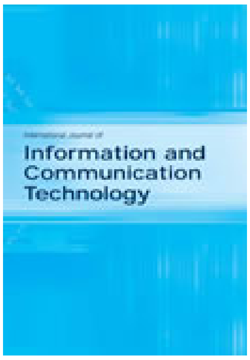


International Journal of
Information and Communication Technology

International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Image fusion using a transfer learning-based convolutional neural network

Mudan Lv, Aiping Cai

Article History:

Received:	28 August 2024
Last revised:	09 December 2024
Accepted:	10 December 2024
Published online:	13 February 2025

Image fusion using a transfer learning-based convolutional neural network

Mudan Lv and Aiping Cai*

School of Information Engineering,
Jiangxi University of Technology,
Nanchang City, 330098, China
Email: caiaiping2016@163.com
Email: aipingcai@yeah.net
*Corresponding author

Abstract: Convolution neural network (CNN) is a deep learning model that is widely used in image recognition, image classification. However, traditional deep learning models require extensive annotation information during training, leading to prolonged training times that directly impacts efficiency and performance. In order to solve this problem, this paper proposes a five-layer convolution neural network structure based on transfer learning to extract, train and fuse features of source images. First, an improved VGG-19 network is used to extract the preliminary features of the source images, and the training samples are transferred to the encoder for deep feature extraction by setting the VGG-19 network parameters. Then, the extracted feature samples and a five-layer U-Net neural network construct the decoder for feature reconstruction. Batch normalisation is applied to prevent over-fitting of the model. Finally, the loss function is applied layer-by-layer in supervised learning to obtain the quadratic decision graphs that are used to fuse the source images to generate the output images. The proposed model in this paper demonstrates a significant enhancement in the visual effect of images compared with other models.

Keywords: deep learning; convolutional neural network; CNN; transfer learning; image fusion.

Reference to this paper should be made as follows: Lv, M. and Cai, A. (2025) ‘Image fusion using a transfer learning-based convolutional neural network’, *Int. J. Information and Communication Technology*, Vol. 26, No. 3, pp.72–88.

Biographical notes: Mudan Lv is a Lecturer and has a Master of Engineering from the East China Institute of Technology, Nanchang, China, in 2017, where she is currently teaches at Jiangxi University of Technology. Her main research fields are image processing and database application technology.

Aiping Cai is a Master’s candidate and an Associate Professor, he is also currently working as a software engineer at the Jiangxi University of Technology in Nanchang, China. His research interests include algorithms, image processing, and computer vision. He received his Master’s in Computer Science and Information Technology from Wuhan Institute of Technology in 2008. His main research interests include image classification and data processing, distributed data structure and database technology.

1 Introduction

Traditional transfer learning methods extract samples or features from different domains, and reduce the distribution difference between the source image and the target image in the original feature space or subspace by weighting (Shuhui et al., 2020; Zhang et al., 2020) and subspace learning (Niu et al., 2016; Wang et al., 2019). Therefore, the model trained on the data with image annotations can obtain higher performance on the unlabelled target images. Although transfer learning can solve the problem of few labelled samples in machine learning (Zhuang et al., 2020; Du and Ikenaga, 2020; Chen et al., 2019; Yu et al., 2023; Zhou et al., 2023; Liu et al., 2023a), in practical application scenarios, the direct difference between the source image and the target data will greatly reduce the effect of machine learning. Zhu et al. (2022) uses class conditional distribution to replace the conditional distribution of sample data, and realises channel attention modelling and spatial attention to enhance features. Compared with before the introduction of attention mechanism, the effect of the algorithm has been greatly improved. Guo et al. (2022) uses attention mechanism to process key regions and obtain beneficial feature information through this region.

With the advancement of deep learning, the use of transfer learning algorithms (Zheng et al., 2023; Li et al., 2023; Yang et al., 2024; Li et al., 2024a, 2024b; Abdul et al., 2023) based on convolutional neural network (CNN) models has become widespread. Typically, the objective function of a deep neural network model is incorporated into the feature alignment method for transfer learning training (Zhang et al., 2023). In Wang et al. (2021), a domain adaptive neural network model (DaNN) was proposed to address the issue of transferability of training samples. In Sun et al. (2020), the deep domain confusion (DDC) method was introduced and the AlexNet network was used for training samples. In Liu et al. (2023b), a domain discriminator was incorporated into the network model and an adversarial learning approach was utilised to identify edge samples. In He et al. (2023), a combination of marginal and conditional distributions was used for deep transfer learning, allowing for the extraction of feature samples from initial data and enabling end-to-end training.

In the study of multi-source transfer learning, a new framework was proposed in Zhu et al. (2019) to address the distribution differences between all domains within the same feature space. Another approach, described in Yun and Lee (2024), utilises an adversarial network for transfer learning, consisting of four components: real data, noise data, generator, and discriminator. This method performs transfer learning on noise data to generate a feature extractor for the target domain. Building upon the concept of a domain discriminator, Jiang et al. (2023) and Hong and Ryu (2020) introduces a domain adversarial network that can distinguish between source and target images based on task-specific features. In Wan et al. (2022), an adaptive factor known as domain-adversarial neural network (DANN) is incorporated into the adversarial transfer learning network to address the challenge of dynamic distribution adaptation. Additionally, Chen et al. (2023) attempts to solve the automatic matching issue in adversarial learning through cyclic mapping. Finally, Ma et al. (2020) proposes a multi-source domain adaptation algorithm based on generative adversarial networks.

After analysing the above information, it can be concluded that traditional transfer learning primarily aims to minimise the disparity between the original image and the desired image. However, with the introduction of neural network transfer learning, the deep transfer learning algorithm has proven to be more effective than traditional methods.

In targeting multi-focused image fusion applications, different researches have been proposed, first of all, in terms of deep learning network structure, Du and Gao (2017) used multi-scale CNN for capturing features at different scales, which is more effective for dealing with fuzzy regions of different sizes. On the other hand, Liu et al. (2017) based on deep CNN for image fusion, which focuses on deep feature extraction and representation. The deep model for multi-focus image fusion based on gradient and connected regions proposed by Xu et al. (2020) focuses more on local features and regional connectivity of the image.

Secondly, in terms of fusion strategy, Liu et al. (2018) directly fused at feature level, CNN based multi-focus image fusion and others fused at decision map or pixel level, firstly generated decision maps by deep network and then fused images based on these decision maps. Wang et al. (2020) proposed a new CNN-based multi-focus image fusion method based on sequence reconstruction strategy of patches.

