

Article

Intelligent Recognition and Restoration of Mural Damage Based on DeepLabv3 and Stable Diffusion

Chong Rong ^{1,2,3,*}, Dashuai Yang ^{2,3}, Wenkai Tian ^{2,3}, Yi Tao ^{1,2}, Qiuwei Wang ^{1,2} and Peng Wang ^{1,2}

¹ State Key Laboratory of Green Building, Xi'an University of Architecture and Technology, Xi'an 710055, China

² School of Civil Engineering, Xi'an University of Architecture & Technology, Xi'an 710055, China

³ Key Lab of Structural Engineering and Earthquake Resistance, Ministry of Education (XAUAT), Xi'an 710055, China

* Correspondence: rongchong@xauat.edu.cn; Tel.: +86-17791318879

Abstract

Murals are not merely independent visual artworks. Rather, they are an integral part of architectural heritage, directly attached to buildings' structural elements, such as brick walls and vaults. However, murals are susceptible to various building-related types of damage, including structural cracks and moisture-induced peeling, due to long-term exposure to environmental factors and geological changes. As the progressive deterioration of these murals hastens the loss of mural value, professional assessment and restoration are urgently required. To tackle the issues of low efficiency in traditional structural damage detection and the absence of predictable repair plans, this paper presents a semi-automatic building-mural protection solution that integrates morphological assessment of mural deterioration with computer vision technology. This study establishes an image prediction system that integrates intelligent damage identification with virtual restoration. First, employing the PaddleSeg deep learning framework and the DeepLabv3 semantic segmentation model, this study used existing mural damage datasets to build a recognition model. The model allows for intelligent identification and labeling of multiple damage types. Subsequently, relying on the ComfyUI platform, Stable Diffusion was used to construct a virtual restoration model. LoRA (low-rank adaptation) technology was introduced to fine-tune the model specifically for the mural style, thus enhancing the directivity and accuracy of virtual restoration. Finally, by applying the results of the recognition model to the virtual restoration model, this study built an integrated system for mural damage diagnosis and virtual restoration. The results show that the damage recognition model achieved a mean intersection over union (mIoU) of 47.8% and a pixel accuracy of 77.97% on the test set, validating the feasibility of using semantic segmentation for mural damage detection. This study presents an integrated workflow framework integrating automatic damage identification and intelligent repair. As an expert-assisted tool, this framework shows application potential for preliminary exploration of mural disease diagnosis and virtual restoration plans, providing technical references for the digital protection of cultural heritage.



Academic Editor: Muhammad Junaid Munir

Received: 31 March 2026

Revised: 16 May 2026

Accepted: 17 May 2026

Published: 20 May 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: murals; damage detection; virtual restoration model; semantic segmentation; digital conservation

1. Introduction

As a crucial component of human cultural heritage, ancient murals not only bear abundant historical and artistic information [1] but also act as direct “indicators” of the

health status of the buildings to which they are attached (e.g., brick walls and arches). However, ancient tomb murals usually suffer from varying degrees of degradation [2]. Existing evidence indicates that this visual degradation is seldom merely superficial. Rather, it typically represents a direct manifestation of underlying building pathologies. For example, due to the differing thermal and moisture expansion coefficients of the mural base and the plaster layer, prolonged temperature and humidity cycles can cause stress accumulation between them, eventually resulting in cracking and delamination [3]. Furthermore, uneven settlement of the foundation and structural stress, such as the collapse of the tomb chamber or changes in the loads on surrounding buildings, can cause cracks to form and propagate [4]. Consequently, contemporary conservation efforts should bridge the gap between visual restoration and initial pathological assessment.

Restoring murals to their original appearance with high-fidelity preservation has become a prominent research topic in the fields of artifact conservation and digital humanities [5]. Traditional mural restoration primarily relies on expert experience and manual restoration work. Although this approach can achieve high-detail results, it is subjective, inefficient, and difficult to standardize [6].

Advances in artificial intelligence and computer vision have empowered researchers to apply digital technologies to aid in the restoration process. In engineering, machine learning shows great potential for structural damage prediction and health monitoring [7,8]. Consequently, specialized digital techniques capitalizing on these cross-disciplinary advancements are increasingly employed to aid mural restoration, thereby enhancing the effectiveness of conservation efforts. Wu et al. [9] introduced a lightweight detection network, Ghost-C3SE YOLOv5. This network enhanced feature extraction for minor damages, expedited training convergence, and thus improved real-time performance. However, their work mainly focused on object detection. This approach lacks sufficient precision for extremely minute or long cracks and only conducts detection without virtual restoration.

Chen et al. [10] developed an improved PLD-YOLO model based on YOLOv8s-seg. This model is tailored to detect and segment pigment loss in ancient murals. By combining object detection and instance segmentation, they achieved pixel-level segmentation of pigment loss areas. However, the PLDS-YOLO model primarily focuses on optimizing pigment loss detection and lacks universality for multiple damage types.

In summary, although extensive research has been conducted on mural damage detection and AI-assisted restoration, most current efforts focus on isolated stages. There is still a shortage of systematic research integrating intelligent identification of various mural damages, automatic mask generation, and AI-based virtual restoration of damaged areas.

This paper aims to transcend the simplistic scope of image restoration by proposing an intelligent framework for mural disease identification and virtual restoration based on computer vision. Furthermore, this framework serves as an auxiliary decision-support tool for the preventive protection of architectural heritage, structural health diagnosis, and the formulation of intervention strategies. This method uses DeepLabv3 semantic segmentation to accurately identify various types of damage. Subsequently, it combines with a Stable Diffusion model fine-tuned on the ComfyUI platform to achieve intelligent restoration in a specific style. This process constructs a human-in-the-loop digital system for intelligent identification and virtual restoration. In this system, the identification process is fully automated, and users refine the virtual restoration outcomes.

Despite the development of diffusion-model-based restoration techniques in recent studies [11,12], these methods typically rely on manually annotated masks, target only one type of damage, and lack automated multi-category semantic segmentation for complex mural scenes. This highlights the novelty of the proposed integrated framework in both theory and practice.

In this context, this paper first analyzes the various types of damage in murals and develops a quantitative extraction method. Next, it employs a semantic segmentation model to accurately identify and label mural damage. Subsequently, it uses a diffusion model to automatically perform virtual restoration of labeled damaged areas. Finally, by integrating damage detection and local virtual restoration, this study proposes a semi-automated digital restoration workflow framework. This framework, which employs a human–computer collaborative mode, serves as an exploratory tool to assist conservation experts in formulating virtual restoration schemes.

2. Mural Damage Analysis

2.1. Damage Types and Characteristics

This research focuses on the murals unearthed in an ancient tomb at the Affiliated Primary School of Xi'an Jiaotong University. The study examined four mural surfaces, as shown in Figure 1. The murals primarily exhibit three types of damage: cracking, brick-related damage, and peeling. These damage types will be analyzed in detail to better understand the condition of the murals and inform the subsequent restoration process.

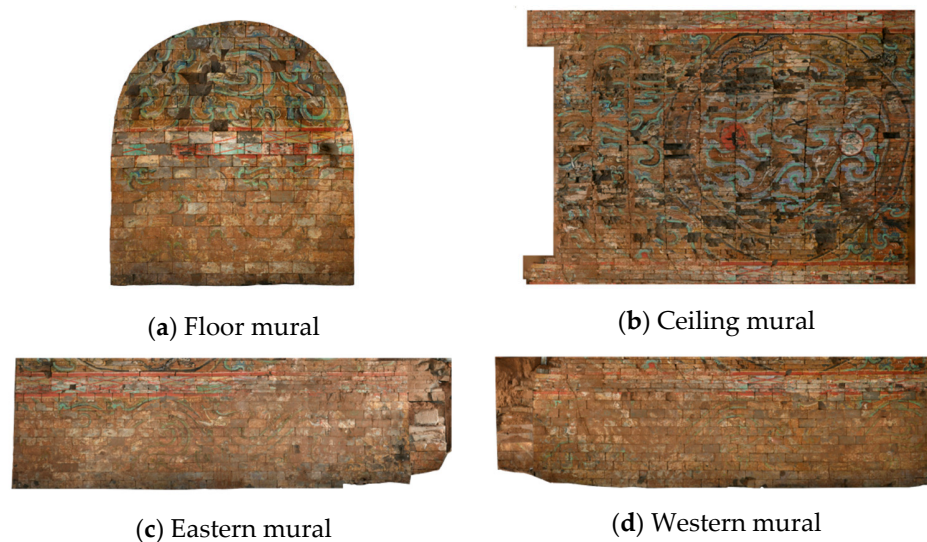


Figure 1. Original painting images.

