Heritage Science

RESEARCH

Open Access

Prediction of broken areas in murals based on MLP-fused long-range semantics

Nanyu Hu[1], Hao Wu[1*] and Guowu Yuan[1]

Abstract

Predicting the broken areas in murals plays a key role in mural virtual restoration. Mural damage may arise for various reasons and the broken areas also vary greatly in terms of type. The existing methods, however, are limited to predicting a single type of damage and often struggle to identify the dispersed damage with accuracy. Moreover, these methods make it difficult to capture the global information in the broken areas for their insufficient understanding of contexts. To fully use the features at different scales, we propose a novel hierarchical multi-scale encoder-decoder framework termed as Mixer of Dual Attention and Convolution (DACMixer). With the introduction of an attention-convolution dual-branch module in the encoder, DACMixer can not only improve its ability to extract intricate features of small broken areas but also capture long-range dependencies of independent broken areas. Within DACMixer, the MFF (Multi-layer perceptron-based feature fusion) module integrates both local and global information in the broken areas, facilitating efficient and explicit modeling image hierarchies in the global and local range. Contrary to the encoder, DACMixer uses only lightweight multi-level decoder to decode the features of the broken masks, thus reducing the computational cost. Additionally, DACMixer preserves skip-connection to effectively integrate features from different levels of the MFF module. Furthermore, we provide a diversified mural dataset with elaborated broken annotation, which is named YMDA [YMDA denotes our dataset Yunnan_Murals_Dataset_Aug.], to further improve DACMixer's generalization ability to predict the broken areas. The experimental results demonstrate that DACMixer is capable of predicting the texture, edges, and details of the broken areas in murals with complex backgrounds. DACMixer outperforms the conventional methods with superb results: it achieves 78.3% broken areas IoU (Intersection over Union), 87.5% MIoU (Mean Intersection over Union), and 85.7% Dice coefficient.

Keywords Mask prediction, Mural digitization, Attention module, Local features, Long-distance modeling, Feature fusion

Introduction

Murals are valuable cultural heritage for their diverse themes, exquisite craftsmanship, and unique styles. Many murals, however, are susceptible to problems such as cracking, peeling, and fading owing to aging or vandalism. To locate these mural diseases accurately is an important step for mural restoration since art restorers must identify the damaged areas before taking further restoration actions. The manual identification of these diseases or damaged locations relies on the restorers' expertise in mural diseases, which is laborious and time-consuming. This challenge is further exacerbated when there are a large number of scattered and small broken areas.

Heritage digitization has provided new insights for the prediction of the broken areas in murals. Instead of relying solely on manual annotation of broken areas, the integration of digital technologies can effectively enhance the efficiency of damage localization. The conventional methods for mural breakage prediction [1–5] achieve

*Correspondence:
Hao Wu
haowu_sise@ynu.edu.cn
[1] School of Information Science and Engineering, Yunnan University, Kunming 650504, China

Springer Open

© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/. The Creative Commons Public Domain Dedication waiver (http://creativecommons.org/publicdomain/zero/1.0/) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

image segmentation by morphological operations, clustering, automatic threshold, etc. Bi et al. [1] applied a local distinction-based segmentation algorithm to effectively segment the broken areas of Thangka, considering the gray value, local complexity and local differences of the broken areas. Jaidilert et al. [2] proposed a semi-automatic detection method to segment the scratches on Thai murals using the region-growing method and morphological operation; however, this method requires the users to supply a small number of seed points, which complicates the prediction of the broken areas in murals. Zhang et al. [3] first used morphological open and closed operations to denoise the murals, and then resorted to local optimal hierarchical clustering with weighted average as similarity measurement to extract the mural disease information to form the broken areas masks. While the clustering process requires experts to determine the threshold for terminating the clustering, which increases manual intervention. Deng et al [4]. studied the temple murals of the Ming Dynasty in Zhilin Temple, and proposed a crack and flaking deterioration correction algorithm for ancient murals based on multidimensional gradient detection, guided filtering and tensor voting, which calibrates the mural cracks and stratum spalling regions to obtain masks. Nevertheless, the algorithm is ill-adapted to various mural deterioration for its poor learning capacity.

Recently, deep learning has also been used to predict the broken areas in murals. Numerous experiments have demonstrated that during the training process, different network models, supported by quantities of data, can extract complicated features such as texture, contour, color, edge, and structural information, resulting in more precise, and robust localization of the broken areas in murals. Cao et al. [7] incorporated the lightweight neural network MobileNetv2 [8] into PSPNet [9] to segment ancient murals, preserving the semantic details while minimizing the number of network parameters, but it merely segmented the mural patterns. Lin et al. [10] adopted Minimum Noise Fraction (MNF) rotation to reduce the dimension of the hyperspectral images, selected feature vectors as the input of the back propagation (BP) neural network, and trained the BP neural network to categorize the mural images into damaged and normal regions. Yuan et al. [11] enhanced the UNet architecture by incorporating ResNet-50 as the encoder network while preserving the skip connections of UNet to fuse features from different levels. Additionally, they employed various loss functions during training to accurately segment craquelure and paint loss on polychrome paintings in the Palace Museum. For their ability to model long-term dependencies, attentional mechanisms have been used for broken area prediction. To address mural crack segmentation, this paper introduces TMCrack-Net [12], a U-shaped network with feature pyramids and Transformer. Instead of utilizing the skip-connections of U-Net, we use an AG-BiFPN network comprising two modules: a channel cross-fusion (CCT) module with a transformer and a bidirectional feature pyramid network.

It is a challenge to accurately predict the broken areas in murals. The first difficulty lies in the murals' complex background structures and diverse content, coupled with the use of various painting techniques. Secondly, the broken areas in murals could be the result of various factors, and a lack of substantial information is a common occurrence. Lastly, the scarcity of mural samples, coupled with the significant domain gap between the feature space of the natural image[1] and the feature space of the mural, further complicates the prediction task. DACMixer is main intended to predict the broken areas in murals distinguished by phenomena like cracking, peeling, and fading. Figure 1 shows a mural image and the mask predicted using the proposed algorithm.

In response to these challenges, this paper proposes DACMixer, a novel cascaded Encoder-Decoder architecture that simultaneously incorporates an attention-convolution dual-branch module in the encoder. Moreover, the MFF module in DACMixer further enhances the interaction between the local and global features within the dual branches. By incorporating dual-branch and MFF modules, DACMixer achieves local and long-range features extraction and integration, enhancing the model's capacity to capture relevant information across different scales and levels of details. Similar staged upsampling operations are used in the decoder. In DACMixer, we retain skip-connection, which integrates the interaction information from the MFF module in the encoder into the upsampling module. Again, this paper alleviates the problem of low availability of annotated murals for the training network. We collect a diversified mural dataset named YMDA, comprising 7282 images cropped from high-resolution original mural images. Each image in YMDA is annotated with a binary mask (each pixel is labeled as broken or non-broken) by clustering and careful manual refinement. To the best of our knowledge, YMDA is the first mural dataset that exceeds the previous efforts in both annotation complexity and diversity.

Figure 2 shows the reconstructive effects of different masks in the mural in virtual restoration. It can be seen from Fig. 2 that the first row, a mask generated randomly by the traditional method does not fully correspond to

[1] Natural image: In this paper, "natural image" refers to the image sample derived from the real world. These images typically encompass a wide range of everyday scenes, objects, and individuals, such as street views, buildings, faces, animals,etc.

