prediction time of a single picture is 0.0063 second. The slowest time of ViT is 1670 seconds and the prediction time of a single picture is 0.0078 second.

Table 5
A comparison of the classification results on train and test sets of 8 × 8 pixels patches. Train (Train times), Test (Test times) and Avg (Single picture prediction time)(In [s].)

model	Train	Test	Avg	Size(MB)
ResNet50	48762	1448	0.0067	114
Inception-V3	61443	1186	0.0055	107
VGG-16	49477	757	0.0035	62.2
X-inception	61247	999	0.0046	103
ViT	22133	1670	0.0078	31.2

3.3.2. *Comparative Experiment of* 224 × 224 *Pixels Patches*

Comparison on Training and Validation Sets: We compare the 224 × 224 pixels patches in the same way as in the



ResNet50

Predicted Label \ True Label	background	foreground	sum-lin
background	165369 76.90%	4156 1.93%	169525 97.54% 2.46%
foreground	17226 8.01%	28289 13.16%	45515 62.15% 37.85%
sum-col	182595 90.57% 9.43%	32445 87.19% 12.81%	215040 90.06% 9.94%

Inception-V3

Predicted Label \ True Label	background	foreground	sum-lin
background	155890 72.49%	2759 1.28%	158649 98.26% 1.74%
foreground	26705 12.42%	29686 13.81%	56391 52.64% 47.36%
sum-col	182595 85.37% 14.63%	32445 91.50% 8.50%	215040 86.30% 13.70%

VGG-16

Predicted Label \ True Label	background	foreground	sum-lin
background	163829 76.18%	3432 1.60%	167261 97.94% 2.06%
foreground	18766 8.73%	29013 13.49%	47779 60.72% 39.28%
sum-col	182595 89.72% 10.28%	32445 89.42% 10.58%	215040 89.68% 10.32%

X-Inception

Predicted Label \ True Label	background	foreground	sum-lin
background	155186 72.17%	3019 1.40%	158205 98.09% 1.91%
foreground	27409 12.75%	29426 13.68%	56835 51.77% 48.23%
sum-col	182595 84.99% 15.01%	32445 90.70% 9.30%	215040 85.85% 14.15%

ViT

Predicted Label \ True Label	background	foreground	sum-lin
background	164744 76.61%	5268 2.45%	170012 96.90% 3.10%
foreground	17851 8.30%	27177 12.64%	45028 60.36% 39.64%
sum-col	182595 90.22% 9.78%	32445 83.76% 16.24%	215040 89.25% 10.75%

Figure 7: Predict the confusion matrix on test set of 8×8 pixels patches

previous work in Section 3.1.3. In Tab 7, we summarize the classification results on validation set of 224 × 224 pixels patches for each network model. The highest Pre of classification foreground is ResNet50 value of 64.6%, the lowest is X-Inception value of 60.8%. However, the highest Pre of background classification is X-Inception value of 97.9%, and the lowest is ViT value of 96.0%. The highest Rec of the five model classification foreground is that the X- Inception value is 90.4%, and the lowest is the ViT value of 80.7%. The Spe results are just the opposite. Overall, the Rec and Spe performance of the five model classification backgrounds are almost 90%. When ResNet50 classifies the foreground and background of transparent images, F1- Score achieves the best result. Meanwhile, the Acc of ResNet50 model training is the highest at 94.99%.

Comparison on Test set: In Tab 8 we summarize the results of these five network predictions. It can be seen that the prediction Acc of X-Inception is 89.11% at the highest, and the prediction Acc of Inception-V3 is the lowest at 88.10%. However, the highest Pre in predicting transparent foreground is the ViT value of 60.6%.

Table 6
A comparison of the classification results on test set of 8×8 pixels patches. MAcc (Max Acc), FG (foreground) and BG (background)(In [%].)