(1) Cracking Damage

Cracking represents one of the most common forms of damage. In this study, the cracks (see Figure 2) can be classified as linear cracks and branching cracks. Linear cracks (see Figure 2a) are mainly caused by structural stress and appear as long, thin, straight (or nearly straight) lines. They usually have a uniform width and distinct direction. Branching cracks (see Figure 2b) exhibit a tree-branch-like pattern, with the primary crack extending into several secondary cracks.

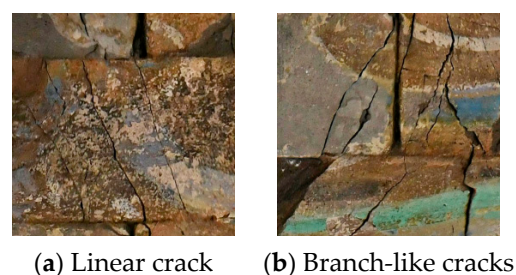


Figure 2. Cracks in murals.

The primary cause is the difference in thermal and moisture expansion coefficients between the base and the plaster layer. Over time, this difference leads to the accumulation of stress. Moreover, structural stress resulting from foundation settlement or external loads is also a significant contributing factor [13].

Both crack types show significant differences in color and texture from the intact mural surface. However, their shapes visually resemble brick joints (the mortar lines between bricks), posing a moderate challenge for automated identification. Structurally, cracks appear as thin interruptions in the image, but the content on either side usually remains continuous, indicating that the loss of visual information is relatively limited. Given these characteristics, this type of damage is generally suitable for digital restoration [14].

(2) Brick Damage

Brick-related damage primarily occurs in murals on brick substrates, as shown in Figure 3. It is characterized by broken corners, fragmentation, and missing sections. These damaged areas not only undermine the integrity of the mural but also can cause secondary detachment of the pigment layer.

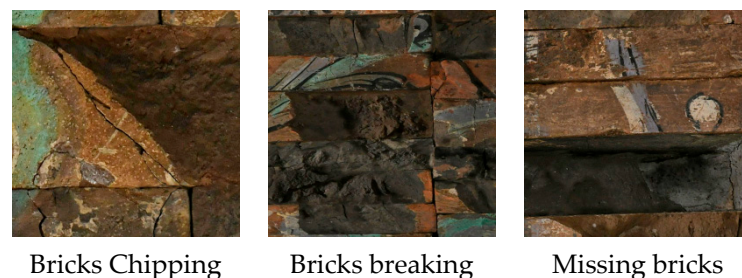


Figure 3. Brick damage.

Specifically, the main causes include long-term geological changes, degradation of base material properties leading to structural failure, foundation settlement, and direct impacts from external forces.

The common characteristic among the three types of brick-related damage is that the bricks experience varying degrees of destruction, with the mural content missing to different extents. These damaged areas are relatively large and are not interfered with by other damage types, making them more amenable to automatic identification. Structurally, brick-related damage appears as obvious concave areas in the image, making it visually distinctive. Therefore, this type of damage usually shows favorable restoration potential.

Damage to the brick structure directly signals the deterioration of the building's load-bearing base material and local structural failure. This type of damage usually results from the gradual weakening of material strength due to long-term geological rheology and groundwater erosion, or from direct impacts by external forces. In construction engineering, the high-precision segmentation of damaged areas in bricks using an artificial intelligence model is of great significance. By quantifying the area and distribution of these deep depressions, experts can objectively assess the loss rate of the effective load-bearing section of the masonry, thus determining the vulnerability of the structure in that area.

(3) Peeling

Peeling damage (see Figure 4) is another major cause of content loss in murals. Peeling presents as a local loss or degradation of the surface pigment layer. It either exposes the underlying brick or is accompanied by evident fading and color loss [15].



Figure 4. Peel off.

The main cause is the long-term exposure of murals to various environmental factors. This exposure triggers physical and chemical changes in the pigment layer, which weakens the bond strength and ultimately leads to detachment.

Structurally, peeling presents as irregular, block-shaped areas distributed unevenly. The visible loss of the pigment layer exposes the underlying brick. Owing to the large quantity and variable morphology of these areas, automatic identification is rather challenging. Consequently, this type of damage is generally more difficult to restore perfectly during the digital restoration process.

2.2. Feature and Importance Analysis

2.2.1. Damage Feature Analysis

Based on the above analysis, the key visual characteristics that form the basis for AI model learning are summarized. Cracks appear as linear or branching textures with distinct edge gradients but occupy a relatively low percentage of pixels. Peeling presents as irregular, block-shaped areas with color and texture loss, distributed sporadically. Brick-related damage shows significant volume loss and structural depressions.

In summary, different damage types show differences in morphology, texture, and frequency distribution. This allows for classification using a deep learning-based multi-class semantic segmentation model. To avoid model confusion during training, the “brick joints” category, which visually resembles cracks, was deliberately included as an additional class. The four final target categories are cracks, brick-related damage, peeling, and brick joints.

2.2.2. Importance Analysis

In the recognition task, restoration priorities are allocated to two different targets.

(1) Primary Targets (Brick-related Damage, Peeling)

In actual murals, these two types of damage account for the vast majority of the damaged area. They directly cause base material exposure and extensive pigment-layer loss, severely undermining integrity and visual aesthetics. Accurate identification of these two types of damage is crucial for subsequent virtual restoration work.

(2) Secondary Target (Cracks)

Although cracks occupy less area than peeling, they are widely distributed and have complex shapes. Cracks break up the details of the image. Identifying them is mainly aimed at fine-texture repair and “healing” to prevent the image from appearing fragmented. Despite their small area, cracks play a vital role in restoring the continuity of the lines.

3. Recognition and Labeling of Mural Damage

This study developed its algorithms in the Python 3.8.20 environment. It used PaddlePaddle-GPU 2.4.2 as the deep learning core framework and combined it with the

PaddleSeg 2.7.0 semantic segmentation development kit to complete model construction and training.

In data processing and experiments, the following key libraries were primarily used: OpenCV-python (4.12.0.88) and Pillow (10.4.0) for image reading, enhancement, and pre-processing; NumPy (1.24.4) for matrix operations, Pandas (2.0.3) for label and data management; VisualDL (2.5.3) to monitor real-time loss function (Loss) and index changes during training, Matplotlib (3.7.5) to draw result comparison charts; Scikit-learn (1.3.2) and SciPy (1.10.1) for statistical analysis of evaluation indicators, and tqdm (4.67.1) for visualizing training progress. These libraries, each with its specific functionality, were essential for the successful execution of data processing and experiments.