Fig. 1 An example of broken murals (Left), A mask predicted using our algorithm (Right)

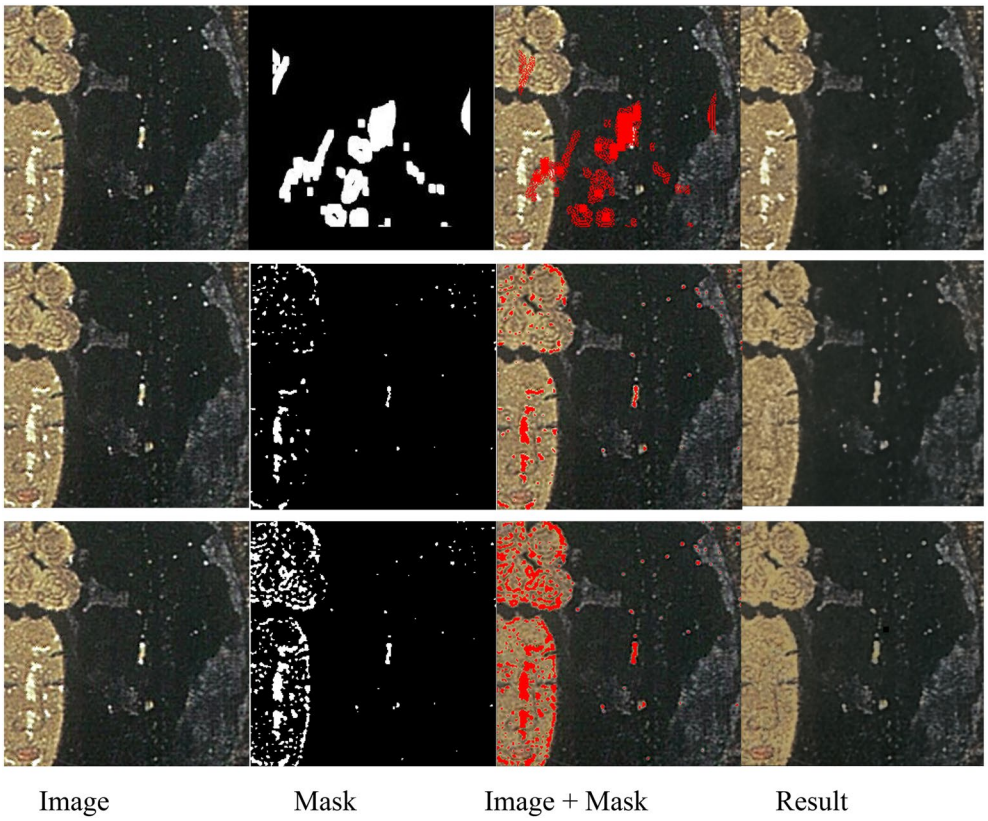


Fig. 2 Restoration results based on prediction masks

the broken areas in the actual mural, which suggests that the restoration result is inferior. While both the second and the third rows are more accurate actual masks which are obtained by deep learning algorithm with the support of a large amount of data and powerful arithmetic by extracting features from different layers. They help to recover large areas of missing images and long-distance structural information. The best visual restoration effect is found in the third row that shows the mask predicted by the proposed algorithm.

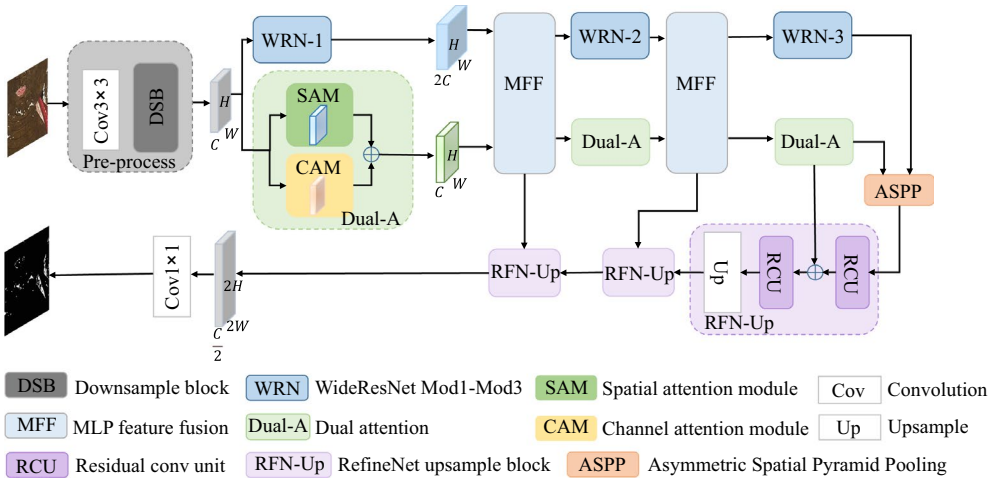


Fig. 3 Architecture overview of the DACMixer

Methods

Overview of the structure

DACMixer is a neural network with U-Net like architecture and the U-net is widely used for tasks such as Image Inpainting [13–15], Object Detection [16], and Image Segmentation [17–20]. DACMixer comprises three primary modules: (1) The encoder, which is composed of a dual-attention branch and a convolution branch, facilitates the top-down extraction of features related to the broken areas. The MMF module is employed to integrate features obtained at various scales by the two branches. (2) The bottleneck layer incorporates the ASPP (asymmetric spatial pyramid pooling) structure, enabling the network to capture information across multiple receptive field sizes. (3) The decoder integrates skip-connection to merge features extracted from different levels in the encoder with the features in the decoder, promoting effective feature fusion and reconstruction. An overview of the complete architecture is shown in Fig. 3. Considering the loss of image details and the overall computation, the final feature map of the encoder is set to 1/4 of the original image. After that, the encoder staged feature representations are gradually combined into full resolution predictions using convolutional decoder.

It is worth mentioning that before feeding the image into the two-branch structure, DACMixer uses a preprocessing module, and more specifically, it uses a 3×3 convolution to project the mapping of the input features as well as a downsampling module that reduces the original image resolution to 1/2, resulting in a rich set of intermediate features. We add a task-specific output head at the end of the model to generate the final prediction.

The structure of the dual attention module in the encoder is shown in Fig. 4.

Dual branch fusion encoder

Parallelized feature representation

DACMixer attempts to explore global contextual information by establishing relation between features and attention mechanisms in the encoder. The method can adaptively aggregate long-range contextual information, while using convolution for local feature extraction in order to preserve low-level semantic information such as color and shape, thus allowing better preservation of detailed information and more accurate pixel-level prediction results.

Inspired by DANet [21], the attention branch of DACMixer consists of a spatial attention module and a channel attention module, which is denoted as $D_{\phi}(I)$,with ϕ denoting learnable parameters, and $I \in \mathbb{R}^{C\times H\times W}$ the output of the preprocessing module.

The specific formula is as follows.

Spatial attention module:

$$L = \text{Transpose}(\textit{Reshape}(B)) \times \text{Reshape}(C) \tag{1}$$

$$M = \alpha[\text{Reshape}(D) \times \text{softmax}(L)] \tag{2}$$

$$E = \textit{Reshape}(M) + A \tag{3}$$

where some of the operations are denoted as:

$$\textit{Reshape} : \mathbb{R}^{C\times H\times W} \rightleftarrows \mathbb{R}^{C\times N} \tag{4}$$

$$\text{Transpose} : \mathbb{R}^{C\times N} \rightarrow \mathbb{R}^{N\times C} \tag{5}$$

From the output feature map $A(A \in \mathbb{R}^{C\times H\times W})$ of the preprocessing module, the corresponding feature maps B ,C,D are generated by three convolutional layers and they

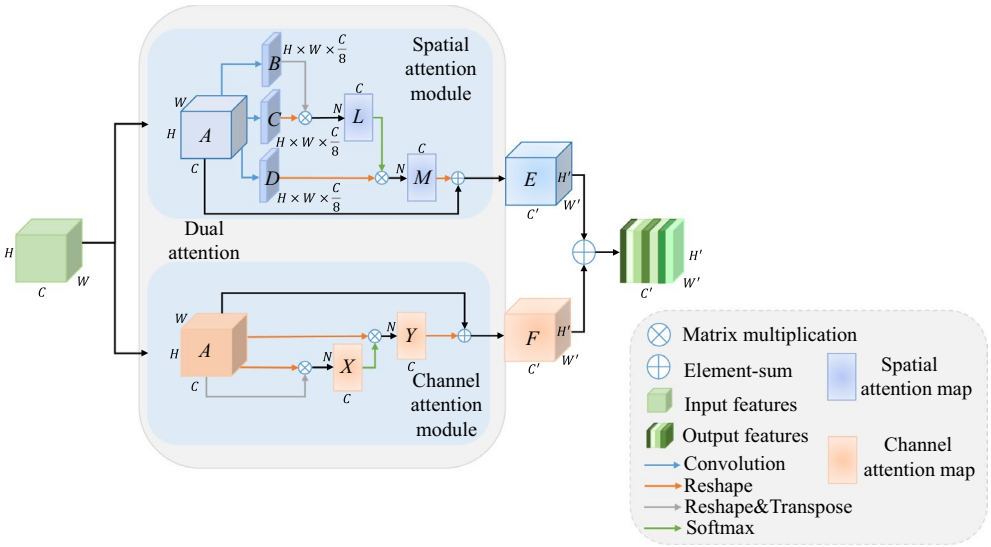


Fig. 4 The structure of the dual attention

are all reshaped, where $N = H \times W$. First,B is transposed, and the matrix is then multiplied with C to generate the spatial attention matrix $L(L \in \mathbb{R}^{N \times N})$, which can model the spatial relationship between any two pixels of the features. Next, the attention matrix L is subjected to a softmax operation, and the resulting matrix is multiplied with the original feature matrix D. This product is then multiplied by the scale factor α to obtain matrix $M(M \in \mathbb{R}^{C \times N})$. Finally, M is reshaped back to its original size, and is summed with the original feature A at element level, resulting in the final representation E that reflects the long-range contextual information, where α is initialized to 0 and gradually learns to get larger weights.