Model	Class	Pre	Rec	Spe	F1	Max Acc
ResNet50	FG	62.2	87.2	90.6	73.0	90.0
	BG	97.5	90.6	87.2	93.4	
Inception-V3	FG	52.6	91.5	85.4	72.8	86.29
	BG	98.3	85.4	91.5	93.4	
VGG-16	FG	60.7	89.4	89.7	73.7	89.6
	BG	97.9	89.7	89.4	93.6	
X-Inception	FG	51.8	90.7	85.0	66.7	85.85
	BG	98.1	85.0	90.7	90.9	
ViT	FG	60.4	83.8	90.2	70.2	89.25
	BG	96.9	90.2	83.8	93.4	

In order to more intuitively express the classification results of CNNs and ViT models on transparent image patches, we summarize the confusion matrices predicted by five models and shown in Fig. 9. We find that the ability of CNNs to classify foreground patches of transparent images is higher than that of ViT. Among them, X-Inception is the best in the CNNs model, which correctly classifies 29559 small patches. ViT correctly classifies 26285 foreground patches. However, the highest accuracy of classifying transparent image

Table 7
A comparison of the classification results on validation set of 224×224 pixels patches. MAcc (Max Acc), FG (foreground) and BG (background)(In [%].)

Model	Class	Pre	Rec	Spe	F1	Max Acc
ResNet50	FG	64.6	87.6	90.7	74.4	94.99
	BG	97.4	90.7	87.6	93.9	
Inception-V3	FG	63.0	88.9	89.9	73.7	92.51
	BG	97.7	89.9	88.9	93.6	
VGG-16	FG	63.2	85.8	90.3	72.8	92.08
	BG	97.1	90.3	85.8	93.6	
X-Inception	FG	60.8	90.4	88.7	72.7	94.72
	BG	97.9	88.7	90.4	93.1	
ViT	FG	63.3	80.7	90.9	70.9	89.28
	BG	96.0	90.9	80.7	93.4	

Table 8
A comparison of the classification results on test set of 224×224 pixels patches. MAcc (Max Acc), FG (foreground) and BG (background)(In [%].)

Model	Class	Pre	Rec	Spe	F1	Max Acc
ResNet50	FG	59.0	88.7	89.0	73.0	88.92
	BG	97.8	89.0	88.7	93.4	
Inception-V3	FG	56.8	90.6	87.7	72.8	88.10
	BG	98.1	87.7	90.6	93.4	
VGG-16	FG	57.1	87.0	88.3	73.7	88.11
	BG	97.4	88.3	87.0	93.6	
X-Inception	FG	59.6	88.0	89.3	66.7	89.11
	BG	97.6	89.3	88.0	90.9	
ViT	FG	60.6	80.5	90.6	69.1	89.09
	BG	96.3	90.6	80.5	93.4	

Table 9
A comparison of the classification results on train and test sets of 224×224 pixels patches. Train (Train times), Test (Test times) and Avg (Single picture prediction time)(In [s].)

model	Train	Test	Avg	SIZE(MB)
ResNet50	51077	1634	0.0076	114
Inception-V3	66095	1296	0.0060	107
VGG-16	50908	1364	0.0063	62.2
X-inception	73465	1049	0.0049	103
ViT	23102	2156	0.0100	31.2

background is that the ViT value is 90.52%, which correctly classifies 165288 background images. Meanwhile, in order to better show the classification results, we reconstruct the transparent images after dicing in Fig. 8.

In Tab 9 we find that the training time of the ViT model is still the fastest at 12418 seconds, and the slowest is 73465 seconds for X-Inception. Besides the fastest prediction time of Inception-V3 is 1049 seconds and the prediction time of a single picture is 0.0060 second. The slowest time of ViT is 2156 seconds and the prediction time of a single picture is 0.0100 second.

3.4. In-depth Analysis

We compare classification results in Tab 4 and Tab 7, and find that when the transparent image patches of the training input increase from 8×8 to 224×224 pixels, the Pre of the classification foreground of the five models has improve. The biggest improvement is X-Ineption from 53.3% to 60.8%. This shows that when the image input size becomes larger, CNNs can extract more features, thereby improving the model's classification performance for transparent images. As the input image size increases, it has little effect on the performance of the five models to classify the transparent image background. Among them, the biggest improvement is X-Inception, which has an Acc increase of 1.2%. With the increase in the size of the input image of the CNNs, the training time of the five models has also increase by 2% to 6%, but the Acc of the model has also improve, and the training Acc of the ResNet50 model is the highest value of 94.99%. VGG-16 and ViT remain basically unchange.