Additionally some methods may use specific loss functions to better handle the fusion problem of multi-focal images, such as considering the proportional maintenance loss of gradients and intensities. The evaluation metrics may also vary, some methods may focus more on objective metrics such as PSNR, SSIM, etc. (Liu et al., 2018; Wang et al., 2020), while others may also consider subjective evaluation metrics.

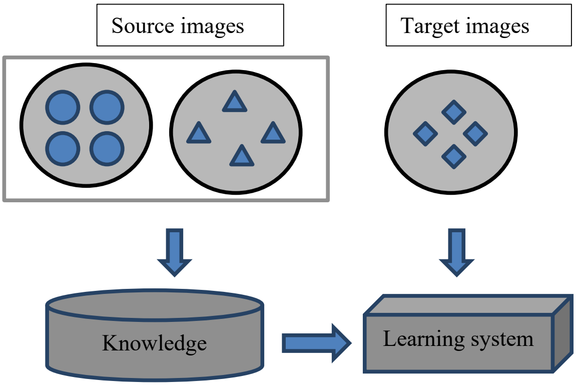
All of these methods use deep learning to solve the multi-focal image fusion problem, but they differ in terms of network structure, fusion strategy, loss function and optimisation, dataset and evaluation. These differences make each method have its unique advantages and applicable scenarios.

2 Transfer learning

Transfer learning is primarily designed for a single task or problem. For a given task, it proposes a learning approach that leverages real-world scenarios and existing data. Traditional machine learning trains each model independently for specific domains, data, and tasks. However, transfer learning offers a means to reapply a model or a piece of knowledge to other related problems. So, transfer learning is the re-application of one model or task to another model or target task for training. That is, given a source image (Ds) and source learning task (Ts), a target image (Dt) and target learning task (Tt), the learning process involves using prior knowledge from the source image and learning task to identify the objective function ft(•) in the target image, where Ds ≠ Dt and Ts ≠ Tt.

The process of transfer learning, based on feature extraction, involves mapping both the source and target images to a lower-dimensional space. This allows for the training of the source image and the prediction of data for the target domain. By reducing the error rate, transfer learning improves the recognition of the target domain and minimises the differences between the two fields. Figure 1 illustrates the schematic diagram of transfer learning. The specific operations include keeping the weights and parameters of the source image training samples unchanged, while replacing the dataset in the middle layer with the dataset for image object recognition. The CNN structure is then reused to retrain the samples. This approach results in a new model capable of recognising whether a target object is present in an image, without the need to train a new model from scratch.

Figure 1 Schematic diagram of transfer learning (see online version for colours)



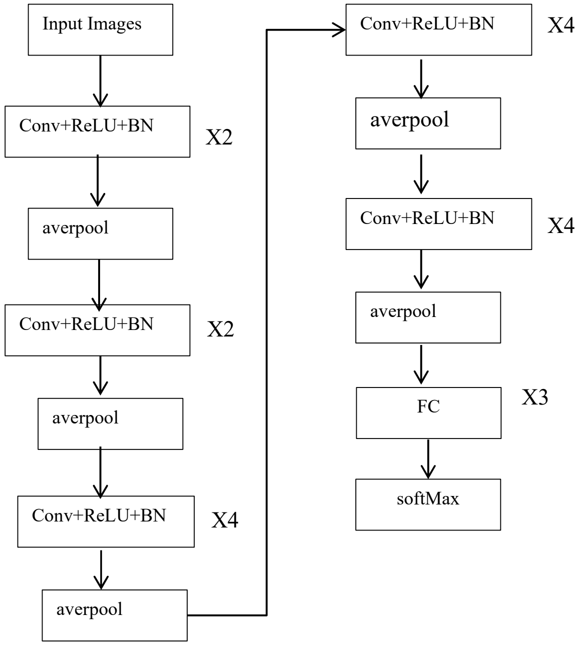
3 Transfer learning-based neural network for image fusion

3.1 The improved VGG-19 network

The improved VGG-19 structure is optimised on the basis of the original VGG-19, which mainly involves two key parts: adding a batch norm layer after each convolutional block, and replacing the original max pooling with average pooling. In the improved VGG-19 structure, a batch normalisation (BN) layer is added after each convolutional block, which helps to accelerate the training process of the model, increase the convergence speed of the model, and reduce the risk of over fitting. With BN, the model can learn the feature representation of the data more consistently, which improves the generalisation ability of the model. In the original VGG-19 structure, maximum pooling is used to perform down sampling operations after each convolutional block. However, maximum pooling may lose some useful information because it only retains the maximum value in each pooling window. In contrast, average pooling can retain more information because it computes the average of all values in each pooling window. In the improved VGG-19 structure, using average pooling can retain more image details and global information, which helps the model to better understand and recognise features in the image.

The improved VGG-19 neural network model has a total depth of 19 layers, with a fixed input source image size of 224 * 224 pixels. The first 16 layers are CNN layers, each consisting of five convolutional sub modules. These sub modules have a convolution kernel size of 3 * 3 and a step size of 1, which are used to extract feature information from the input image. The remaining three layers are fully connected layers, with a ReLU activation function for nonlinear processing. The network has a total of five convolutional sub modules, with sub modules I and II sharing the same structure of two convolutional layers, and sub modules III, IV, and V sharing the same structure of four convolutional layers. The output feature maps from each sub module are then fed into a maximum pooling layer for down sampling. This down sampling reduces the size of the feature maps to one-fourth of the original while keeping the number of channels unchanged. To prevent over fitting and information loss caused by mean pooling, normalisation is incorporated into the network. The improved structure of the VGG-19 network is illustrated in Figure 2.