The same procedure is obtainable for the calculation of channel attention module.

$$X = Transpose(Reshape(A)) \times \text{Reshape}(A) \tag{6}$$

$$Y = \beta[Reshape(A) \times \text{softmax}(X)] \tag{7}$$

$$F = Reshape(Y) + A \tag{8}$$

where β is initialized to 0 and gradually learns to get larger weights.

Finally, the output of the two attentions is summed to obtain the output of the attention module d.

First, a self-attention mechanism is introduced in the spatial attention module to selectively aggregate features at each location by weighting the sum of features at all locations. Similar features will correlate with each other and improve each other regardless of the distance in the spatial dimension. Secondly, each channel of the feature implies a piece of corresponding information (e.g. color, texture, etc.), and a similar self-attention mechanism is used for the channel attention module. Selectively emphasize the existence of interdependent channels by integrating the relevant features among all channel features, and update each channel feature using the weighted sum of all channel features. Such improvements can enhance both the perceived contextual information and the representational capability in the encoder.

The convolution branch is denoted as $\mathcal{R}_\theta(I)$, where θ is the parameter, and $I \in \mathbb{R}^{C \times H \times W}$ is the output from the preprocessing module. This branch uses modified WideResNet [22] as the backbone, which is divided into three stages with a block distribution of {3, 6, 3}. The convolution calculation is usually performed using a filter with kernel size of 3×3 and stride of 1. To increase the receptive field of convolution while keeping the resolution of image resolution or coverage, we also use dilated convolution with a dilation factor of 2 in the last stage of the convolution branch.

The WideResNet used in this paper consists of four types of residual blocks that are stacked repeatedly. The first and second types are regular residual blocks, which consist of 3×3 convolution, BN (Batch Normalization) layer, and employ the ReLU activation function. The third and fourth types of residual blocks are built upon the foundation of the regular residual blocks and incorporate dilated convolutions with a dilation factor of 2. A 1×1 convolution on the shortcut connection is used to adjust the channel dimension. Figure 5 illustrates the structures of these four distinct types of residual blocks.

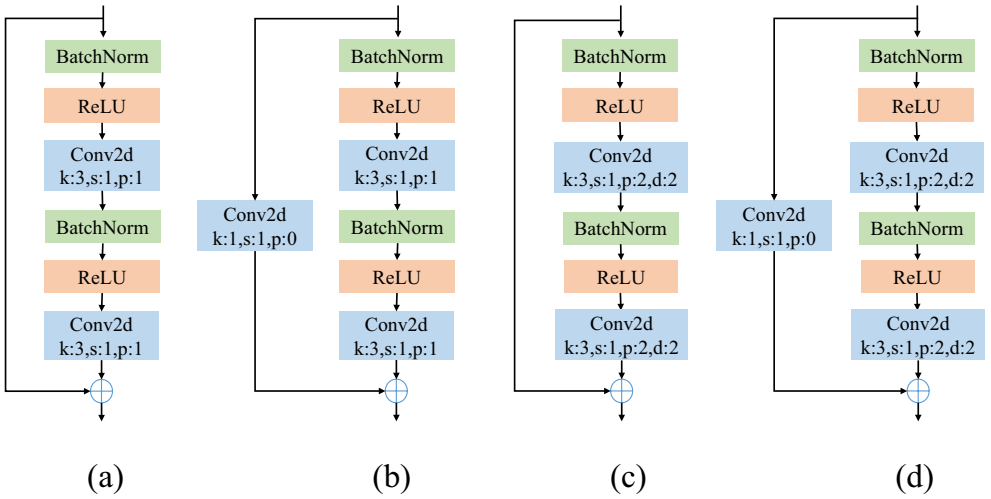


Fig. 5 Structure of WideResNet residual blocks. **a** Structure of the regular residual block. **b** Structure of the regular residual block with 1 × 1 convolution on the shortcut connection. **c** Structure of the residual block with dilated convolutions. **d** Structure of the residual block with dilated convolutions and 1 × 1 convolution on the shortcut connection, where k represents the convolution kernel size, s represents the stride, p represents the padding, and d represents the dilation factor

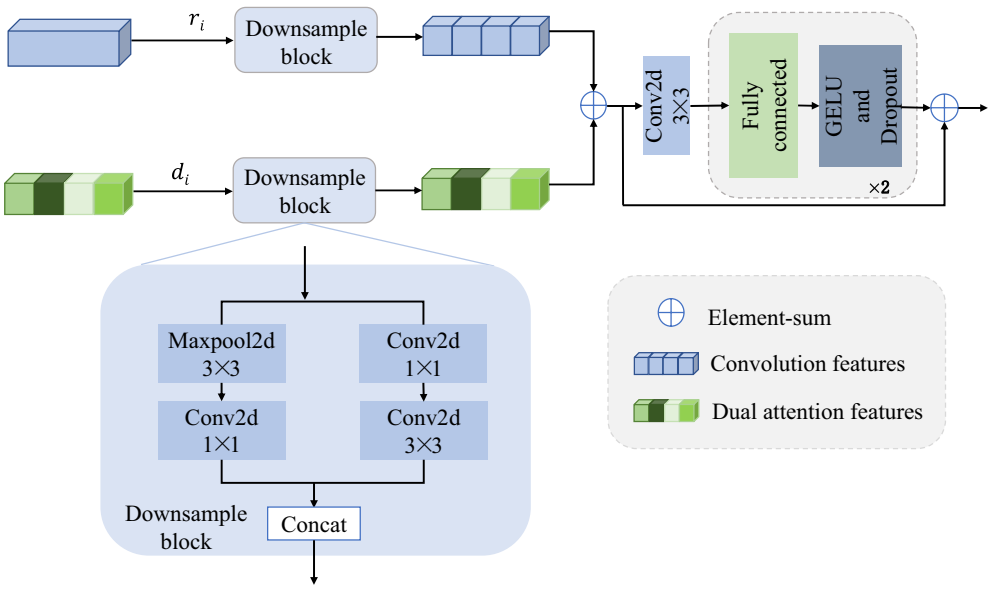


Fig. 6 The details of the MLP Feature Fusion Module

Notably, for WideResNet in the last stage of output, an asymmetric spatial pyramid pooling [6] is used to capture richer contextual information by aggregating features at different resolutions. Finally, the outputs of the two branches of the encoder are concatenated to form the final enhanced feature representation.

Feature fusion module

The feature fusion module uses an MLP structure to embed channel dependencies from the dual-attention module into convolution branches, allowing for better extraction of local features. Meanwhile, spatial dependencies can refine the features more effectively. The effective integration of the attention mechanism with convolution enhances the encoder's capacity to express features. The feature fusion module is shown in Fig. 6.

The module is denoted as $\mathcal{F}$, where W is the weight of the fully connected layer, and it uses the MLP structure to fuse the output d_i from the dual-attention branch and the output r_i from the convolution branch.