All experiments were conducted on a server equipped with an 11th Gen Intel(R) Core(TM) i7-11850H @ 2.50 GHz processor, 32 GB of memory, and an NVIDIA RTX A4000 Laptop GPU with 8 GB of memory. The high-performance hardware configuration, including the specified processor, memory, and GPU, ensured efficient execution of the computationally intensive tasks involved in the experiments. CUDA 11.2 was used for hardware acceleration. The complete list of the software environment and libraries used is presented in Figure 5.

```
(paddleseg2.7) E:\Program Files\PaddleSeg-2.7.0>conda list
# packages in environment at E:\Program Files\miniconda\envs\paddleseg2.7:
#
# Name                    Version            Build                Channel
astor                     0.8.1              pypi_0              pypi
babel                     2.17.0             pypi_0              pypi
bce-python-sdk           0.9.46             pypi_0              pypi
clicker                   1.8.2              pypi_0              pypi
ca-certificates          2025.9.9           haa95532_0          https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
certifi                   2025.8.3           pypi_0              pypi
charset-normalizer        3.4.3              pypi_0              pypi
click                     8.1.8              pypi_0              pypi
colorama                  0.4.6              pypi_0              pypi
contourpy                 1.1.1              pypi_0              pypi
cycler                    0.12.1             pypi_0              pypi
decorator                  5.2.1              pypi_0              pypi
filelock                  3.16.1             pypi_0              pypi
flask                      3.0.3              pypi_0              pypi
flask-babel               4.0.0              pypi_0              pypi
fonttools                 4.57.0             pypi_0              pypi
future                     1.0.0              pypi_0              pypi
idna                       3.10               pypi_0              pypi
importlib-metadata        8.5.0              pypi_0              pypi
importlib-resources       6.4.5              pypi_0              pypi
itsdangerous              2.2.0              pypi_0              pypi
jinja2                    3.1.6              pypi_0              pypi
joblib                    1.4.2              pypi_0              pypi
kiwisolver                 1.4.7              pypi_0              pypi
libffi                     3.4.4              hd77b12b_1          https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
markupsafe                 2.1.5              pypi_0              pypi
matplotlib                3.7.5              pypi_0              pypi
numpy                     1.24.4             pypi_0              pypi
opencv-python             4.12.0.88          pypi_0              pypi
openssl                    3.0.17             h3562f6_0           https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
opt-einsum                 3.3.0              pypi_0              pypi
packaging                  25.0               pypi_0              pypi
paddle-bfloat              0.1.7              pypi_0              pypi
paddlepaddle-gpu         2.4.2.post112     pypi_0              pypi
paddleseg                 2.7.0              pypi_0              pypi
pandas                     2.0.3              pypi_0              pypi
pillow                     10.4.0             pypi_0              pypi
pin                        24.2               py38haa95532_0     https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
prettytable               3.11.0             pypi_0              pypi
protobuf                  3.20.0             pypi_0              pypi
psutil                    7.1.0              pypi_0              pypi
pycryptodome              3.23.0             pypi_0              pypi
pyparsing                  3.1.4              pypi_0              pypi
python                     3.8.20             h8205438_0          https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
python-dateutil           2.9.0.post0       pypi_0              pypi
pytz                       2025.2             pypi_0              pypi
pyyaml                     6.0.3              pypi_0              pypi
rarfile                    4.2                pypi_0              pypi
requests                   2.32.4             pypi_0              pypi
scikit-learn              1.3.2              pypi_0              pypi
scipy                      1.10.1             pypi_0              pypi
setuptools                 75.1.0             py38haa95532_0     https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
six                         1.17.0             pypi_0              pypi
sklearn                   0.0                pypi_0              pypi
sqlite                     3.50.2             hda9a48d_1          https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
threadpoolctl             3.5.0              pypi_0              pypi
tqdm                      4.67.1             pypi_0              pypi
tzdata                     2025.2             pypi_0              pypi
ucrt                       10.0.22621.0      haa95532_0          https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
urllib3                    2.2.3              pypi_0              pypi
vc                          14.3               h2df5915_10         https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
vc14_runtime              14.44.35208       h4922774_10         https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
visualdl                   2.5.3              pypi_0              pypi
vs2015_runtime            14.44.35208       ha6b5a95_10         https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
wcwidth                   0.2.14             pypi_0              pypi
werkzeug                   3.0.6              pypi_0              pypi
wheel                      0.44.0             py38haa95532_0     https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
zipp                       3.20.2             pypi_0              pypi

(paddleseg2.7) E:\Program Files\PaddleSeg-2.7.0>
```

Figure 5. Integrating the required software infrastructure and library dependencies for recovery.

3.1. Model Establishment

(1) Model Selection

To select the most suitable semantic segmentation model for this study, this section introduces the classic U-Net, based on convolutional neural networks, and SegFormer, based on the Transformer architecture, as benchmark models for comparative experiments. To ensure the fairness and reliability of the results, all experiments were conducted under the same dataset, training strategy, and hardware environment. The quantitative evaluation results of each model are shown in Table 1.

Table 1. Comparison of Performance of Different Semantic Segmentation Models.

Model	mIoU	Acc	Kappa
U-Net	0.4281	0.7229	0.4395
SegFormer	0.4695	0.7539	0.4959
DeepLabv3	0.4780	0.7797	0.5181

The experimental results clearly show that DeepLabv3 achieved the best performance in all three key evaluation metrics. Specifically, compared to the classic U-Net model, the average intersection-over-union (mIoU) and pixel accuracy (Acc) of DeepLabv3 increased by 0.0499 and 0.0568, respectively. Moreover, the Kappa coefficient showed a relative improvement of approximately 17.88%. Furthermore, when compared with the SegFormer model, which has shown excellent performance in recent years, DeepLabv3 maintained its performance superiority. Specifically, its mIoU, Acc, and Kappa indicators were 0.0085, 0.0258, and 0.0222 higher, respectively.

The comparison results indicate that DeepLabv3 demonstrates superior performance in feature extraction and in capturing multi-scale contextual information. DeepLabv3 achieves higher segmentation accuracy and overall reliability compared to U-Net and SegFormer. Therefore, it was reasonable and beneficial to choose DeepLabv3 as the core model for this study.

Based on the DeepLabv3 model [16], this study develops a Convolutional Neural Network (CNN)-based semantic segmentation model [17]. Figure 6 shows the logic flow of this model. This model exhibits high-precision segmentation capability and can effectively identify damage ranging from large-scale peeling to fine cracks.

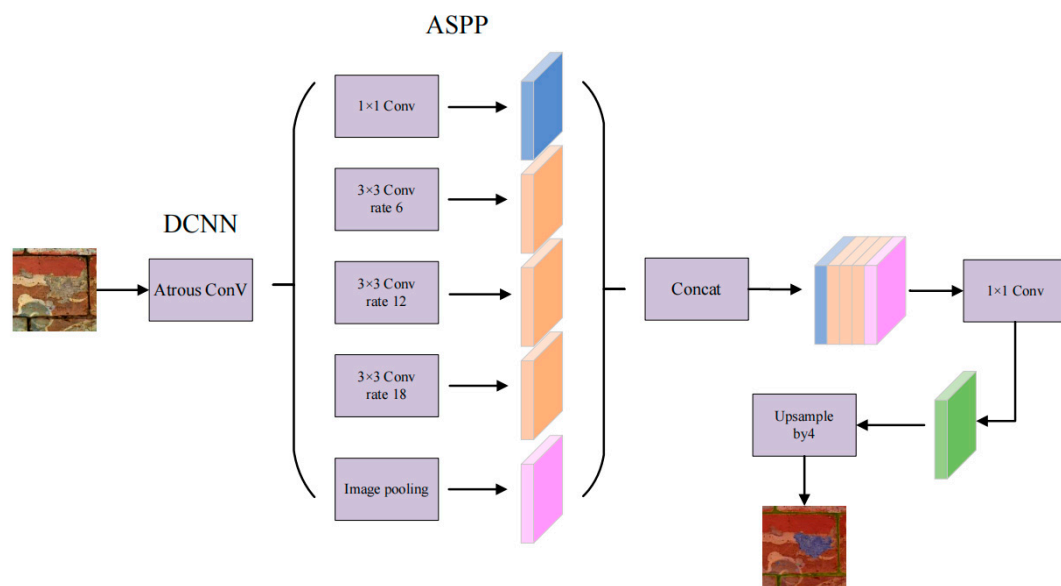


Figure 6. Semantic Segmentation Flowchart.

Note: Feature extraction layer (DCNN with Atrous Convolution): The input mural image first enters the deep convolutional neural network (DCNN). To expand the receptive field without losing spatial resolution, the model introduces atrous convolution (Atrous Convolution). This allows the model to obtain more dense feature responses, which is crucial for identifying subtle cracks or small areas of peeling in the mural.

The Atrous Spatial Pyramid Pooling (ASPP) module is crucial for recognizing multi-scale damage. This module contains five parallel branches: a 1×1 standard convolution layer; three 3×3 atrous convolution layers with sampling rates of $r = 6, 12,$ and 18 ; and an image-level pooling layer to obtain global statistical information to enhance the model's understanding of the complex mural background.