Figure 2 Structure diagram of the improved VGG-19 network model



3.2 Feature extraction module

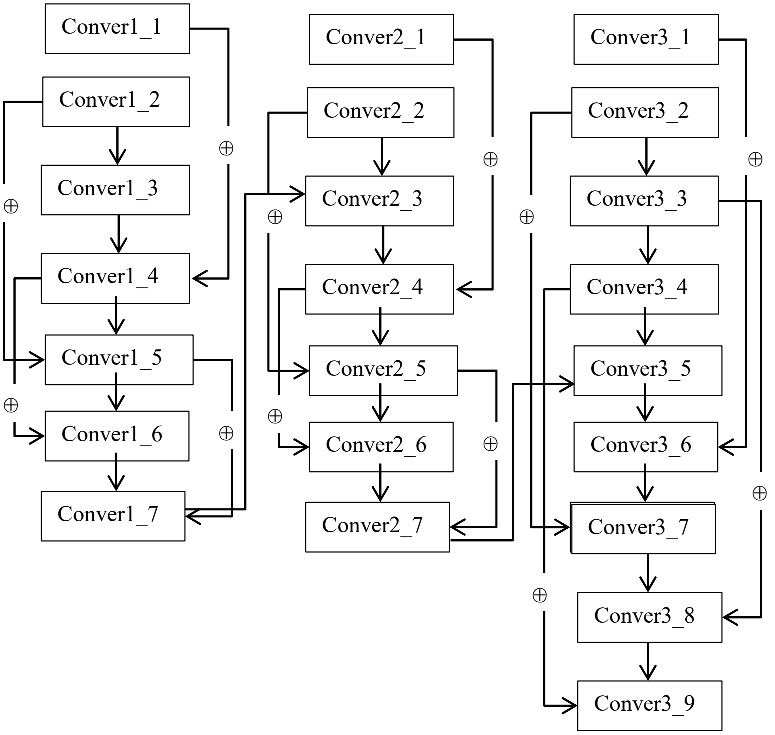
First, we prepared multiple pairs of colour multi-focus source images as input data. A convolution operation is performed on each of the multi-focus source images using an already pre-trained VGG-19 network model. The purpose of this step is to initially extract the key features in the images through the powerful feature learning capability of the VGG-19 network. We then transfer the decoded convolutional layer parameters from the VGG-19 network to our newly designed feature extraction module. This migration process leverages the a priori knowledge of the pre-trained model, allowing our feature extraction module to adapt to new task requirements more quickly and accurately. After migrating the parameters, we employ the redesigned feature extraction module to process the initially extracted features further. Through deeper convolutional operations and feature transformations, we are able to extract finer and more discriminative image features. After this series of feature extraction and processing steps, we have completed the feature training process and obtained an in-depth feature representation of each multi-focus source image. These features will provide an important information base for our subsequent decision map generation and pixel-level fusion.

The feature extraction module is comprised of multiple encoders. Encoders I and II are constructed with a seven-layer CNN. The network architecture for encoders III, IV, and V is identical.

Firstly, we utilised the VGG-19 neural network model to train the source image and extract its initial features. Secondly, we established the parameters for an enhanced VGG neural network model, with a 3 * 3 convolution kernel size, a stride of 1, and SAME padding in the middle layer. Thirdly, we performed secondary feature extraction on the source image using the improved VGG-19 neural network model. This approach employs the enhanced VGG-19 neural network model to segment various regions within the image, facilitating the transfer of parameters and models.

As shown in Figure 3, after the source image is input into the network, it goes through five encoders for feature extraction. Thus, after the source image is feature extracted by the encoders, the output image is reduced to one-fourth the size of the source image, and the feature extracted image of the whole image is changed from the original 224 * 224 to 7 * 7 pixels. A step-by-step convolution kernel block (3 * 3) is used to pool the feature images of each layer, which serves to reduce the dimensionality of the extracted feature images, with the aim of retaining the detailed information of the source image, and the parameter stride of the step-by-step convolution block is set to 2 and padding is set to SAME.

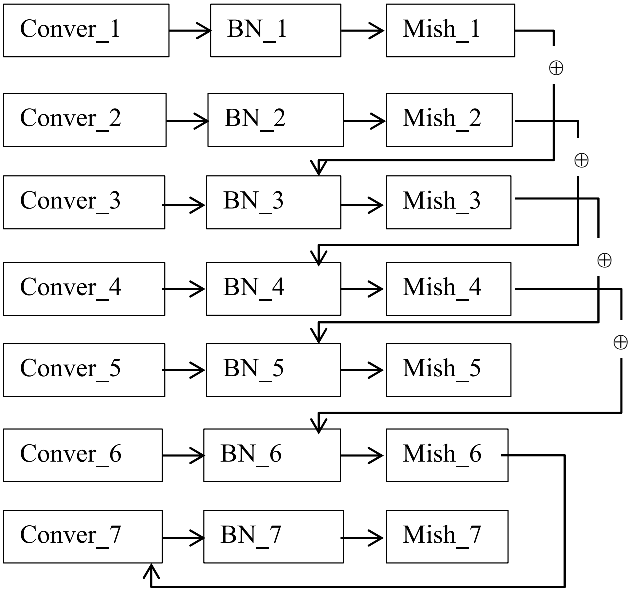
Figure 3 Diagram of feature extraction module



The structure of Encoder1 (Encoder1) is shown in Figure 4, where each convolutional block consists of a convolutional layer, a BN layer and a Mish activation function layer. Similarly, the convolutional blocks of other encoders have the same structure, i.e., each convolutional layer is followed by a BN layer and a Mish activation function layer.

The idea of implementing transfer learning in this paper is shown in Figure 5. In the feature extraction module, first, we input the source images into the pre-trained VGG-19 network to utilise its powerful feature learning capability for initial feature extraction. The VGG-19 network, as a deep CNN that performs well in a variety of source-domain tasks, has accumulated rich visual knowledge. Subsequently, we transfer all the convolutional layer parameters from the VGG-19 network to our newly designed network. This step is based on the principle of transfer learning, which speeds up learning and improves performance by reusing parameters from pre-trained models and avoiding training from scratch. We then combine the migrated convolutional layers with our newly designed layers to form a new feature extraction module. This module incorporates both the generic features learned by the VGG-19 network and the specialised feature extraction capabilities we designed for a specific task (i.e., colour image fusion).