The input features of the attention branch and convolution branch are first downsampled. In the downsample block, we use both max pooling and convolution to reduce the size of the feature map, which increases the invariance of the network to translations and rotations and better aggregates the features. Then the attention branch and convolution branch are summed pixel-wise before entering the linear projection layer, which maps C (the second fusion module dimension: 2C) dimensions to C (the second fusion module dimension: 2C) dimensions. This is done to fuse the long-range dependent and local information to facilitate the interaction between the two dimensions, and then assign weights to the inputs before activating them using a nonlinear function, with the residual structure used for the final connection. We set C as 256 dimensions in this paper unless otherwise stated.

$$\mathcal{F}_{\text{in}} = \text{downsample}(D_\phi) + \text{downsample}(\mathcal{R}_\theta) \tag{9}$$

$$\mathcal{F}_{\text{MLP}} = X_i + \sigma(WX_i)(\text{for } i = 1...S) \tag{10}$$

$$\mathcal{F}_{\text{out}} = \mathcal{F}_{\text{in}} + \mathcal{F}_{\text{MLP}} \tag{11}$$

where σ denotes the activation function GELU [23],S denotes the number of adjustable hidden layer nodes in the MLP, and the hidden layer width is chosen to be half of the input nodes for fusing spatial and channel semantic information, $\mathcal{F}_{\text{in}}$ denotes the feature map generated by summing the convolution branch and dual attention branch after the downsample block,$\mathcal{F}_{\text{MLP}}$ denotes the new features obtained by linear projection,$\mathcal{F}_{\text{out}}$ denotes the result of the residual connection.

In the DAMixer, after obtaining the long-range semantic information from the attention branch and the local features from the convolution branch, they are simultaneously fed into the MLP feature fusion module for interaction to maintain multi-scale contextual information. The module combines the global and local features and outputs more refined results.

The fused semantic information can be modeled on both local features and global features simultaneously, which helps improve the segmentation accuracy. Specifically speaking, for the segmentation prediction of K semantic classes, the fused semantic information outputs a categorical distribution, which represents a probability that a pixel belongs to each of the K classes.

$$f = p(y|d,r) = F(d,r) \in \mathbb{R}^{K\times H\times W} \tag{12}$$

d has three stages of outputs: $d_1 \in \mathbb{R}^{C\times H\times W}$,$d_2 \in \mathbb{R}^{2C\times \frac{H}{2}\times \frac{W}{2}}$,$d_3 \in \mathbb{R}^{4C\times \frac{H}{2}\times \frac{W}{2}}$

Similarly,r has three stages of outputs: $r_1 \in \mathbb{R}^{2C\times H\times W}$,$r_2 \in \mathbb{R}^{4C\times \frac{H}{2}\times \frac{W}{2}}$,$r_3 \in \mathbb{R}^{8C\times \frac{H}{2}\times \frac{W}{2}}$.

Refining the features in the decoder

The DACMixer's decoder uses a staged structure to propose a simple three-stage recombination operation that gradually fuses feature profiles from different levels of the encoder into a final pixel-level prediction. The coarse features of the pre-encoder stage help the decoder to recover some information on object segmentation details.

The final feature representation from the encoder first enters the residual module of the first upsampling module to refine the features using convolution. Next, an element-by-element summation is performed with the shallow features before they are fed into the next residual module for aggregation. The shallow features are subjected to a channel-down operation before summation because the corresponding low-level features usually contain a large number of channels, which may outweigh the importance of fine features while making training more difficult. The summed feature matrix is then fed into the upsampling module in the next stage. The DACMixer decoder embeds the corresponding resolution features for recombination from the encoder feature fusion modules 1, 2, and 3 in three different stages. Using the RefineNet-based convolution module [24], the shallow features extracted from the successive stages in the encoder are combined with the deep features and progressively upsampled by a factor of 2 in the first two fusion modules. The feature representation size is eventually restored to full resolution.

The prediction concludes with DACMixer using a 1×1 convolution to specify the number of classes to be segmented.

Total loss

Due to the small percentage of pixels in the foreground of the murals, the learnable information is extremely limited, and the data categories of the murals are disproportionately distributed. To alleviate the class-imbalance problem, we optimize the learning of image details by jointly adopting cross-entropy loss and Dice loss.

The overall loss function of DACMixer is set to:

$$\mathcal{L}_{\text{total}}{}^{\theta,\phi,\gamma} = \mathcal{L}_{\text{CE}}{}^{\theta,\phi,\gamma}(y,\hat{y}) + \mathcal{L}_{\text{Dice}}{}^{\theta,\phi,\gamma}(y,\hat{y}) \tag{13}$$

The overall loss function $\mathcal{L}_{\text{total}}{}^{\theta,\phi,\gamma}$ consists of the standard cross-entropy loss function $\mathcal{L}_{\text{total}}{}^{\theta,\phi,\gamma}$ and

the Dice loss function $\mathcal{L}_{\text{total}}{}^{\theta,\phi,\gamma}$, where θ is the parameter of the convolution branch in the encoder, ϕ is the parameter of the attention branch in the encoder, and γ is the parameter of the fusion module. $y \in \mathbb{R}^{H\times W}$ is the final pixel-level prediction result, and $\hat{y} \in \mathbb{R}^{H\times W}$ is the corresponding segmentation ground-truth. The overall loss is the sum of cross-entropye loss, and the loss ratio of foreground and background is set to [2.0:1.0].

Experiments

Dataset and experimental settings

Dataset

We conduct the validation experiments on the proposed dataset YMDA. As a dataset for predicting the broken areas in murals, YMDA consists of 7282 finely annotated images which come in two categories: broken and non-broken. Its rich semantic annotations cover various types of murals in China's Yunnan province and the dataset falls into two folders, images and masks. The image folder contains three subfolders, namely train, val, and test, which are the original images of the broken ones. The mask folder also contains three subfolders, namely train, val, and test, which are the segmentation ground truth corresponding to each original image. To ensure image variability for each dataset, a random data selection for the broken images is performed in each style of the original mural images, with 80% used as the training set, 10% as the validation set, and 10% as the test set. The basic selection criteria include: mural style, degree of damage, and image theme.

We obtained a total of 55 full-scale mural images for this purpose. Seven out of these 55 mural images were scanned using a handheld color optical scanner, the EinScan Pro 2X, with a spatial point distance of 0.2 mm to scan the murals' surfaces. The resulting scans were then subjected to projection. For the remaining 48 murals, we captured their images using a Nikon D850 camera with a DPI of 300. The images captured were then treated with orientation correction and distortion correction. Additionally, we manually adjusted the brightness and contrast of the mural data to achieve consistency as much as possible.

The steps for creating YMDA are detailed as follows. The first step is to prepare data. Since murals are subject to inevitable consequences like breakage, fading and discoloring, operations such as denoising, smoothing and equalization are necessary for improving the image quality. The mural images varied in size, with the lowest resolution of the scanned images being 900×870 and the highest resolution being 5042×4451. These mural images were cropped to the same size, and then a total of 7282 images were manually selected for intensive pixel-level annotation so as to achieve a high degree of diversity in foreground objects, backgrounds, and overall scene layout.

The second step is annotation protocol. Firstly, a division-based clustering algorithm is used to divide the area for pixel-wise classification of images and generate coarse labeled images from the processed images, as shown in Fig. 7b. The lack of performance and accuracy in the clustering algorithm makes it difficult to segment the broken details. Manual reprocessing, therefore, is performed on the initially annotated images after clustering. Specifically, the original image is used as a layer and the roughly annotated image is superimposed on the original image as another mask layer. By changing the transparency and observing the difference between the two, the brush tool is used to make manual pixel-wise adjustments again, resulting in high-quality annotation results. On average it takes more than 20 min for each image to complete annotation and quality control. The complete uncropped original image is provided for contextual information reference during the annotation process. Figure 7(c) presents the manual precise labels. The above approach allows annotations to be readily extended to cover additional or precise classes later.

In summary, YMDA is a carefully designed dataset in terms of annotation density and complexity, image variety richness, and it provides a reference for future in-depth analysis of mural characteristics.

In this paper, we predict the background and non-broken areas of mural images as 0, representing "background or non-broken pixels," which are visualized as black in the images. Conversely, we predict the broken areas of mural images as 1, representing " broken pixels," which are visualized as white in the images.

A comparison of coarse and precise labels is shown below. We have indicated the manually refined correction regions in the precise labels (red rectangles in Fig. 7c).

The manual refinement is involved to correct the background, clear the edges, and redraw the outline.