Feature fusion and dimensionality reduction (Concat and 1×1 Conv): The feature maps produced by each branch of the ASPP are concatenated along the channel dimension (Concat). Subsequently, a 1×1 convolution is used to fuse features and adjust the number of channels, reducing computational complexity and integrating multi-scale information.

Upsampling and output (Upsample by 4): The fused feature layer is upsampled by a factor of four using bilinear interpolation to restore the feature map to the original image size.

Damage prediction result: The final output layer assigns a category label to each pixel, generating a color damage identification mask consistent with the original image size.

(2) Dataset Creation

To maintain high resolution, enable effective feature learning under limited video memory constraints, and expand the sample size simultaneously to enhance training stability and accuracy, we used a custom script to segment the original images into 768×768 non-overlapping patches. After completing the division, a total of 1344 local images were obtained. Each local image is distinct and non-overlapping. The reasons for selecting the segmentation size are as follows: if the size is too small (e.g., 512×512), the contextual semantic information would be insufficient, hindering the model's understanding of the macroscopic damage distribution; if the size is too large (e.g., 1024×1024), it would increase the memory load and decrease the training efficiency.

Following general experience [18], which indicates that over 70% of the data should be dedicated to training, the dataset was split at an 8:1:1 ratio. This splitting yielded a training set of 1075 images, a validation set of 134 images, and a test set of 135 images. The training, validation, and test sets were derived from different original images.

(3) Data Labeling

Precise labeling is essential to ensure that the model learns visual features accurately. The damage boundaries were manually annotated point by point using the "Labelme" and "Polygon" tools. Four categories were labeled: crack, brick_damage, peel_off, and the interfering category "brick_joint", as shown in Figure 7.

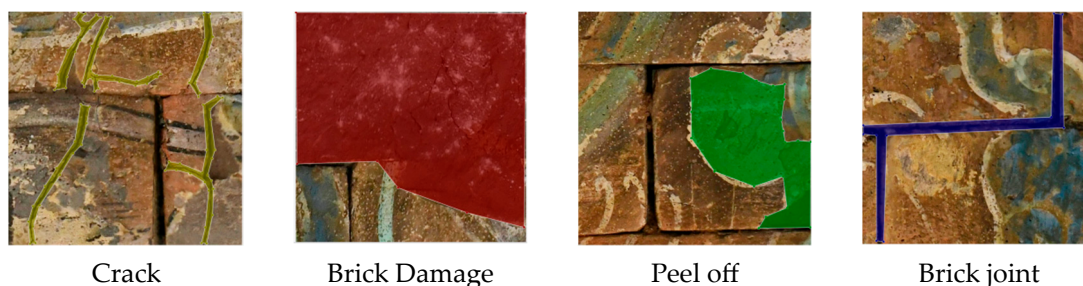


Figure 7. Annotation status by category. Note: For the sake of ensuring the clarity of the illustrations, the annotations in the figures have been simplified and enhanced solely for illustrative purposes.

To ensure the reproducibility of scientific experiments and dataset standardization, this study formulated strict annotation protocols.

First, all annotation work was carried out directly on the 768×768 -pixel local images after segmentation, rather than on the ultra-high-resolution original images. This ensured that the annotation coordinates were strictly aligned with the resolution of the model training input, thereby avoiding boundary deformation caused by scaling.

Second, the annotation operation principles for various categories are elaborated upon in Section 2.1 on injury types and characteristics.

Third, when categories overlapped, the principle of prioritizing injury over structure was applied. If cracks or peeling extended into the brick-joint area, the overlapping part should be uniformly annotated as the corresponding injury category (crack or peel_off), excluding the brick_joint label. In transitional regions with extremely blurred boundaries, annotators should delineate only the core areas where damage has been visually verified, thereby avoiding subjective over-extension. Notably, in the collected dataset, the vast majority of mural damages displayed distinct, sharp boundaries, making them highly amenable to precise pixel-level semantic segmentation. Although a small number of cases with blurred boundaries exist, the proportion of such cases is negligible and has minimal impact on the overall training and evaluation of the segmentation model.

Fourth, segmenting the images into local parts significantly clarifies the range of fine cracks, facilitating more accurate annotation.

Finally, the annotation process was completed by one person and then cross-checked by another. The cross-check focused on detecting missed annotations of fine cracks and label-conflict areas. For disputed boundaries, the final mask outline was determined through joint discussions.

To ensure the objectivity and reproducibility of the dataset, a quantitative assessment of the annotation consistency was carried out. Specifically, 5% of the dataset samples were randomly selected, and another researcher was asked to re-label them independently in accordance with the same annotation criteria. Subsequently, the Dice Similarity Coefficient (DSC) was employed to measure the consistency between the two sets of annotations. The analysis showed an average DSC of 0.8039, which suggested a high level of annotation consistency. This finding further verified that the proposed annotation guidelines could successfully mitigate subjective bias in complex situations.

After labeling, the JSON files generated by Labelme were converted into semantic segmentation mask images using a conversion script, thereby creating a dataset ready for training.

(4) Model Settings and Training

The model was configured with five classes (four target categories plus background). In this study, the background category is defined as the undamaged and intact areas of the mural. The category "back_ground" is assigned the value 0, "crack" is assigned the value 2, "brick_joint" is assigned the value 3, "peel_off" is assigned the value 4, and "brick_damage" is assigned the value 5. Then, to determine the contribution of the "brick-joint" interference category to the overall performance improvement of the model, this study conducted ablation experiments for verification. The performance indicators before and after the experiments are shown in Table 2.

The experimental results show that removing the "brick-joint" category significantly decreased the model's overall performance. Therefore, the experiment demonstrates the necessity of including the "brick-joint" category. Retaining this category helps the model comprehend the overall image structure and maintain favorable overall indicators.

Table 2. Quantitative impact of isolating the ‘Brick Joint’ category on model performance.

Specific Indicators	Has “Brick_Joint” Category	No “Brick_Joint” Category
mIoU	0.4780	0.4447
Acc	0.7797	0.7451
Kappa	0.5181	0.4559
Dice	0.6319	0.6003

This network is based on the DeepLabV3 architecture, using ResNet50_vd as the backbone network. Subsequently, ReLU was employed as the standard activation function throughout the network. Spatial features were primarily extracted using 3×3 convolution kernels (including dilated convolutions), supplemented by 1×1 convolutions for channel projection. To enhance the detection of fine damages (e.g., linear cracks), the output stride was strictly set to 8, thereby maintaining a higher spatial resolution of the feature maps.

During training, empirical hyperparameters were adopted instead of automatic optimization. The training parameters included a batch size of 2, a total of 54,000 iterations, and a learning rate of 0.01. Optimization was carried out using Stochastic Gradient Descent (SGD) with a momentum of 0.9. CrossEntropyLoss was selected as the loss function to quantify the difference between the predicted probability distribution and the true labels. For each pixel in the image, the loss-function formula is as follows:

$$L = - \sum_{c=1}^M y_c \log(p_c) \quad (1)$$

Among them, M represents the total number of damage categories, y_c is a binary indicator variable (1 if the pixel belongs to the category, 0 otherwise), and p_c is the probability that the model predicts the pixel to belong to category c . This loss function encourages the model to assign a higher probability to the correct damage type.

Furthermore, data augmentation techniques, such as random scaling, cropping, horizontal flipping, and adjustments to brightness, contrast, and saturation, were applied. After completing the above configuration, the model training was initiated.

3.2. Model Prediction Results

To objectively evaluate the DeepLabv3 model’s recognition accuracy for mural damage, this study introduced two key semantic segmentation evaluation indicators: pixel accuracy (Acc) and mean intersection over union (mIoU).

Acc represents the ratio of correctly classified pixels to the total number of pixels. This metric reflects the intuitive accuracy of the model’s overall image-classification performance. The formula is as follows:

$$Acc = \frac{\sum_{i=1}^k n_{ii}}{\sum_{i=1}^k t_i} \quad (2)$$

The mIoU is the most commonly used metric for evaluating semantic segmentation performance. Specifically, it calculates the average intersection-over-union ratio across all classes. Compared with Acc, mIoU can better measure the model’s ability to capture the boundaries of damage, particularly in murals where the damaged areas account for a small proportion. The formula is as follows:

$$mIoU = \frac{1}{k} \sum_{i=1}^k \frac{n_{ii}}{t_i + \sum_{j=1}^k n_{ji} - n_{ii}} \quad (3)$$

Here, k represents the number of categories; n_{ji} denotes the number of pixels that are correctly classified; and t_i represents the total number of pixels in this category.