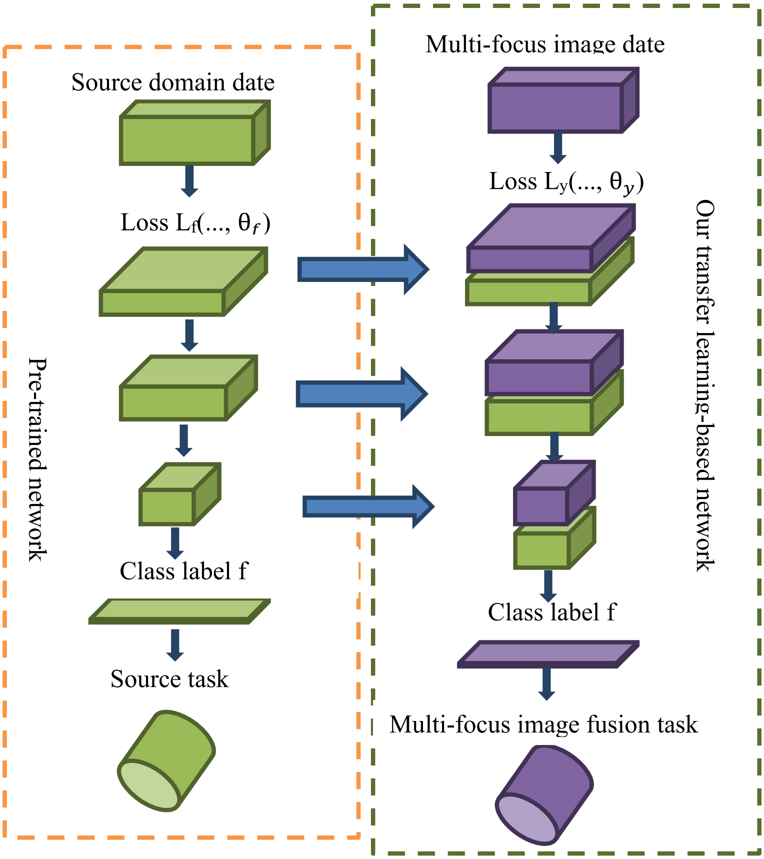
Figure 4 Encoder structure



3.3 Feature reconstruction module

After the feature map output from the feature extraction module is input to the feature reconstruction module, it will be processed by five decoders in order to up-sample the feature map and reconstruct the features. The workflow of decoder 1 is shown in Figure 6, the feature map will be enlarged to four times of its original size after the processing of the inverse convolutional layer, and fused with the feature map of the same size output from the convolutional layer in the encoder through the hopping connection structure, and then the fused feature map is inputted into the next feature reconstruction convolutional layer for convolution in order to reconstruct the features. And so on, the feature maps from the previous feature extraction convolutional layer will be fused with the output feature maps from the convolutional layer of the encoder in the same way, and then fed into the next feature reconstruction convolutional layer for feature reconstruction. Finally, after processing by five decoders, the output of the feature reconstruction module is a binary decision map labelling the focused and unfocused regions in the source image.

Figure 5 Diagram of the implementation of migratory learning (see online version for colours)


Source domain date
Loss $L_f(\ldots, \theta_f)$
Class label f
Source task
Pre-trained network
Multi-focus image date
Loss $L_y(\ldots, \theta_y)$
Class label f
Multi-focus image fusion task
Our transfer learning-based network

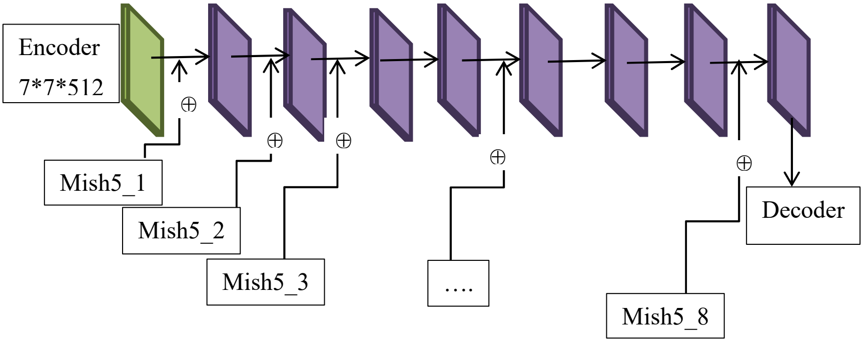
3.4 *Jump connection structure*

The feature reconstruction module is the inverse operation of the feature encoder in the feature extraction module, meaning it performs the decoding process. It is composed of one group of DE convolution neural network blocks and eight groups of feature decoding CNN blocks. To ensure proper connectivity, a BN layer and Mishap activation function are added after each DE CNN block and feature decoding convolutional block. These blocks are connected using a '⊕' mark, indicating their fusion into the next layer of the network. This structure allows for the integration of low-frequency features from the low-level layer with high-frequency features from the high-level layer, enabling the network to learn more details. The formula for this connection operation is as equation (1):

$$y_i = h(x_{j-1}) + M(C_{i-1} * W_i) \tag{1}$$

where $h(x_{j-1})$ denotes the output feature mapping of the $j - 1^{th}$ network layer, C_{i-1} is the output feature map of the $i - 1^{th}$ network layer, W_i denotes the convolution kernel parameter of the ith layer, M(.) denotes the Mish activation function, and y_i denotes the input feature map of the ith layer.

Figure 6 Workflow diagram of the decoder (see online version for colours)



3.5 Fusion strategy

The fusion process involves combining the focus area $M_1(i, j)$ and the non-focus area $M_2(i, j)$ of the feature map extracted from the source image. The source image is a binary image with values of 0 and 1. During the fusion process, an error value may occur. If this value is not 1, $M_3(i, j)$ is obtained through an AND operation. M_1 and M_2 are then multiplied with M_3 to obtain $M_4(i, j)$ and $M_5(i, j)$, respectively. These images are then multiplied with the source image to obtain the primary fusion images $F_1(i, j)$ and $F_2(i, j)$. The QABF values of F_1 and F_2 are then calculated, and the image with the higher QABF value is selected as the final output, or target image. The specific steps involved in this process are outlined below.

```
Input feature image: focus area $M_1(i, j)$ and the non-focus area $M_2(i, j)$
Output fused image:
1    Primary fusion image $F_1(i, j)$ and $F_2(i, j)$
2    If $M_1(i, j) == 1$ and $1 – M_2(i, j) == 1$
     $M_3(i, j) = 1$,
3    Else
     $M_3(i, j) = 0$, among them $1 << i, j << x$
4    Then,
     $M_4(i, j) = M_1(i, j) * M_3(i, j)$
     $M_5(i, j) = M_2(i, j) * M_3(i, j)$
5    It is obtained by fusion calculation
     $F_1(i, j) = M_4(i, j) * S_2(i, j) + (1 – M_4(i, j)) * S_1(i, j)$
     $F_2(i, j) = M_5(i, j) * S_1(i, j) + (1 – M_5(i, j)) * S_2(i, j)$
6    If
     $Q^{ABF} (F_1(i, j)) >= Q^{ABF} (F_2(i, j))$
7    Return $F_1(i, j)$
8    Else return $F_2(i, j)$
```