Implementation details

We implemented DACMixer using PyTorch on an NVIDIA GeForce RTX 3090 GPU card with 24 GB of RAM. For weight initialization, we adopted the method proposed by He et al. [35], which involves random initialization at the beginning of the learning process. The training process took one day and thirteen hours. We applied data augmentation methods such as rotation, random horizontal flip with 0.5 probability to the dataset YMDA and normalization of the input using mean (0.485, 0.456, 0.406) and standard deviation (0.229, 0.224, 0.225). The model was trained and evaluated on the dataset with the original image resolution set to 256×256 and

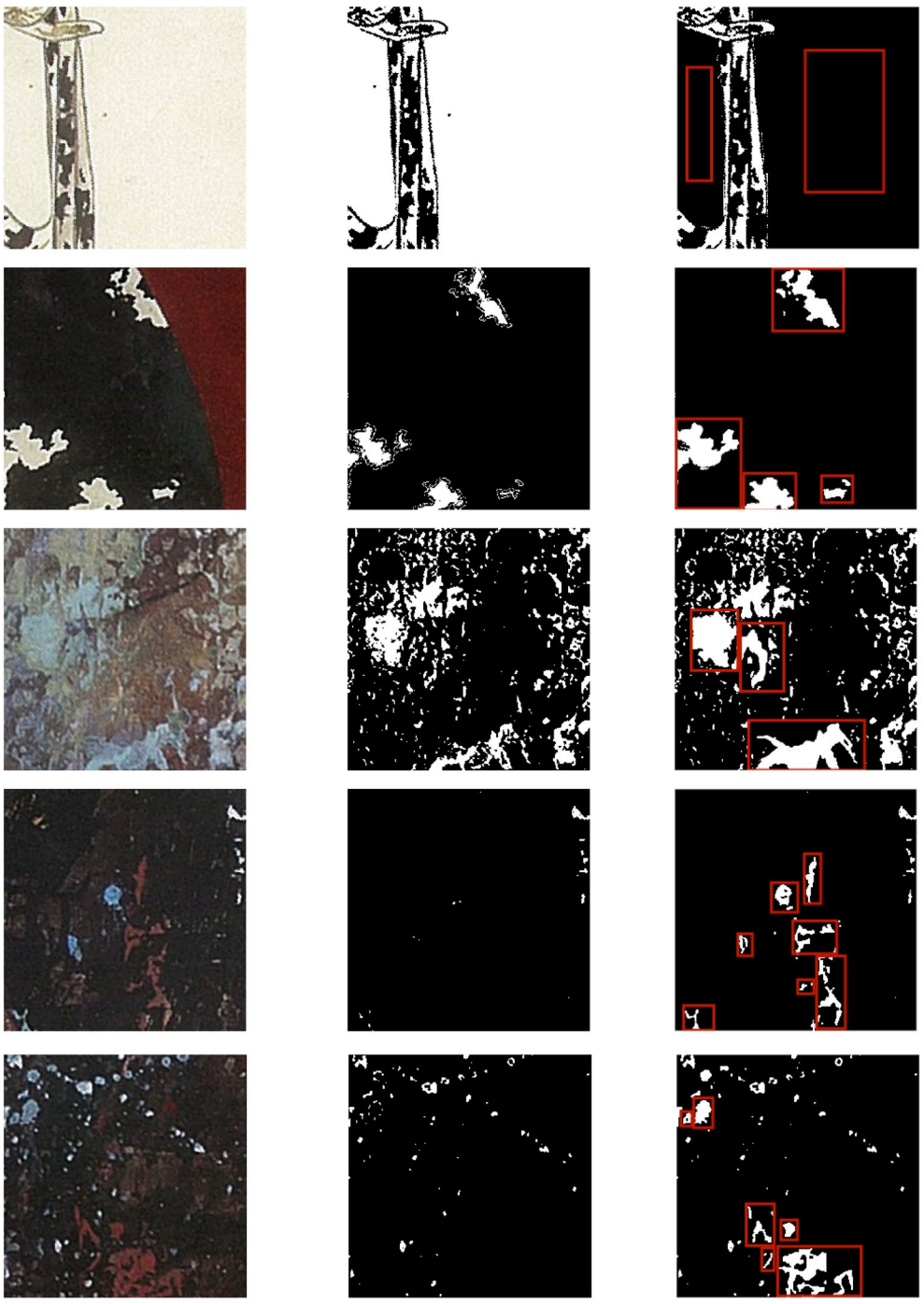


Fig. 7 Comparison of coarse and precise labels

the batch size set to 2. The model was trained on YMDA for 150 epochs using an SGD optimizer with momentum coefficient of 0.9 and decay coefficient set to 1e-4. Following the training strategy of Deeplabv2 [6], we used a "poly" learning policy with the initial learning rate set to 0.0001, and also improved the learning rate with a warm-up policy. In the ablation studies, we trained the model for 150 epochs. In the evaluation process, we cropped

Table 1 Ablation experiment of the Feature Fusion module

Evaluation Metrics	None fusion	C-Fusion	MLP-fusion
Broken.IoU[%]	69.6	74.6	78.3
MIoU[%]	82.4	85.5	87.5
Dice coefficient[%]	80.4	83.8	85.7

Table 2 Ablation experiment for skip-connection

Evaluation Metrics	None-connection	Fusion-connection
Broken.IoU[%]	69.9	78.3
MIoU[%]	82.5	87.5
Dice coefficient[%]	82.6	85.7

the image size to 256×256 as well and used three semantic segmentation evaluation metrics, namely Intersection over Union (IoU) [25] and Mean Intersection over Union (MIoU) and Dice coefficient, to compare the performance of different models. IoU, MIoU, and Dice coefficient are defined as follows:

$$IoU = \frac{TP}{TP + FP + FN} \tag{14}$$

$$MIoU = \frac{1}{n_{cls}} \text{sum}(IoU) \tag{15}$$

$$Dice\ coefficient = \frac{2TP}{2TP + FP + FN} \tag{16}$$

where n_{cls} means there are n classes in total, TP (true positive) means that broken areas in the ground truth are correctly identified as broken pixels, FP (false positive) means that non-broken areas in the ground truth are mistakenly predicted as broken areas, FN (false negative) means that broken areas in the ground truth are incorrectly recognized as non-broken areas. The Intersection over Union of the broken category is specifically reported in the presentation.

Experimental results

Ablation studies

We conducted ablation experiment to investigate the different effects of fusing global and local features using pure convolution and using MLP structures.

In Table 1, C-Fusion denotes the fusion of global and local features using pure convolution, and similarly, MLP-fusion denotes the fusion of global and local features using fully connected layers. It can be seen from the table that the MLP structure outperforms the pure convolutional structure in fusion capability. In terms of IoU in the broken areas, the relative performance improves by 3.7%, MIoU by 2%, and Dice coefficient by 1.9%. This also confirms that using the fully connected operation helps the model to better handle long-range dependencies and allows the model to have a lower inductive bias, making it dependent only on the original data for learning.

Meanwhile, we conducted ablation experiment to explore whether the interaction information in the fusion module is helpful to the reconstruction of mask details.

In Table 2, Fusion-connection signifies that the feature maps from the fusion module pass through skip-connection into the upsampling module of Decoder. A comparison between the first column and the second column of the table reveals that the latter achieves an improvement of 8.4%, 5%, and 3.1%, respectively, with better feedback. The visualization results show that the information interacted by the fusion module facilitates the image's recovery of more precise outcomes during the upsampling process, consequently displaying more distinct edges and contours (Fig. 8).

Comparison of the proposed method and the state-of-the-art methods

After DACMixer is trained for 150 epochs on the YMDA dataset, it is validated on the val set. Table 3 compares the performance of DACMixer and models that have performed well in semantic segmentation in recent years on the val set, where none indicates that no backbone network is used. To ensure fairness, the comparison models use the same training and validation scheme as DACMixer.