We compare Tab 6 and Tab 8. When the transparent image is cropped into pathes of 8×8 pixels, the prediction Acc of the ResNet50 model is the highest value of 90%, and the lowest is the X-Inception value of 85.85%. However, when the transparent image is enlarged and cropped into patches of 224×224 pixels, the X-Inception prediction Acc rate is the highest 89.11%, and the second is that the ViT Acc rate is 89.09%. By increasing the size of the input transparent images patches, the prediction Acc of the ViT model exceeds that of ResNet50, Inception-V3 and VGG-16. Moreover, when the input image is a small patches of 224×224 pixels, the Pre of the ViT model to classify the foreground of the transparent image is higher than that of the CNNs network. This shows that the advantage of ViT for global information description is higher than that of some CNNs networks.

In the predicted 215040 patches, we compare the performance of five types of network classification foreground and background. In Fig. 7, we find that Inception-v3 has the largest number of correct foregrounds under 8×8 pixels patches. ResNet50 has the largest number of correctly classify backgrounds. In Fig. 9 we find that Inception-v3 has the largest number of correctly classify foregrounds under 224×224 pixels patches, and the largest number of correctly classify background patches is ViT. In addition, the number of foreground patches misclassify by the ViT network model is much smaller than that of the CNNs network. At the same time, the number of correctly classify foreground in the CNNs network is greater than that of the ViT network.

4. Conclusion and Future Work

In this paper, we aim at the problem that transparent images are difficult to classify by cropping the image into patches and classifying the foreground and background. We use CNNs (ResNet50, Inception-V3, VGG-16, X-Inception) and ViT deep learning methods to compare the performance of classifying patches of transparent images. In addition, we also compare the effects of patches of 8×8 and 224×224 pixels on the classification performance of deep learning methods. We