3.6 Layer-by-layer loss function

In the feature reconstruction module, the final output feature map of each encoder is enlarged to match the size of the source image. This enlarged feature map is then input into the mixed loss function along with the label image to calculate the primary loss value. As there are five encoders, five different loss values are calculated. These values are then weighted and averaged to obtain the final loss value, which is used as the iterative factor for further calculations. The formula for the loss function is shown in equation (2):

$$F = \frac{1}{x}\sum_{i=1}^{x}\left(l_1^i + l_2^i\right) \tag{2}$$

where x represents the number of layers of the CNN block l^i represents the loss value calculated by the layer-by-layer loss function, the normalised probability of the feature image S is expressed as equation (3), and the layer-by-layer loss function is expressed as equations 4 and (5):

$$S_i = \frac{e^{V_i}}{\sum_j e^{V_i}} \tag{3}$$

$$l_1 = L_{\text{Cross entropy}} = -\frac{1}{k}\sum_{i=1}^{k} y_i' \log\left(y_i\right) \tag{4}$$

$$l_2 = L_{L2} = -\frac{1}{k}\sum_{i=1}^{k}\left(f\left(x_i\right) - y_i'\right)^2 \tag{5}$$

Here, S denotes the output component used for network normalisation, which also represents the predicted probability of each category. Y' signifies the true value of the i^{th} label image, while y denotes the probability component normalised by S. Additionally, k represents the number of feature images included in this training set.

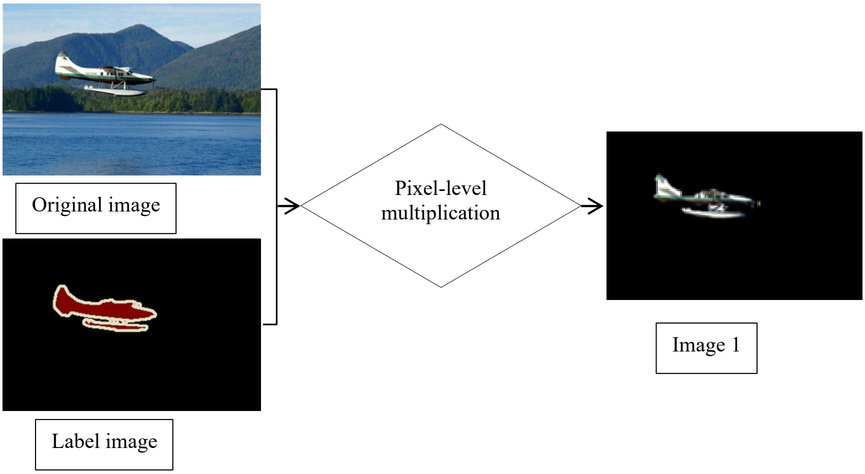
Here, $f(x_i)$ denotes the predicted value for the i^{th} image, and y' denotes the true value for the i^{th} label image.

4 Experiments

4.1 Parameter settings

We used the initialisation method of ESTAR (Suhartono et al., 2018) to initialise the convolution kernel parameters of all convolutional layers and DE convolution layers, except the convolutional layer migrated from the VGG-19 network. The RMSprop (Reham et al., 2023) optimiser is ultimately adopted to train the network following extensive experimentation with various optimisers. The initial learning rate is set to 1e-6, weight decay is set to 0.001 and momentum is set to 0.9.

Figure 7 Flowchart of dataset production (see online version for colours)



4.2 Dataset training

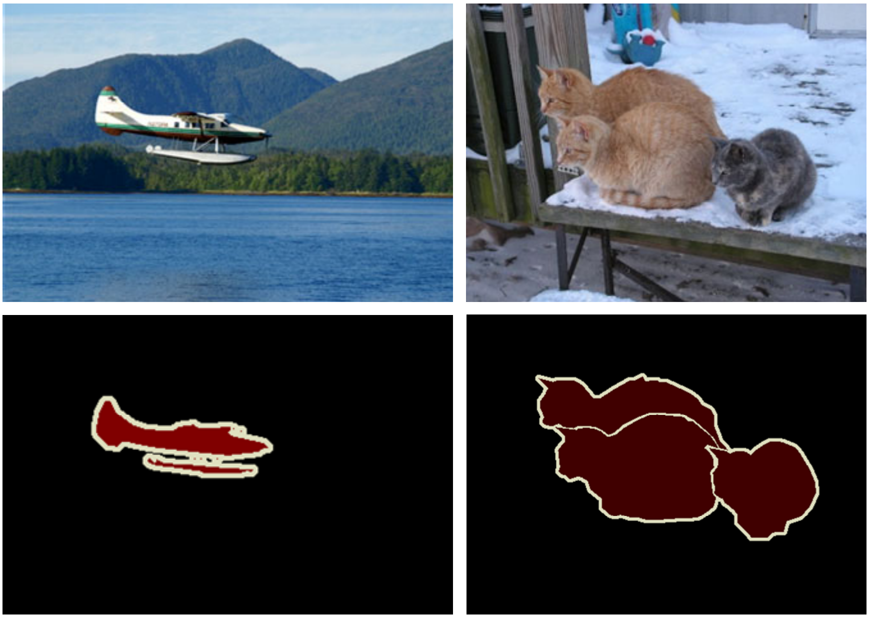
2,000 natural images and corresponding 2,000 labelled images were selected from the PASCAL VOC (Tong and Wu, 2023) image dataset for making the multi-focus image dataset. Multi-focus image fusion is the process of fusing two or more (usually two) images containing different focus regions in the same scene into a single image, so that the fused image presents a fully focused scene. This segmented image dataset is categorised into 21 classes and contains a total of 2,913 natural images of 6,929 objects and multiple scenes for the image segmentation task. Next, we blurred these 2,000 natural and people images after 5 Gaussian filtering algorithms, one Gaussian filtering (block size 3 * 3, standard deviation set to 2) blurring operation was performed on the source image to obtain a blurring level of 1, then another Gaussian filtering was performed on the blurring image to obtain a blurring level of 2, and so on and so forth, after five Gaussian filtering operations, the five levels of fuzzy images are obtained respectively, each fuzzy level contains 2,000 images, a total of 10,000 fuzzy level images are obtained. After that, the source image (clear image) and the fuzzy image are extracted by pixel-level multiplication of the source image (clear image) and the fuzzy image respectively to extract the corresponding focused and unfocused regions, and finally, the extracted focused and unfocused regions are fused by pixel-level addition to obtain the multi-focused image set of five different focusing levels.