As can be seen from the table, DACMixer compares favorably with most of the previous baselines using convolution or attention mechanisms, and DACMixer yields 78.3% results for IoU of the broken areas, 87.5% for MIoU, and 85.7% for Dice coefficient. Compared with the better performing GSCNN model, DACMixer brings 5.2% improvement in IoU of the broken areas, 3% improvement in MIoU, and 4.3% improvement in Dice coefficient. Compared with DPT using the Vit backbone, DACMixer also produces very competitive results with 10.7% relative improvement in IoU of the broken areas, 6.2% relative improvement in MIoU, and 5% relative improvement in Dice coefficient. This proves that DACMixer, by fusing long-distance dependencies with the local features, performs slightly better than other models in predicting small, discrete broken areas in murals. Simultaneous modeling of the local and global features allows DACMixer to better capture local detailed

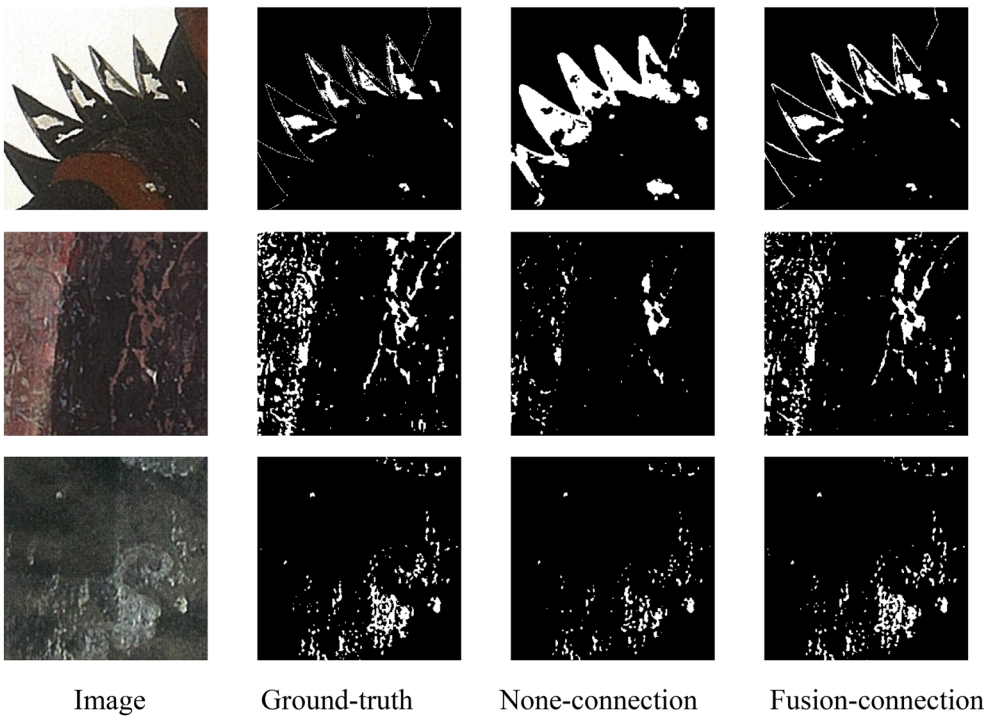


Fig. 8 Visualization results of the skip-connection ablation experiment

Table 3 Comparison of DACMixer with state-of-the-art methods on YMDA test set

Model	Backbone	Broken. IoU[%]	MIoU[%]	Dice coefficient[%]
GSCNN [26]	Wide-ResNet-38 [22]	73.1	84.5	81.4
DANet [21]	ResNet-50 [32]	61.3	77.6	76.0
HRNetv2 [27]	none	66.7	80.8	80.0
OCRNet [28]	HRNetv2-W18 [27]	67.0	81.1	80.3
GCNet [29]	ResNet-50 [32]	62.4	78.1	76.8
Res-Unet [11]	ResNet-50 [32]	66.8	80.7	76.8
TMCrack-Net [12]	ConvNext-S [34]	71.2	81.8	78.2
STDC1 [30]	STDC1 [30]	56.8	75.1	72.4
DPT [31]	Vit-Base [33]	67.6	81.3	80.7
Ours	Wide-ResNet-38 [22]	**78.3**	**87.5**	**85.7**

The bold value in the table represents the best values

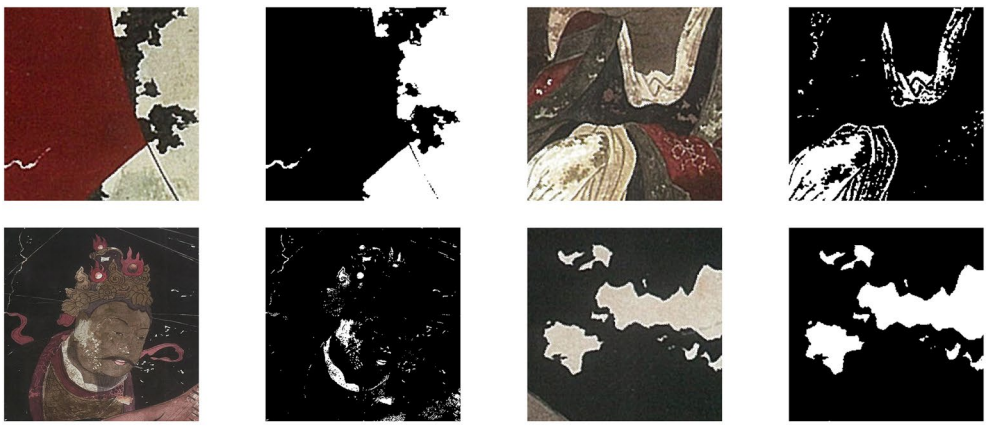
information and global consistency of images, and achieve the best performance by combining richer local and global contexts.

Visualization results

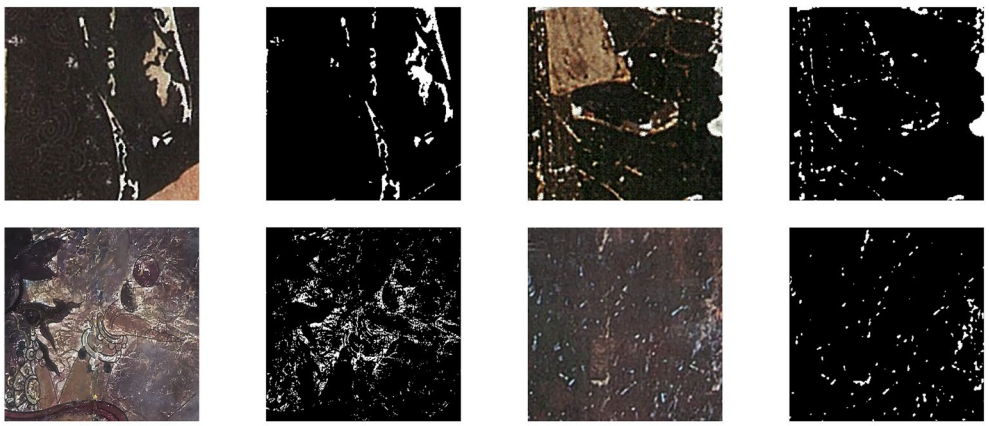
Below we show some of the DACMixer results on the YMDA test set from three perspectives: size, texture and distribution of the broken areas. It can be seen that DACMixer generates better masks for various types of broken areas (Fig. 9).

Qualitative results

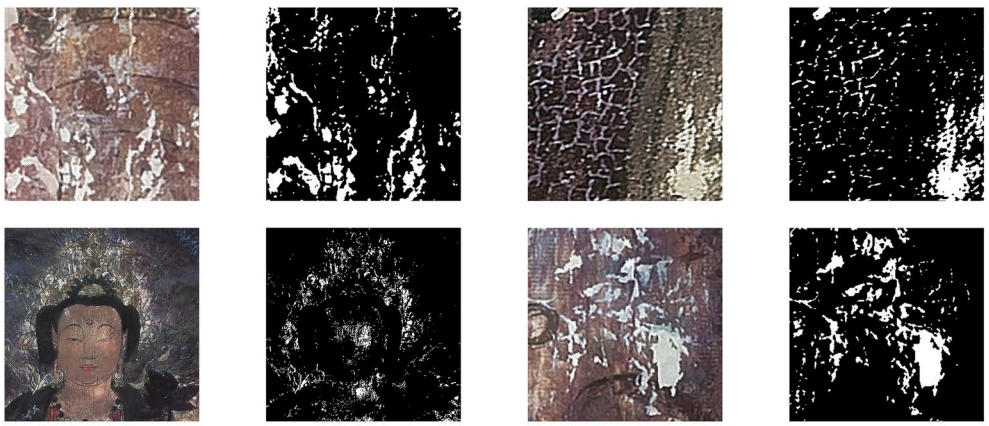
Figure 10 shows qualitative results on YMDA. As can be seen from the comparison of the visualization results of the first and second images, only DACMixer predicts both white and red broken areas in the case of a cluttered and mixed background, while all other models, without exception, only respond to the white broken areas. The results of the third, fourth and fifth rows indicate that DACMixer predicts more complete boundaries, and its performance on broken edges without additional



(a) Examples of large broken areas.