After training, the model attained a mIoU of 47.8% and an Acc of 77.97% on the test set. The prediction results are presented in Figure 8.

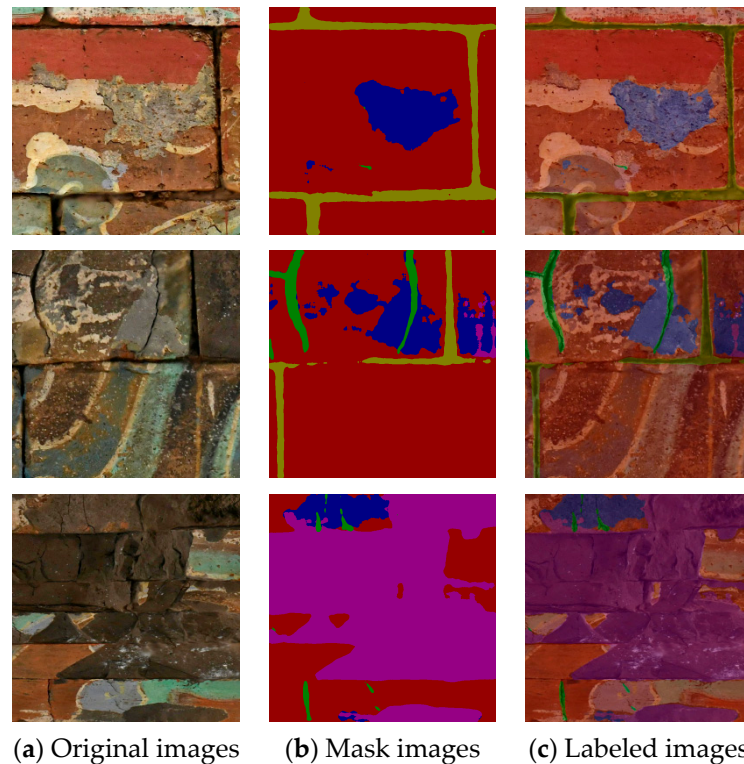


Figure 8. The predicted mask results of the deep learning model DeepLabv3. (red represents the background, green represents the cracks, yellow represents the brick joints, blue represents the peel off, and purple represents the brick damage).

The results indicate that the model exhibits effective detection capabilities for all three types of damage. It successfully generates segmentation masks that match the specific morphology of each damage type. In classification, the model accurately captures crack paths of different scales and effectively distinguishes them from brick joints, which look visually similar. Regarding peeling damage, although identifying areas with blurred edges is challenging, the overall localization of these areas is accurate.

Despite the relatively low mIoU value, constrained by the extremely small proportion of crack-occupied pixels, the visualization in Figure 8 shows that the generated masks fully cover the damaged areas and exhibit good edge alignment. Therefore, the model meets the basic requirements for localizing mural damage. Consequently, the resulting segmentation maps effectively serve as suitable masks for the subsequent virtual restoration module, validating the efficacy of the DeepLabv3-based semantic segmentation approach in mural damage detection.

Figure 9 shows the mIoU curve, Acc curve, loss curve, and learning rate (lr) graph obtained from the training. Initially, the model's mIoU increased rapidly during the early training phase, accompanied by a significant decrease in loss. After approximately 100 epochs, the mIoU of the validation set stabilized, and the losses of both the validation and training sets converged synchronously. No divergence was observed, where the training loss continued to decline while the validation set loss rebounded. Moreover, no significant overfitting was detected. Table 3 shows the IoUs for various damage types in the training and validation sets.

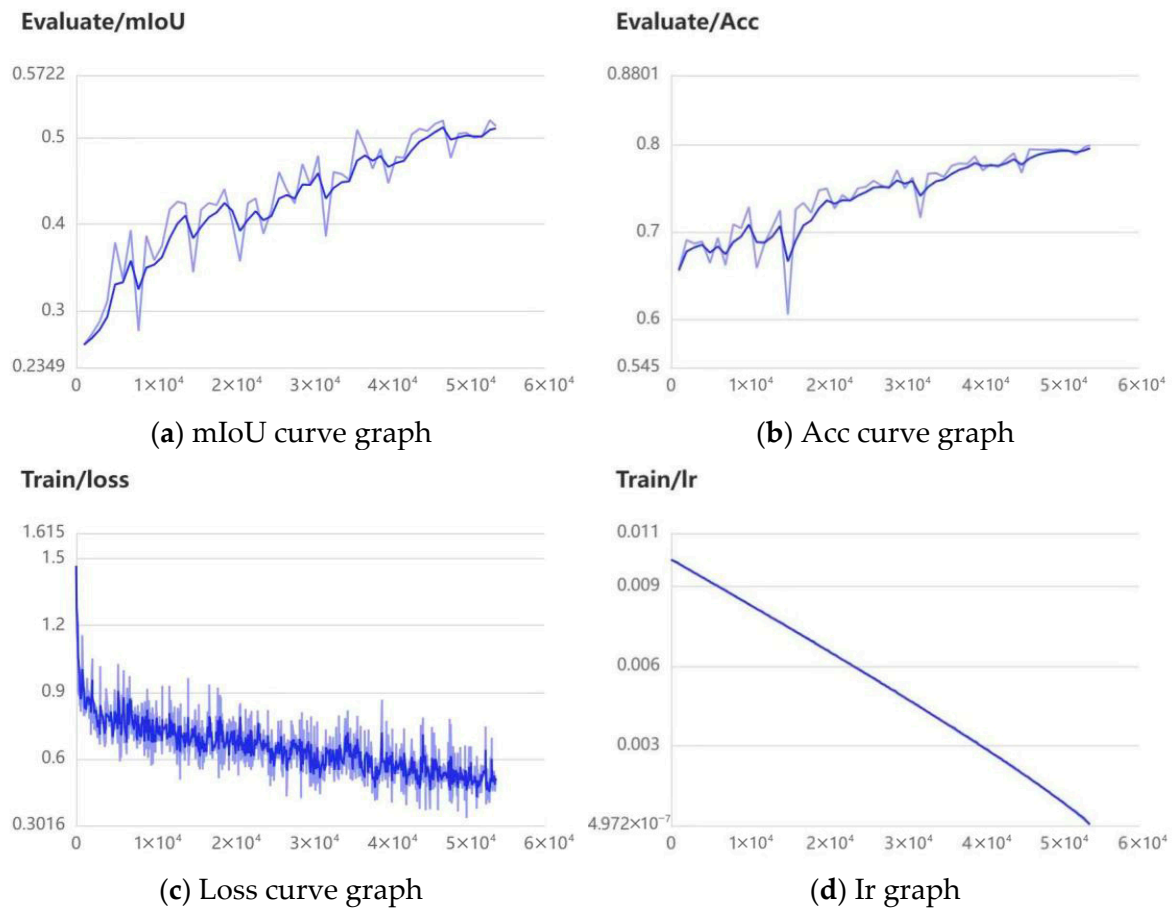


Figure 9. Evaluation curves of the model.

Table 3. Per-class Intersection over Union (IoU) evaluation across training and validation sets.

Type of Injury	Training Set IoU	Validation Set IoU
back_ground	0.7554	0.7432
crack	0.3656	0.3351
brick_joint	0.3733	0.3369
peel_off	0.4665	0.3963
brick_damage	0.6403	0.5787
mIoU	0.5202	0.4780

Although the value of 0.5202 may seem low for standard semantic segmentation, this is mainly due to the severe imbalance of classes in the damaged mural dataset. Specifically, cracks are represented as extremely fine linear structures, typically occupying less than 1% of the image's total pixels. This phenomenon significantly affects the calculation of the intersection-over-union (IoU), leading to a relatively low overall mIoU score. To address the model's poor performance on imbalanced data, a combined loss function integrating dice loss and cross-entropy loss was adopted instead of the standard cross-entropy loss.