The specific dataset production process is shown in Figure 7, in the labelled image 1, the pixel value of the black area is 0, and the pixel value of the white area is 1, so that when the labelled image 1 is pixel-level multiplied with the fully blurred image, the extracted image 1 with a background pixel value of 0 and a blurred foreground object can be obtained. Similarly, when the labelled image 1 is pixel-level multiplied with the source image (clear image), the extracted image 2 with a background pixel value of 0 and extracted image 2 with clear foreground objects.

4.3 Experimental comparison

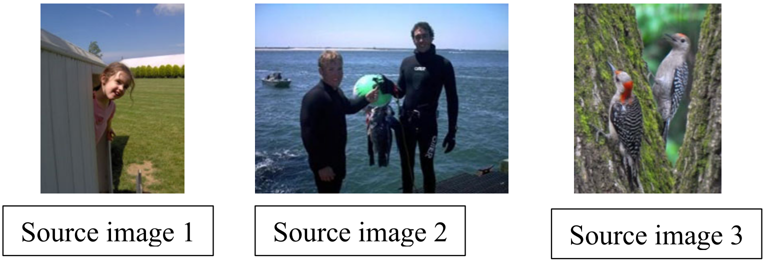
In the annotated image 1 of Figure 8, the pixel value in the black area is represented as 0; the pixel value in the white underlined area is represented as 1. The extracted image 1 was obtained by multiplying label image 1 and the full fuzzy image at a pixel level, and its background pixel has a value of 0 and the foreground object appears fuzzy. Similarly, the extracted image 2 was obtained by multiplying label image 1 and the source image (clear image) at a pixel level, and its background pixel has a value of 0 and the foreground object appears clear. After acquiring the four extracted images, adding them at the pixel level obtains the multi-focus image 1 with a clear foreground and fuzzy background and the multi-focus image 2 with fuzzy foreground and clear background, respectively. The Original image and its corresponding annotated images are shown in Figure 8.

Figure 8 Original image and its corresponding annotated images (see online version for colours)



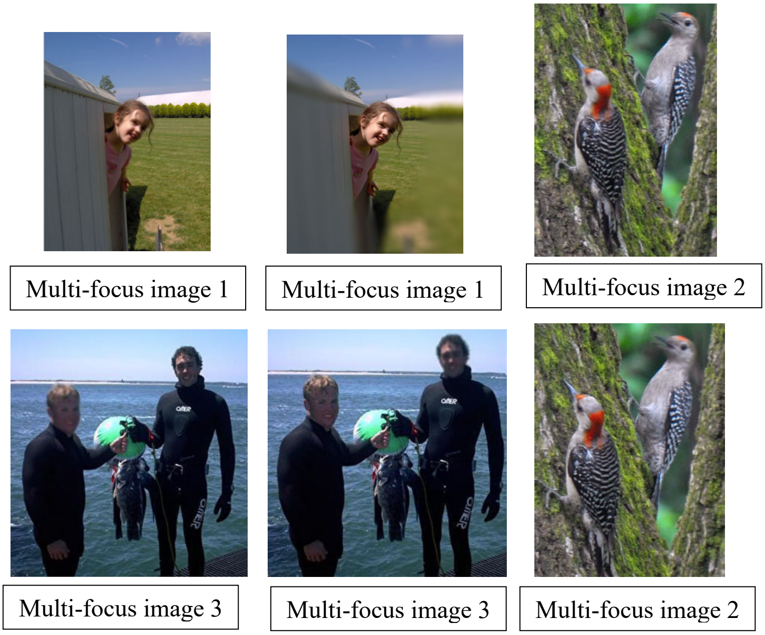
We used the quantitative and qualitative aspects of the fusion image quality assessment for evaluation, and compared the evaluation results with ASR (Liu and Wang, 2015), CSR (Liu et al., 2016), DWT (Oliver, 1990), MSVD (Naidu, 2011), GD (Paul et al., 2016), MSTSR (Liu et al., 2015), MGIVF (Bavirisetti et al., 2016) and other methods. The comparative experimental data are shown in Tables 1–3. QABF denotes edge retention, QAG denotes average gradient, QLABF denotes overall information loss, QMI denotes mutual information, and QSF denotes spatial frequency. The natural scene images are shown in Figure 9 and the multi-focus scene images are shown in Figure 10.

Figure 9 Natural scene images (see online version for colours)



Source image 1 Source image 2 Source image 3

Figure 10 Multi-focus scene images (see online version for colours)



Multi-focus image 1 Multi-focus image 1 Multi-focus image 2

Multi-focus image 3 Multi-focus image 3 Multi-focus image 2

Table 1 Source image 1 quantitative assessments comparison of different methods

Methods	Q^{ABF}	Q^{AG}	Q^{LABF}	Q^{MI}	Q^{SF}
ASR	0.733	7.743	0.248	6.535	19.320
CSR	0.705	7.621	0.238	6.560	19.282
DWT	0.671	7.142	0.298	5.882	17.331
MSVD	0.481	6.988	0.429	6.090	16.981
GD	0.714	6.612	0.370	5.880	15.884
MSTSR	0.742	7.923	0.242	6.880	19.551
MGIVF	0.610	6.750	0.360	5.780	16.382
PCA	0.465	9.013	0.519	3.812	17.011
Proposed method	*0.756*	*8.148*	*0.210*	*8.510*	*20.725*

The above important formulas are shown in equations (6)–(8).

$$Q^{ABF} = \frac{\sum_{i=1}^{n1}\sum_{j=1}^{n2}\left(Q^{AF}(i,j)W^{A}(i,j) + Q^{BF}W^{B}(i,j)\right)}{\sum_{i=1}^{n1}\sum_{j=1}^{n2}\left(W^{A}(i,j) + W^{B}(i,j)\right)} \tag{6}$$

$Q^{AF}(i, j)$ represents the edge information transferred from image source A to fused image F.

$$Q^{AG} = \frac{1}{(M-1)(N-1)}\sum_{i=1}^{M-1}\sum_{j=1}^{N-1}\sqrt{\frac{\left(I(i+1,j)-I(i,j)\right)^2 + \left(I(i,j+1)-I(i,j)\right)^2}{2}} \tag{7}$$

M, N is the width and height of the image, and I(i, j) represents the pixel value of the image at position I and j.

$$Q^{MI} = \frac{JE_{A,F} + JE_{B,F}}{IE_A + IE_B} \tag{8}$$

A, B represent two multi-focus source images respectively, F represents fusion image, $JE_{A,F}$ represents the joint entropy between A and F, IE_A and IE_B represent the information entropy of image A and image B respectively.