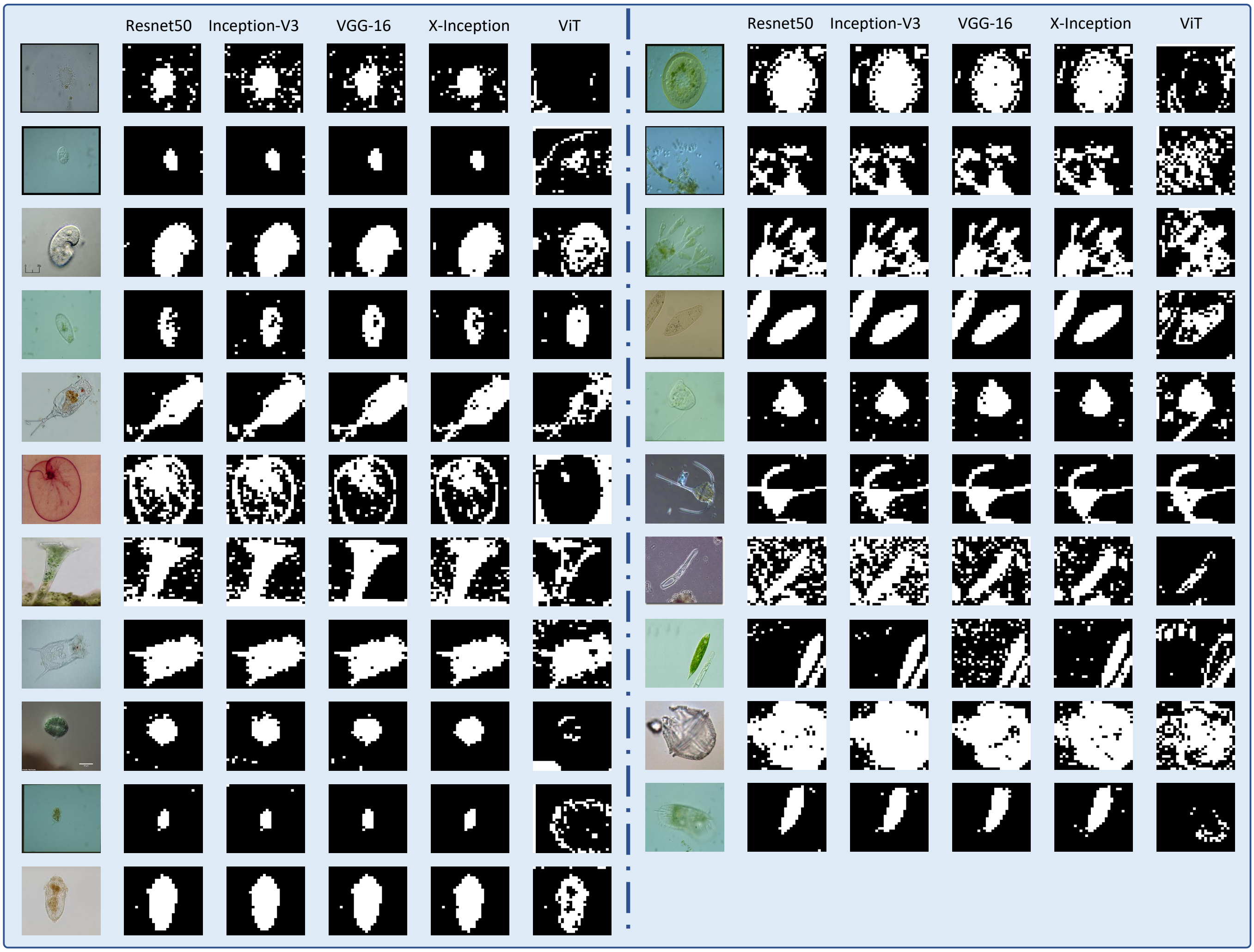


Figure 8: Reconstruction of 224 × 224 pixels transparent images classification results.

conclude that CNNs have better classification performance than ViT in patches of 8×8 pixels. However, the classification performance of ViT at 256×256 pixels is better than that of most CNNs. Therefore, we conclude that CNNs and ViT network models have more advantages in image classification. CNNs are good at extracting local features of images, and ViT is good at extracting images global features.

In the future, we plan to increase the amount of data to improve the stability of the comparison. Meanwhile, the images reconstructed by deep learning classification can be extended to the positioning, segmentation, recognition and detection of transparent images. We need to further strengthen the application of results.

5. Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 61806047). We thank Miss Zixian Li and Mr. Guoxian Li for their important discussion.

References

[1] Shu-Yuan Liao, Oscar N Aurelio, Kevin Jan, Jan Zavada, and Eric J Stanbridge. Identification of the mn/ca9 protein as a reliable diagnostic biomarker of clear cell carcinoma of the kidney. *Cancer research*, 57(14):2827–2831, 1997.

[2] Zihan Li, Chen Li, Yudong Yao, Jinghua Zhang, Md Mamunur Rahaman, Hao Xu, Frank Kulwa, Bolin Lu, Xuemin Zhu, and Tao Jiang. Emds-5: Environmental microorganism image dataset fifth version for multiple image analysis tasks. *Plos one*, 16(5):e0250631, 2021.