The results are shown in Table 4. The results indicate that after incorporating dice loss, the model's IoU for the crack category increased by 2.87%. Moreover, it effectively addresses the issue of imbalanced crack damage categories.

The steady convergence of the loss curve indicates that this model has achieved stable feature-extraction capabilities. The research results show that the model can separate the damage features from the complex background. This satisfies the accuracy requirement for using the detected damage areas as a mask for the repair module.

Table 4. Comparative evaluation of per-class IoU before and after integrating dice loss.

Classes	Use Only CE Loss	Combining Dice + CE
back_ground	0.7432	0.7366
crack	0.3351	0.3638
brick_joint	0.3369	0.4325
peel_off	0.3963	0.3965
brick_damage	0.5787	0.5423

However, the accuracy of the model remains subject to certain limitations. The primary factors influencing the accuracy are as follows.

- (1) The diversity of the dataset is severely limited because it is derived from only four original mural images. Although the block-based segmentation method avoids direct data leakage and increases the sample size to 1344 local images, the dataset is still highly specific to a single case study. Limited exposure to different damage conditions and an unbalanced sample distribution limit the model's robustness.
- (2) Substantial visual differences exist between damage types. For example, cracks and peeling exhibit significant variations in morphology, texture, and color. These differences pose challenges to the model's ability to generalize its recognition capabilities.
- (3) Constrained by the 8 GB video-memory capacity of the RTX A4000, the experiment cannot directly process high-resolution original images during recognition model training. To sustain model training, the input images were divided into 768×768 -pixel blocks. Quantitative analysis indicated that the receptive field of a single image block accounted for merely about 0.3% of the total area of the entire mural ($13,012 \times 13,822$). This high cropping ratio prevents the model from obtaining macroscopic structural information across regions. This represents the primary hardware constraint that impedes the further improvement of recognition accuracy.

To rigorously evaluate the model's robustness, a four-fold cross-validation strategy with one area left out was adopted. Specifically, the 1344 slices were derived from four distinct spatial regions of the mural: the bottom, top, east, and west. In each fold, slices from three regions were combined to form the training set, while slices from the remaining, entirely unseen area were carefully separated as the validation set.

The cross-validation results are presented in Table 5. The model achieved an mIoU of 42.82% across four distinct mural areas, and its Acc reached 73.69%. Although the model's performance may show minor fluctuations depending on the specific test area, these fluctuations reflect the inherent differences in the damage characteristics of distinct mural walls. However, the low standard deviation attests to the stability of the DeepLabv3-based architecture, suggesting that the model has successfully learned the general damage features instead of overfitting to specific structural areas.

Table 5. Quantitative results of the spatial four-fold cross-validation.

Fold	mIoU (%)	Acc (%)
1	43.65	75.15
2	39.90	73.25
3	43.61	75.06
4	44.13	71.30
Mean \pm Std	42.82 \pm 1.97	73.69 \pm 1.81

3.3. The Specific Details of the Model in Terms of Mathematical and Engineering Implementation

- (1) Target variable encoding: In our dataset, the annotated damage masks are stored as single-channel tensors of shape $H \times W$ by integer label encoding. For example, 0 represents the background, and 1 represents a crack. Subsequently, to evaluate the objective function, these integer labels are internally transformed into one-hot-encoded vectors of shape $H \times W \times C$, where C represents the total number of categories. This vectorized representation allows the model to compute the standard cross-entropy classification loss by directly comparing the true values with the probability distribution of the network output.
- (2) For performance evaluation across multiple pixels, the metrics are globally aggregated over the entire dataset. Rather than calculating metrics for each image and then averaging them, our evaluation framework maintains a global confusion matrix. Specifically, it accumulates the true positives (TP), false positives (FP), and false negatives (FN) for each pixel of every image in the test set. Only after processing the entire dataset are these global totals used to calculate the final metrics, such as precision, recall, and mIoU. Therefore, this global micro-average method ensures the robustness of the overall metrics and avoids disproportionate biases resulting from including images with very small damaged areas.
- (3) Handling of multiple damages: When two or more types of damage co-exist in the same image, the DeepLabV3 model inherently addresses this issue at the pixel level. Our approach treats damage classification as an exclusive problem for each pixel. Consequently, despite the possible presence of multiple damage categories within one image, any single pixel is strictly allocated to one category. In the final layer of our network, the Softmax activation function is applied, followed by the argmax operation. This setup allows the model to predict the most prominent or most probable single-damage type for any given pixel, even in overlapping regions.

3.4. Analysis of Segmentation Errors Cases

To evaluate the model's applicability, Figure 10 shows four typical cases of prediction failures. The analysis result reveals that the model predominantly encounters issues, such as missed detection of crack targets and misdetection of non-crack textures in complex backgrounds. The reasons for these issues primarily encompass the following three aspects:

- (1) Feature sampling sparsity: Although DeepLabv3 introduces the ASPP module to obtain multi-scale context information, when dealing with slender linear targets such as cracks, a high hole rate leads to overly sparse convolution sampling points, giving rise to the "grid effect". This leads to the loss of sub-pixel-level geometric features during transmission. As a result, the predictions are distributed in a discrete, point-like pattern, making it difficult to maintain topological continuity.
- (2) Spatial resolution loss: Due to the multiple strides in convolutional and downsampling operations within the encoder, the spatial resolution of high-level feature maps significantly decreases. Specifically, for cracks that are only a few pixels wide, their feature representations are severely weakened in the deep network.
- (3) Imperfection in the decoding mechanism: Due to the absence of a precise high- and low-level feature fusion mechanism (e.g., the symmetrical decoder) within DeepLabv3, only simple bilinear interpolation is employed for upsampling. Consequently, the model has difficulty accurately reconstructing edge details when faced with interference from complex background textures (see Figure 10). This imbalance between spatial positioning accuracy and semantic discrimination ultimately leads to the fragmentation of the topological structure.

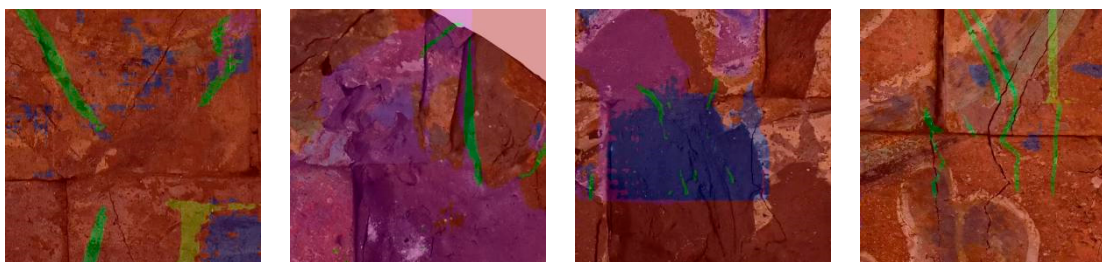


Figure 10. Failure cases involving incorrect detection and missed detection of crack types. (red represents the background, green represents the cracks, yellow represents the brick joints, blue represents the peel off, and purple represents the brick damage).

This imbalance between spatial positioning accuracy and semantic discrimination ultimately disrupts the topological structure of the damaged patterns. To address the existing limitations, the research team will perform more uncertainty analyses and conduct larger-scale validations in future studies. Owing to the limitations of experimental conditions and time, these details are not presented in this article.

4. Local Mural Damage Restoration Model

4.1. Model Selection

To choose the appropriate virtual restoration model, this study evaluated the Stable Diffusion and Lama models. In the practical application of ancient mural restoration, obtaining high-quality, complete original images as references (Ground Truth) is essentially impossible. Thus, traditional reference-based metrics (such as PSNR, SSIM, or LPIPS) are unsuitable for objectively assessing the restoration quality. To tackle this challenge, we developed a comprehensive evaluation framework based on no-reference image quality assessment (NR-IQA), that uses two complementary metrics: natural image quality evaluation (NIQE) and multi-scale deep learning quality (MUSIQ).