As can be seen from Figure 10, the image obtained by the fusion of MSVD and GD does not perform well in some details and edge areas, and the image obtained by the fusion of DWT and MGIVF has colour distortion. In general, our method has better results in processing image details and edges, and there are no obvious artefacts or blurred areas in the fused images, which indicates that our method is superior to most comparison methods in the subjective evaluation of human vision.

Table 2 Source image 2 quantitative assessments comparison of different methods

Methods	Q^{ABF}	Q^{AG}	Q^{LABF}	Q^{MI}	Q^{SF}
ASR	0.694	5.239	0.276	7.262	14.534
CSR	0.684	5.182	0.285	7.312	14.440
DWT	0.611	4.781	0.351	6.492	12.657
MSVD	0.541	4.742	0.417	6.912	12.621
GD	0.614	4.587	0.361	6.493	12.181
MSTSR	0.704	5.382	0.268	7.510	14.634
MGIVF	0.595	4.691	0.355	6.332	12.481
PCA	0.677	8.702	0.302	6.221	19.103
Proposed method	*0.745*	*6.102*	*0.242*	*9.016*	*14.642*

It can be seen from the above table that the QSF results obtained by most methods are relatively close each other, the QMI, QAG and QABF values obtained by the fusion of the proposed method are the best, and all indexes are closest to the MSTSR method, the QLABF value of the proper standard method is better, and the index is close to the ASR and CSR methods, Our proposed method outperforms other methods in terms of overall performance.

Table 3 Source image 3 quantitative assessments comparison of different methods

Methods	Q^{ABF}	Q^{AG}	Q^{LABF}	Q^{MI}	Q^{SF}
ASR	0.732	16.632	0.728	5.391	30.231
CSR	0.731	16.792	0.243	5.412	30.686
DWT	0.689	14.366	0.316	4.538	25.796
MSVD	0.501	13.612	0.469	4.151	25.761
GD	0.674	9.712	0.341	4.291	24.601
MSTSR	0.724	16.501	0.378	5.299	29.651
MGIVF	0.635	13.502	0.343	4.413	25.251
PCA	0.609	12.442	0.377	4.656	22.218
Proposed method	0.743	17.554	0.238	7.394	32.103

5 Conclusions

In this paper, a transfer learning network model with multilevel mixed functions is proposed. The parameter transfer method is used to extract the features of the improved VGG-19 neural network model, and the neural network convolution layer is transferred to the five-layer feature extraction module for deep feature extraction. The extracted image features are multi-decoded and reconstructed, and the low-frequency features and high-frequency features are fused with the connection operation to obtain more image detail information. In this paper, 5 layers of stride convolutional kernel blocks (3 * 3) are used to pool the feature images of each layer, which makes the CNN model easier to converge. Through the multiplication of the trained feature images and the source images, the fused image has higher image quality, indicating that our method has certain competitiveness.

Acknowledgements

This paper is supported by Foundation of the Science and Technology Research Project of Jiangxi Provincial Department of Education: Research on image fusion model of brain diseases based on multi-scale CNN (No. GJJ2202608).

References

Abdul, R., Dongsun, K., Anand, P. et al. (2023) 'Convolutional neural network model for fire detection in real-time environment', *Computers, Materials & Continua*, Vol. 77, No. 2, pp.2289–2307.

Bavirisetti, D.P., Xiao, G., Zhao, J.H. et al. (2019) 'Multi-scale guided image and video fusion: a fast and efficient approach', *Circuits Systems and Signal Processing*, December, Vol. 38, No. 12, pp.5576–5605.

Chen, P., Li, P., Li, Q. et al. (2019) 'Semi-supervised fine-grained image categorization using transfer learning with hierarchical multi-scale adversarial networks', *IEEE Access*, No. 2019, pp.7118650–118668.

Chen, S., Ma, D., Lee, S. et al. (2023) 'Segmentation-guided domain adaptation and data harmonization of multi-device retinal optical coherence tomography using cycle-consistent generative adversarial networks', *Computers in Biology and Medicine*, Vol. 159, No. 2023, pp.106595–106595.

Du, C. and Gao, S. (2017) 'Image segmentation-based multi-focus image fusion through multi-scale convolutional neural network', *IEEE Access*, Vol. 5, No. 2017, pp.15750–15761.

Du, S. and Ikenaga, T. (2020) 'STED-net: self-taught encoder-decoder network for unsupervised feature representation', *Multimedia Tools and Applications*, Vol. 80, No. 3, pp.1–19.

Guo, M.H., Xu, T.X., Liu, J.J. et al. (2022) 'Attention mechanisms in computer vision: a survey', *Computational Visual Media*, Vol. 8, No. 3, pp.1–38.

He, L., Bai, L., Yang, X. et al. (2023) 'Exploring the role of edge distribution in graph convolutional networks', *Neural Networks: The Official Journal of the International Neural Network Society*, Vol. 168, No. 2023, pp.459–470.

Hong, S. and Ryu, J. (2020) 'Attention-guided adaptation factors for unsupervised facial domain adaptation', *Electronics Letters*, Vol. 56, No. 16, pp.816–818.

Jiang, N., Fang, J., Shao, Y. et al. (2023) 'Learning invariant representation using synthetic imagery for object detection', *AI Communications*, Vol. 36, No. 1, pp.13–25.

Li, D., Peng, Y., Sun, J. et al. (2023) 'Unsupervised deep consistency learning adaptation network for cardiac cross-modality structural segmentation', *Medical & Biological Engineering & Computing*, Vol. 61, No. 10, pp.2713–2732.

Li, G., Chen, L., Fan, C. et al. (2024) 'Improved convolutional neural network chiller early fault diagnosis by gradient-based feature-level model interpretation and feature learning', *Applied Thermal Engineering*, Vol. 236, No. PB, p.121549.

Liu, D., Bai, L., Yu, T. et al. (2023a) 'Generalized few-shot classification with knowledge graph', *Neural Processing Letters*, Vol. 55, No. 6, pp.7649–7666.