(b) Examples of discrete broken areas.



(c) Examples of textured broken areas.

Image Prediction Image Prediction

Fig. 9 Prediction results of DACMixer on YMDA test set

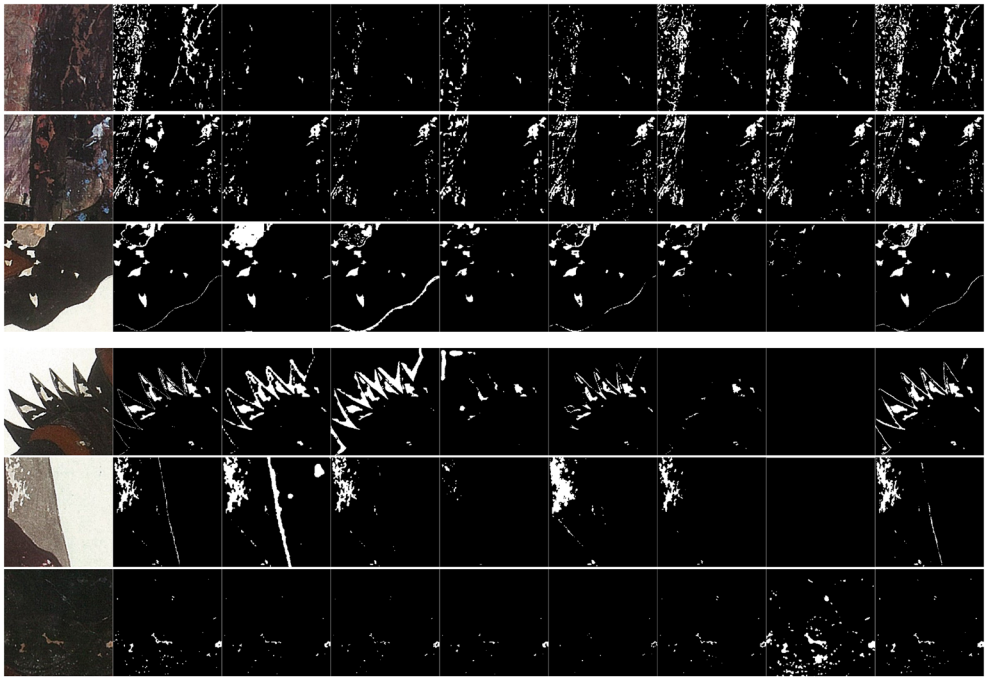


Fig. 10 Qualitative comparison of prediction results

Table 4 The Params and FLOPs of methods

Model	Params (M)	Flops (GMac)	Broken.IoU[%]	MIoU[%]	Dice coefficient[%]
GSCNN	137.28	188.92	73.1	84.5	81.4
UNet + +	47.19	200.13	68.9	82.0	79.0
DANet	46.72	125.23	61.3	77.6	76.0
DPT	124.0	51.39	67.6	81.3	80.7
Ours	45.0	187.33	78.3	87.5	85.7

boundary information is comparable to that of GSCNN, while it performs more detailed segmentation for different broken contours than DANet, HRNetv2 and DPT. For the last image with discrete and small broken areas, DACMixer can also accurately locate the broken areas, and it outstrips GCNet and OCRNet in overall segmentation performance. In summary, DACMixer achieves accurate, and clear pixel-level classification of broken areas, whether they are densely, sparsely, or otherwise distributed.

Model complexity

We used the number of floating point operations (FLOPs) to measure the computational complexity of the different models; the smaller the FLOPS, the smaller the model's demand for computation. Params are used to measure the spatial complexity of different models, and the parameter size indicates the memory size occupied by the model, and the results are shown in Table 4.

Table 4 presents the FLOPS and PARAMS of various pure convolutional models and attention-based models, all executed on the same GPU. Specifically, in Table 4, DPT exhibits the lowest computational complexity, but it requires more memory to store a large number of parameters. GSCNN involves significantly more parameters compared to DACMixer at a similar level of computational complexity. The proposed method may not be optimal in computational efficiency, but this drawback, to some extent, is compensated by its excellent prediction results.

Table 5 The Params and FLOPs of modules in DACMixer

Module	Params (M)	Flops (GMac)
Encoder	35.37	142.62
Bottleneck layer	8.52	32.77
Decoder	1.11	11.94

Meanwhile, we utilize FLOPS and PARAMS to measure the computational complexity and spatial complexity of the encoder, bottleneck layer, and decoder in DACMixer.

Table 5 indicates that the decoder only requires 11.94 Gmacs, exhibiting a lower computational complexity compared to the encoder. This is because DACMixer exclusively employs convolutional operations for upsampling in the encoder.

Discussion

The broken areas of murals contain both global and local feature hierarchies, which can be effectively predicted by DAMixer. Local regions, such as small cracks (red box in Fig. 11a), covering dozens of pixels can be effectively modeled using convolution. However, for the same type of the broken area, global-spanning features can also occur (green box in Fig. 11a). In this case, the context information and representational capability perceived by the convolution are insufficient. Additionally, murals exhibit various global features, including multi-scale texture similarity (red and green boxes in Fig. 11b), symmetrical broken areas (green and blue box in Fig. 11b), and content structural similarity (Fig. 11b). For global features, the attention mechanism can generate attention weights through correlation calculations to achieve global modeling and effectively capture long-distance dependencies. Inspired by this, we consider combining the local feature extraction ability of convolution with the long-distance modeling ability of attention mechanisms to predict the broken areas in murals. Current fusion techniques face two main challenges: The transformer structures, which are computationally expensive, only accept relatively small patches, and using CNN structures for fusion can lose non-locality for they are limited by a finite receptive field. We, therefore, choose the MLP structure as the fusion module, which has a global receptive field with no harsh requirements for input shapes, and the structure also controls fusion in both complexity and computational cost.

Despite its effectiveness in predicting the broken areas in murals, there is still room for progress in DACMixer's performance in predicting broken areas where the fading phenomenon is more prominent.

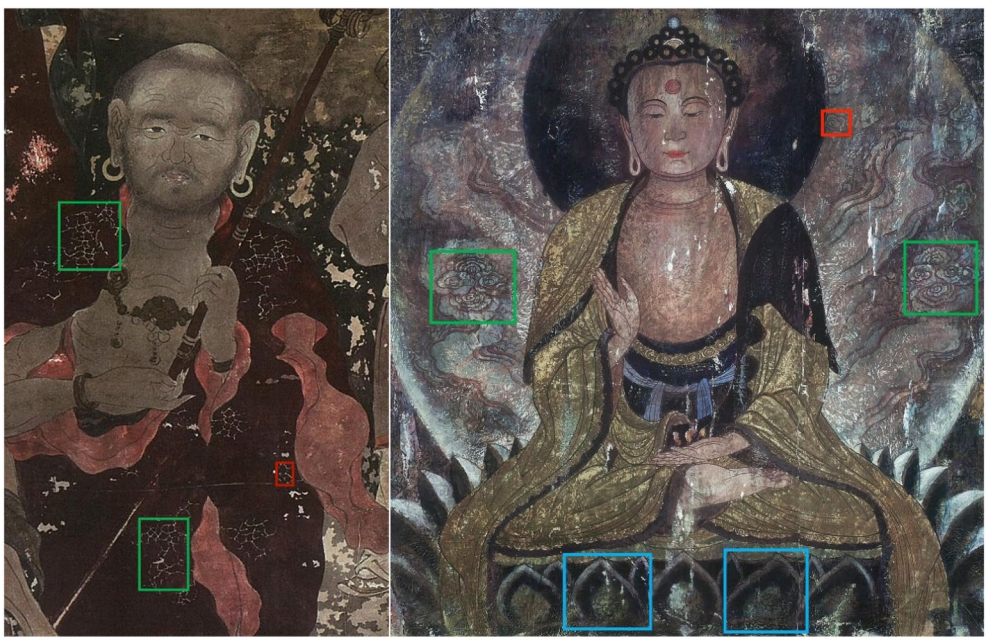


Fig. 11 Examples of murals with broken areas (**a**) Luohan Painting (**b**) Buddha Statue

Conclusion

This paper introduces a novel encoder-decoder network called DACMixer to enhance the accuracy in predicting the broken areas in murals. By extracting semantic information from different scales of broken areas, modeling long-range dependencies, and integrating low-level semantic details, DACMixer improves the accuracy in predicting cracking, peeling, and fading in murals with complex backgrounds. A comparison of the proposed model with other methods shows that our model achieves the highest values for IoU of the broken areas, MIoU, and Dice coefficient, which are 78.3%, 87.5%, and 85.7% respectively. These results are strong evidence that the extraction and fusion of multi-scale convolutional and attentional features in the network are key to accurately predicting the broken areas in murals. Additionally, to alleviate the problem of low availability of annotated murals for the training network, we present a diverse mural dataset with pixel-level annotations, consisting of 7282 mural images.