NIQE is used to measure the degree to which the result of virtual restoration diverges from the statistical characteristics of natural images. A lower score indicates a more natural-looking generated image with no noise or structural artifacts introduced.

MUSIQ is an assessment metric used to measure the overall quality in line with human subjective visual preferences. A higher score indicates a better-perceived quality of the image.

By integrating these two indicators, we provide a statistically rigorous assessment of the restoration results from both the “naturalness” and “visual quality” perspectives, eliminating the need for a reference image, as summarized in Table 6.

Table 6. Quantitative Evaluation of Mural Virtual Restoration Effects of Stable Diffusion and LaMa Models Based on NIQE and MUSIQ.

Image	Stable Diffusion		Lama	
	NIQE	MUSIQ	NIQE	MUSIQ
bottom	3.9748	56.5122	9.9731	46.8821
top	8.1514	57.6384	15.5330	48.1845
east	16.8712	47.7858	26.9618	46.2033
west	15.7483	47.3499	26.1555	45.6765

The quantitative evaluation results shown in Table 6 indicate that Stable Diffusion (SD) exhibits a significant and comprehensive superiority over LaMa in the virtual restoration of all examined mural regions.

Specifically, SD brings about a significant reduction in NIQE scores across all test areas compared to LaMa, suggesting that the textures and structural details generated by SD more closely resemble the statistical characteristics of genuine images. Conversely, LaMa often produces blurred or structurally inconsistent outputs, particularly in regions with complex damage.

Moreover, SD consistently obtains higher MUSIQ scores, indicating improved technical quality and aesthetic features, such as sharpness and structural coherence. Overall, these findings imply that SD's superior performance is not due to unrealistic artifact generation or excessive synthesis. Instead, it effectively provides high-quality semantic and textural reconstructions that conform to both statistical naturalness and human visual preferences. Therefore, compared to the more conservative LaMa approach, SD is better adapted to blind restoration tasks for severely damaged murals.

This study adopts ComfyUI as the platform for digital mural restoration, with SD 1.5 serving as the foundational model. Essentially, ComfyUI serves as a visual workflow interface for SD. Rather than generating images directly, it uses a node-based system to connect and activate components, such as the Stable Diffusion model, LoRA, and VAE for image generation and restoration. Therefore, ComfyUI serves as the workflow controller, while the actual virtual restoration capability stems from the underlying diffusion model.

SD represents a generative deep learning model based on diffusion model theory [19]. Its core concept comprises two processes. (1) Forward diffusion, where noise is gradually added to an original image until it turns into pure noise. (2) Reverse diffusion, during which the model learns to progressively “denoise” the data to restore a clear image. Through extensive training, the model learns to reconstruct precise image details from random noise, thereby forming the foundation of its restoration capabilities.

When applying this principle to the virtual restoration of murals, the first step is to mask the damaged areas. The model retains the unmasked regions unchanged and applies a precisely calculated amount of noise solely to the masked areas. During the denoising process, the U-Net utilizes the features of the surrounding undamaged pixels, along with text prompts, to ensure that the filled content matches the original image in terms of structure and lighting. The core architecture and workflow are shown in Figure 11 [20].

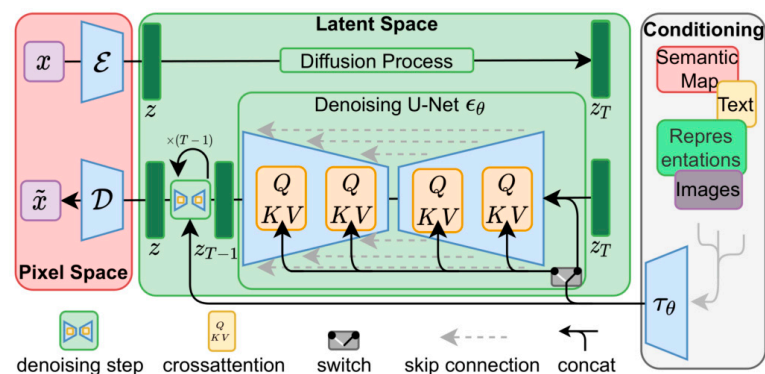


Figure 11. Stable Diffusion core architecture and workflow.

4.2. Model Establishment

4.2.1. LoRA Fine-Tuning Model Training

In traditional full-parameter fine-tuning, the process of updating model weights is expressed as:

$$W_{new} = W_0 + \Delta_w \quad (4)$$

where, W_0 represents the pre-trained weights; Δ_w represents the gradient update.

The basic principle of the LoRA method is to restrict the high-dimensional update matrix to a low-rank space. Specifically, the high-dimensional update matrix (Δ_W) can be decomposed into the product of two low-rank matrices:

$$\Delta_w = B \cdot A \quad (5)$$

where, $A \in \mathbb{R}^{r \times k}$; $B \in \mathbb{R}^{d \times r}$; $r \ll \min(k, d)$.

LoRA presents distinct advantages in both the model training and deployment phases. During the training phase, the original parameters of the pre-trained model remain frozen, and backpropagation updates only the parameters of the newly introduced low-rank matrices. This strategy substantially reduces computational costs and avoids interference with the model's pre-existing capabilities. During the inference phase, these low-rank matrices are directly added to the original parameters. This approach ensures that there is no additional computational latency. It sustains model performance while significantly reducing the number of trainable parameters.

To verify the contribution of the trained LoRA model to the experimental results, this study made a simple qualitative assessment by comparing the experiments before and after adding the LoRA model to the base images. Figure 12 shows the comparison chart of the experimental results.



(a) Experimental results before adding LoRA (b) Experimental results after adding LoRA

Figure 12. Comparison of generation effects before and after the introduction of the LoRA model.

The qualitative experimental comparison results show that the introduction of the LoRA model significantly enhances the overall quality and detail fidelity of the generated images. Without LoRA, issues include pattern breaks, rough smearing, and abrupt color changes. Conversely, the LoRA model effectively enables the network to learn specific fine-tuning features, making the line direction and semantic structure of the picture more coherent and smoother.

Furthermore, this model accurately reproduces the fine texture and physical gaps of the paint naturally adhering to rough bricks and their weathered surfaces. It not only eliminates local artifacts but also makes the overall color transition more rustic and harmonious. Consequently, it replicates the historical weight and artistic style of ancient painted murals, verifying LoRA's core contribution to enhancing the expressiveness of texture details and style consistency.

This study constructed a dataset of 30 high-fidelity, stylistically consistent mural images. All samples were uniformly standardized to a 768×768 -pixel resolution to ensure consistent tensor dimensions and enhance batch-computing efficiency. Semantic annotations were automatically extracted using the WD1.4 model, supplemented by manual cleaning to eliminate redundant labels, ensuring high alignment between the text prompt space and the image feature space.

The model architecture adopted SD1.5, and the LoRA technology was introduced for low-rank adaptive fine-tuning. The training hyperparameters were set as follows: the input resolution was 768×768 , the batch size was 2, and the number of training epochs was set to 10–20. This configuration aims to balance the model's ability to fit the texture features of the murals and its generalization performance, achieving the “restoring as it was” effect while effectively preventing overfitting.

4.2.2. Workflow Establishment

Figure 13 shows the workflow framework. Due to the high sensitivity of the mural virtual restoration process to generated parameters, this study systematically analyzed and optimized the selection criteria of key parameters. This was carried out to ensure the reliability of the virtual restoration results and the reproducibility of the experiments.