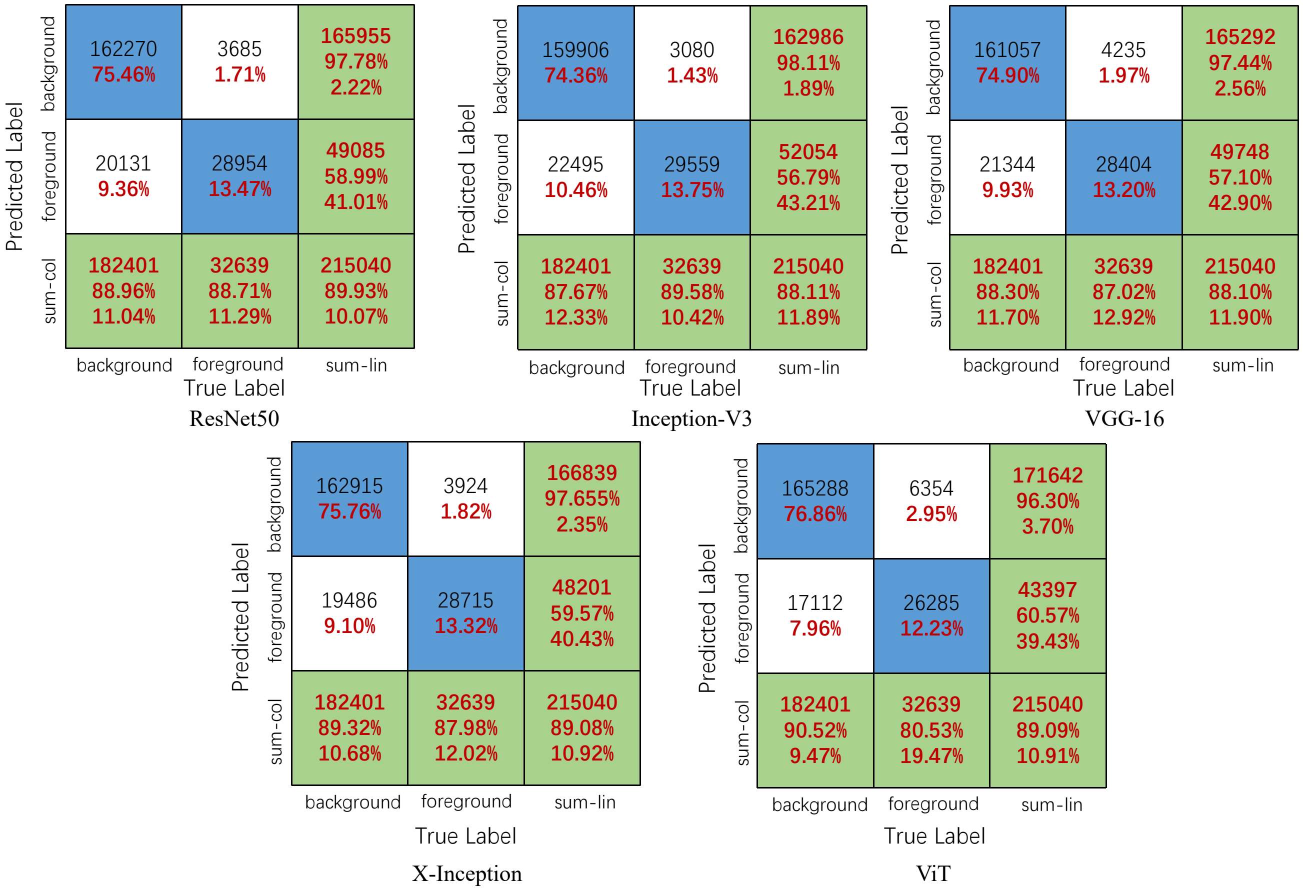
[3] May Phyo Khaing and Mukunoki Masayuki. Transparent object detection using convolutional neural network. In *International Conference on Big Data Analysis and Deep Learning Applications*, pages 86–93. Springer, 2018.

[4] Jay Martin Tenenbaum. Accommodation in computer vision. Technical report, Stanford Univ Ca Dept of Computer Science, 1970.

[5] Michael Isard and Andrew Blake. Contour tracking by stochastic propagation of conditional density. In *European conference on computer vision*, pages 343–356. Springer, 1996.

[6] Michael D Kelly. Edge detection in pictures by computer using planning. Technical report, STANFORD UNIV CALIF DEPT OF COMPUTER SCIENCE, 1970.

[7] Vicki Bruce and Andy Young. Understanding face recognition.



	background	foreground	sum-lin
background	162270 75.46%	3685 1.71%	165955 97.78% 2.22%
foreground	20131 9.36%	28954 13.47%	49085 58.99% 41.01%
sum-col	182401 88.96% 11.04%	32639 88.71% 11.29%	215040 89.93% 10.07%

ResNet50

	background	foreground	sum-lin
background	159906 74.36%	3080 1.43%	162986 98.11% 1.89%
foreground	22495 10.46%	29559 13.75%	52054 56.79% 43.21%
sum-col	182401 87.67% 12.33%	32639 89.58% 10.42%	215040 88.11% 11.89%

Inception-V3

	background	foreground	sum-lin
background	161057 74.90%	4235 1.97%	165292 97.44% 2.56%
foreground	21344 9.93%	28404 13.20%	49748 57.10% 42.90%
sum-col	182401 88.30% 11.70%	32639 87.02% 12.92%	215040 88.10% 11.90%

VGG-16

	background	foreground	sum-lin
background	162915 75.76%	3924 1.82%	166839 97.655% 2.35%
foreground	19486 9.10%	28715 13.32%	48201 59.57% 40.43%
sum-col	182401 89.32% 10.68%	32639 87.98% 12.02%	215040 89.08% 10.92%

X-Inception

	background	foreground	sum-lin
background	165288 76.86%	6354 2.95%	171642 96.30% 3.70%
foreground	17112 7.96%	26285 12.23%	43397 60.57% 39.43%
sum-col	182401 90.52% 9.47%	32639 80.53% 19.47%	215040 89.09% 10.91%

ViT

Figure 9: Predict the confusion matrix on test set of 224 × 224 pixels patches

British journal of psychology, 77(3):305–327, 1986.