Abbreviations

DACMixer	The Mixer of Dual Attention and Convolution
MLP	Multilayer perceptron
MFF	MLP-based feature fusion
YMDA	Yunnan Murals Dataset Aug
IoU	Intersection over Union
MIoU	Mean Intersection over Union
Broken.IoU	The Intersection over Union of the broken category
PSPNet	Pyramid scene parsing network
MNF	Minimum Noise Fraction
BP	Back propagation
ResNet	Residual network
Res-UNet	Residual UNet
DANet	Dual attention network
GELU	Gaussian error linear unit
RefineNet	Refinement network
SGD	Stochastic Gradient Descent
C-Fusion	Convolution-fusion
CNN	Convolutional neural network
GSCNN	Gated Shape Convolutional Neural Network
HRNetv2	High-Resolution Network
OCRNet	Object-Contextual Representations Network
GCNet	Global Context Network
STDC	Short-Term Dense Concatenate Network
DPT	Dense Prediction Transformer
Vit	Vision transformer

Acknowledgements

Not applicable.

Author contributions

HNY designed and implemented the proposed model. WH and YGW guided the writing process and supervised the entire process. All authors read and approved the final manuscript.

Funding

This research was supported by the National Natural Science Foundation of China (Grant No. 62061049, Grant No. 12263008).

Availability of data and materials

The datasets generated and/or analysed during the current study are available in the Dataset repository, https://github.com/Thehunans/Yunnan.Murals.Dataset.Aug

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 10 May 2023 Accepted: 22 July 2023
Published online: 03 August 2023

References

1. Bi X, Liu H, Wang X, Wang W, Yang Y. The segmentation of Thangka damaged regions based on the local distinction. J Phys Conf Ser. 2017;787(1): 012010.
2. Jaidilert S, Farooque G. Crack detection and images inpainting method for Thai mural painting images 2018 IEEE 3rd international on image, vision and computing (ICIVC). IEEE. 2018;143–8.
3. Zhang Z, Shui W, Zhou M, Xu B, Zhou H. Research on disease extraction and inpainting algorithm of digital grotto murals. Appl Res Comput. 2021;38(8):2495–24982504 (**in Chinese**).
4. Deng X, Yu Y. Automatic calibration of crack and flaking diseases in ancient temple murals. Herit Sci. 2022;10:163. https://doi.org/10.1186/s40494-022-00799-y.
5. Cao J, Li Y, Cui H, Zhang Q. Improved region growing algorithm for the calibration of flaking deterioration in ancient temple murals. Herit Sci. 2018;6:67. https://doi.org/10.1186/s40494-018-0235-9.
6. Chen C, Papandreou G, Kokkinos I, Murphy K, Yuille L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans Pattern Anal Mach Intell. 2017;40(4):834–48.
7. Cao J, Tian X, Chen Z, Rajamanickam L, Jia Y. Ancient mural segmentation based on a deep separable convolution network. Herit Sci. 2022;10:11. https://doi.org/10.1186/s40494-022-00644-2.
8. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen C. Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018;4510–20.
9. Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017;2881–90.
10. Lin Y, Xu C, Lyu S. Disease Regions recognition on mural hyperspectral images combined by MNF and BP neural network. J Phys Conf Ser. 2019;1325(1): 012095.
11. Yuan Q, He X, Han X, Guo H. Automatic recognition of craquelure and paint loss on polychrome paintings of the Palace Museum using improved U-Net. Herit Sci. 2023;11:65. https://doi.org/10.1186/s40494-023-00895-7.
12. Wu M, Jia M, Wang J. TMCrack-Net: a U-shaped network with a feature pyramid and transformer for mural crack segmentation. Appl Sci. 2022;12(21):10940.
13. Yi Z, Tang Q, Azizi S, Jang D, Xu Z. Contextual residual aggregation for ultra high-resolution image inpainting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020;7508–17.
14. Liu G, Reda F A, Shih K J, Wang T C, Tao A, Catanzaro B. Image inpainting for irregular holes using partial convolutions. In: Proceedings of the European conference on computer vision (ECCV). 2018;85–100.
15. Zhou Y, Barnes C, Shechtman E, Amirghodsi S. Transfill: Reference-guided image inpainting by merging multiple color and spatial transformations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021;2266–76.
16. Qin X, Zhang Z, Huang C, Dehghan M, Zaiane OR, Jagersand M. U2-Net: going deeper with nested U-structure for salient object detection. Pattern Recognit. 2020;106: 107404.
17. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. Medical Image Computing and

Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18. 2015;234–41.
18. Lou A, Guan S, Loew M. DC-UNet: rethinking the U-Net architecture with dual channel efficient CNN for medical image segmentation. Medical Imaging 2021: Image Processing. 2021;11596:758–68.
19. Cao H, Wang Y, Chen J, Jiang D, Zhang X, Tian Q, Wang M. Swin-unet: Unet-like pure transformer for medical image segmentation. Computer Vision-ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III. 2023;205–218.
20. Xie E, Wang W, Yu Z, Anandkumar A, Alvarez JM, Luo P. SegFormer: simple and efficient design for semantic segmentation with transformers. Adv Neural Inf Process Syst. 2021;34:12077–90.
21. Fu J, Liu J, Tian H, Li Y, Bao Y, Fang Z, Lu H. Dual attention network for scene segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019;3146–54.
22. Zagoruyko S, Komodakis N. Wide residual networks. arXiv preprint. 2016. arXiv:1605.07146.
23. Hendrycks D, Gimpel K. Gaussian error linear units (gelus). arXiv preprint. 2016. arXiv:1606.08415.
24. Lin G, Milan A, Shen C, Reid I. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017;1925–34.
25. Everingham M, Eslami SMA, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes challenge: a retrospective. IJCV. 2015;111(1):98–136.
26. Takikawa T, Acuna D, Jampani V, Fidler S. Gated-scnn: Gated shape cnns for semantic segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV). 2019;5229–38.
27. Sun K, Zhao Y, Jiang B, Cheng T, Xiao B, Liu D, Mu Y, Wang X, Liu W, Wang J. High-resolution representations for labeling pixels and regions. arXiv preprint. 2019. arXiv:1904.04514.
28. Yuan Y, Chen X, Wang J. Object-contextual representations for semantic segmentation. Computer Vision-ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16. 2020;173–90
29. Cao Y, Xu J, Lin S, Wei F, Hu H. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In: Proceedings of the IEEE/CVF international conference on computer vision workshops. 2019;1–10.
30. Fan M, Lai S, Huang J, Wei X, Chai Z, Luo J, Wei X. Rethinking bisenet for real-time semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021;9716–25.
31. Ranftl R, Bochkovskiy A, Koltun V. Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021;12179–88.
32. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016;770–8.
33. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, et al. An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint. 2020. arXiv:2010.11929.
34. Liu Z, Mao H, Wu C Y, Feichtenhofer C, Darrell T, Xie S. A convnet for the 2020s. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022;11976–86. arXiv:2201.03545.
35. He K, Zhang X, Ren S, Sun J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. 2015;1026–34. arXiv:1502.01852.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com