Liu, X., Ji, Z., Pang, Y. et al. (2023b) 'Dual distillation discriminator networks for domain adaptive few-shot learning', *Neural Networks: The Official Journal of the International Neural Network Society*, Vol. 165, No. 2023, pp.625–633.

Liu, Y. and Wang, Z.F. (2015) 'Simultaneous image fusion and denoising with adaptive sparse representation', *IET Image Processing*, Vol. 9, No. 5, pp.347–357.

Liu, Y. et al. (2017) 'Multi-focus image fusion with a deep convolutional neural network', *Information Fusion*, Vol. 36, No. 2017, pp.191–207.

Liu, Y. et al. (2018) 'Deep learning for pixel-level image fusion: recent advances and future prospects', *Information Fusion*, Vol. 42, No. 2018, pp.158–173.

Liu, Y., Liu, S.P. and Wang, Z.F. (2015) 'A general framework for image fusion based on multi-scale transform and sparse representation', *Information Fusion*, July, Vol. 24, pp.147–164.

Liu, Y., Xun, C., Ward, R. et al. (2016) 'Image fusion with convolutional sparse representation', *IEEE Signal Processing Letters*, Vol. 23, No. 12, pp.1882–1886.

Ma, A., You, F., Jing, M. et al. (2020) 'Multi-source domain adaptation with graph embedding and adaptive label prediction', *Information Processing and Management*, Vol. 57, No. 2020, p.102367.

Naidu, V.P.S. (2011) 'Image fusion technique using multi-resolution singular value decomposition', *Defence Science Journal*, September, Vol. 61, No. 5, pp.479–484.

Niu, L., Cai, J. and Xu, D. (2016) 'Domain adaptive fisher vector for visual recognition', *European Conference on Computer Vision*, Vol. 9910, No. 2016, pp.550–566.

Oliver, R. (1990) 'Pixel-level image fusion and the image fusion toolbox', *Defense, Security, and Sensing*, December, Vol. 3374, No. 1998, pp.378–388.

Paul, S., Sevcenco, I.S. and Agathoklis, P. (2016) 'Multi-exposure and multi-focus image fusion in gradient domain', *Journal of Circuits Systems and Computers*, October, Vol. 25, No. 10, pp.1650123–1650123.

Reham, E., Osama, A., Mohamed, E. et al. (2023) 'Improving the efficiency of RMSProp optimizer by utilizing Nestrove in deep learning', *Scientific Reports*, Vol. 13, No. 1, pp.8814–8814.

Shuhui, J., Haiyi, M., Zhengming, D. et al. (2020) 'Deep decision tree transfer boosting', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 31, No. 2, pp.383–395.

Suhartono, N.S., Prastyo, D.D. et al. (2018) 'Design of experiment to optimize the architecture of deep learning for nonlinear time series forecasting', *Procedia Computer Science*, Vol. 144, pp.269–276.

Sun, H., Chen, X., Wang, L., Liang, D., Liu, N. and Zhou, H. (2020) 'C2DAN: an improved deep adaptation network with domain confusion and classifier adaptation', *Sensors*, Vol. 20, No. 12, p.3606.

Tong, K. and Wu, Y. (2023) 'Rethinking PASCAL-VOC and MS-COCO dataset for small object detection', *Journal of Visual Communication and Image Representation*, Vol. 93, No. 2023, p.110752.

Wan, L., Li, Y., Chen, K. et al. (2022) 'A novel deep convolution multi-adversarial domain adaptation model for rolling bearing fault diagnosis', *Measurement*, Vol. 191, No. 2022, p.110752.

Wang, C. et al. (2020) 'A novel multi-focus image fusion by combining simplified very deep convolutional networks and patch-based sequential reconstruction strategy', *Applied Soft Computing*, Vol. 91, No. 2020, pp.106253–106253.

Wang, P., Lu, L., Li, J. et al. (2019) 'Transfer learning with joint distribution adaptation and maximum margin criterion', *Journal of Physics: Conference Series*, Vol. 1169, No. 1, pp.012028–012028.

Wang, Q., Zhan, Z., Xie, X. et al. (2021) 'Globally adaptive neural network tracking for uncertain output-feedback systems', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 34, No. 2, pp.814–823.

Xu, H. et al. (2020) 'A deep model for multi-focus image fusion based on gradients and connected regions', *IEEE Access*, Vol. 8, No. 2020, pp.26316–26327.

Yang, P., Li, W., Wen, C. et al. (2024) 'Fault diagnosis method of multi-rotor UAV based on one-dimensional convolutional neural network with adaptive batch normalization algorithm', *Measurement Science and Technology*, Vol. 35, No. 2, pp.025102.

Yu, Y., Zhang, D., Ji, Z. et al. (2023) 'Balancing feature alignment and uniformity for few-shot classification', *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society*, No. 2023, pp.1–19.

Yun, J. and Lee, J-S. (2024) 'Learning from class-imbalanced data using misclassification-focusing generative adversarial networks', *Expert Systems with Applications*, Vol. 240, No. 2024, p.122288.

Zhang, Z., Chen, H., Li, S. et al. (2020) 'Unsupervised domain adaptation via enhanced transfer joint matching for bearing fault diagnosis', *Measurement*, Vol. 165, pp.10871–10887.

Zhang, Z., Ren, X., Yang, X. et al. (2023) 'Parametric chamfer alignment based on mesh deformation', *Measurement and Control*, Vol. 56, No. 12, pp.192–201.

Zheng, W., Zhou, X., Bai, C. et al. (2023) 'Adaptation of deep network in transfer learning for estimating state of health in electric vehicles during operation', *Batteries*, Vol. 9, No. 11, pp.547–562.

Zhou, B., Zhao, J., Yan, C. et al. (2023) 'Global and local knowledge distillation method for few-shot classification of electrical equipment', *Applied Sciences*, Vol. 13, No. 12, pp.7016–7088.

Zhu, Y., Zhuang, F. and Wang, D. (2019) 'Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources', *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33, No. 1, pp.5989–5996.

Zhu, Y., Zhuang, F., Wang, J. et al. (2022) 'Multi-representation adaptation network for cross-domain image classification', *Neural Networks*, Vol. 119, No. 2019, pp.214–221.

Zhuang, F., Qi, Z., Duan, K. et al. (2020) 'A comprehensive survey on transfer learning', *Proceedings of the IEEE*, Vol. 109, No. 1, pp.43–76.