[8] Pierre Baldi and Yves Chauvin. Neural networks for fingerprint recognition. *neural computation*, 5(3):402–418, 1993.

[9] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3):362–386, 2020.

[10] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19:221–248, 2017.

[11] Bernd Jähne and Horst Haußecker. Computer vision and applications. 2000.

[12] Joao Carreira, Henrique Madeira, and Joao Gabriel Silva. Xception: A technique for the experimental evaluation of dependability in modern computers. *IEEE Transactions on Software Engineering*, 24(2):125–136, 1998.

[13] Qing Guan, Yunjun Wang, Bo Ping, Duanshu Li, Jiajun Du, Yu Qin, Hongtao Lu, Xiaochun Wan, and Jun Xiang. Deep convolutional neural network vgg-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study. *Journal of Cancer*, 10(20):4876, 2019.

[14] A Sai Bharadwaj Reddy and D Sujitha Juliet. Transfer learning with resnet-50 for malaria cell-image classification. In *2019 International Conference on Communication and Signal Processing (ICCSP)*, pages 0945–0949. IEEE, 2019.

[15] Xiaoling Xia, Cui Xu, and Bing Nan. Inception-v3 for flower classification. In *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pages 783–787. IEEE, 2017.

[16] Hong-Yen Chen and Chung-Yen Su. An enhanced hybrid mobilenet. In *2018 9th International Conference on Awareness Science and Technology (iCAST)*, pages 308–312. IEEE, 2018.

[17] Fredy Martínez, Fernando Martínez, and Edwar Jacinto. Performance evaluation of the nasnet convolutional network in the automatic identification of covid-19. *Int J Adv Sci Eng Inform Technol*, 10(2):662, 2020.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[19] Andy Zeng, Kuan-Ting Yu, Shuran Song, Daniel Suo, Ed Walker, Alberto Rodriguez, and Jianxiong Xiao. Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 1386–1383. IEEE, 2017.

[20] Shreeyak Sajjan, Matthew Moore, Mike Pan, Ganesh Nagaraja, Johnny Lee, Andy Zeng, and Shuran Song. Clear grasp: 3d shape estimation of transparent objects for manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3634–3642. IEEE, 2020.

[21] Zheng Chunjiao. The application and development of photoelectric sensor. In *Intelligence Computation and Evolutionary Computation*, pages 671–677. Springer, 2013.

[22] Seiji Hata, Yoko Saitoh, Syoji Kumamura, and Ken'ichi Kaida. Shape extraction of transparent object using genetic algorithm. In *Proceedings of 13th International Conference on Pattern Recognition*, volume 4, pages 684–688. IEEE, 1996.

[23] Yichao Xu, Hajime Nagahara, Atsushi Shimada, and Rin-ichiro Taniguchi. Transcut: Transparent object segmentation from a light-field image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3442–3450, 2015.

[24] Yuanhao Guo, Zhan Xiong, and Fons J Verbeek. An efficient and robust hybrid method for segmentation of zebrafish objects from bright-field microscope images. *Machine vision and applications*, 29(8):1211–1225, 2018.

[25] Abozar Nasirahmadi and Seyed-Hassan Miraei Ashtiani. Bag-of-feature model for sweet and bitter almond classification. *Biosystems engineering*, 156:51–60, 2017.

[26] Yichao Xu, Kazuki Maeno, Hajime Nagahara, Atsushi Shimada, and Rin-ichiro Taniguchi. Light field distortion feature for transparent object classification. *Computer Vision and Image Understanding*, 139:122–135, 2015.

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[29] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[30] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[31] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[32] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[33] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.