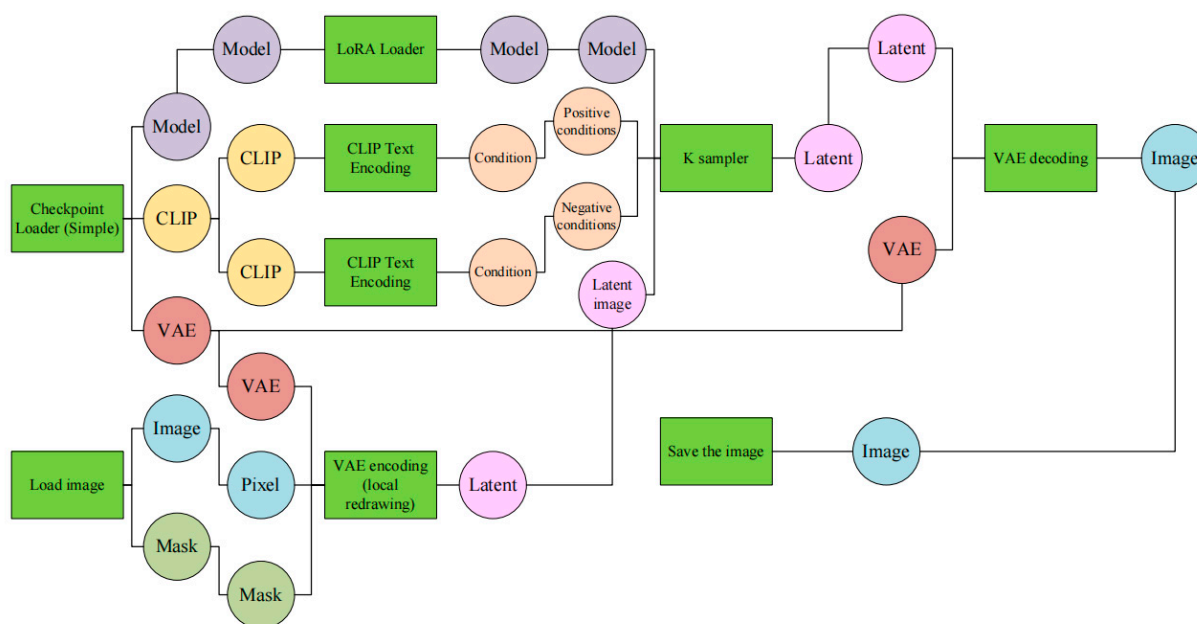


Figure 13. Inpainting workflow.

In the Checkpoint loader node, this study chose the SD1.5 model. Numerous experiments have shown that this model exhibits greater stability and controllability in the local inpainting task. Specifically, it can accurately lock the pixels in the non-masked areas, thus achieving a natural restoration of the damaged details while preserving the original structure and style of the mural.

To control the quality and degree of integration of the restoration process, this workflow conducts a series of sensitivity analyses on the parameters of core modules, such as the K-sampler:

- (1) Noise Reduction Intensity (Optimization Range: 0.4–0.7)

This parameter is crucial in determining restoration boundaries and content generation. Figure 14 shows the results of the step-by-step comparative experiment. Step-by-step comparative experiments indicated that when the noise-reduction intensity is below 0.4, insufficient noise is introduced into the latent space.

Conversely, when the noise-reduction intensity exceeds 0.7, the model-generated weights become large, easily disrupting the mural's original topological structure and generating abrupt content that is inconsistent with the surrounding textures. Therefore, setting this parameter within the range of 0.4–0.7 strikes the optimal balance between “preserving the image's original color” and “reconstructing damaged semantics”.

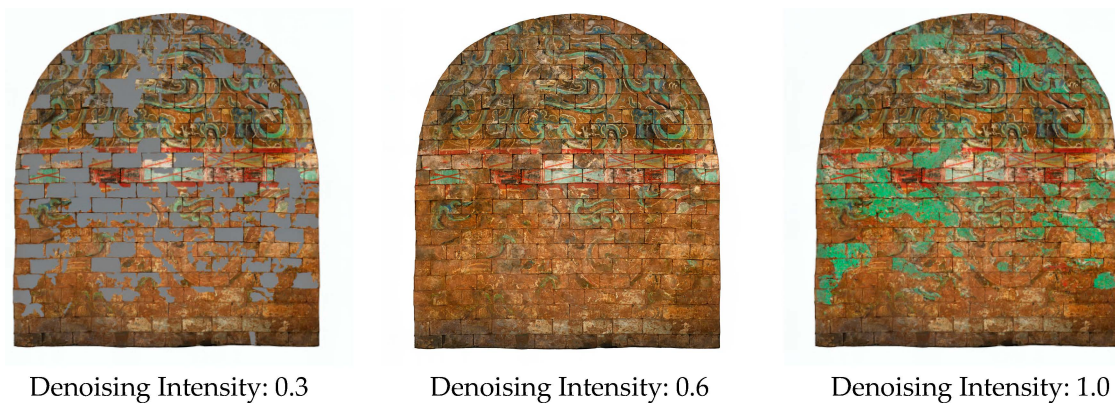


Figure 14. Sensitivity Analysis of Denoising Intensity on Virtual Restoration of Murals.

(2) Prompt Coefficient (CFG scale, Set Value: 7–8)

The CFG scale determines the alignment between the generated content and the CLIP text encoder (prompt), as shown in Figure 15. When the CFG scale is set in the range of 1–5, the AI has a high degree of operational freedom and might ignore specific prompts. When the CFG scale is set from 7 to 9, it ensures that the AI accurately identifies the specific objects and image guidance for local virtual restoration while preserving the picture's color-structure harmony. When the CFG scale is set from 10 to 15, the AI strictly adheres to the prompts, which is highly likely to cause a fragmented picture.



Figure 15. Sensitivity Analysis of the CFG scale for Virtual Restoration of Murals.

(3) Sampler and Scheduler Combination (dpmpp_2m + Karras)

Virtual mural restoration requires a high level of texture continuity. As a deterministic sampler, dpmpp_2m provides excellent convergence stability and consistency. When combined with the step-size distribution optimization of the Karras scheduler, the model can quickly achieve high-quality detail convergence within 20–30 iterations. This not only enhances computational efficiency but also avoids the ghosting phenomenon resulting from excessive sampling.

To explore the influence of randomness in the generation process, we selected a typical sample (the top mural) with diverse damage features as the benchmark. While maintaining core parameters, such as noise-reduction intensity and prompt-guidance coefficient at constant values, six repeated experiments were carried out. Quantitative evaluations were performed, as summarized in Table 7.

Table 7. Statistical analysis of restoration stability across different random seeds.

Metric	Mean	Standard Deviation (SD)	Coefficient of Variation (CV)
NIQE	5.737	1.871	32.6%
MUSIQ	53.80	3.88	7.2%

The quantitative results show that the coefficient of variation (CV) of the MUSIQ metric is merely 7.2%, indicating that the virtual restoration results maintain a high degree of consistency in visual aesthetics and overall quality across different instances. This remarkably low CV suggests that the model consistently converges to similar outputs, ensuring structural stability and aesthetic coherence.

These findings illustrate that our research has effectively regulated the generative behavior of Stable Diffusion via LoRA fine-tuning. By anchoring the model to specific mural-restoration features, we have successfully transformed a stochastic diffusion process into a robust and reproducible virtual-restoration tool, effectively suppressing unreasonable hallucinations while preserving high restoration fidelity.

4.2.3. Virtual Restoration Results

Figure 16 shows the results after virtual restoration. As previously stated, the NIQE and MUSIQ metrics were introduced for quantitative assessment. The quantitative evaluation results of each image obtained through experiments are shown in Table 8.

Table 8. Quantitative Evaluation of Virtual Restoration Effects of Murals Based on NIQE and MUSIQ.

Image	NIQE	MUSIQ
bottom	3.9748	56.5122
top	8.1514	57.6384
east	16.8712	47.7858
west	15.7483	47.3499

Based on the data presented in Table 8, the restoration performance shows noticeable variations across different mural areas. For the bottom and top regions, the model achieves superior performance, characterized by significantly lower NIQE scores and higher MUSIQ scores. These indicators show that the model successfully performed high-quality structural and texture reconstruction, which significantly enhanced the naturalness and overall perceptual quality of the images.

Conversely, the eastern and western images show less optimal NIQE and MUSIQ values. This indicates that in these specific areas, the model encountered challenges in providing effective semantic completion for the damaged sections, leading to less natural textures and a lower subjective quality rating. These results emphasize that the complexity of the mural's original content and the nature of the damage significantly influence the effectiveness of the generative restoration.

However, restoration based on diffusion models has inherent randomness. Results are influenced by several factors, including prompts, sampler settings, denoising strength, and random seeds. Since different parameter combinations can lead to significantly different results, achieving the optimal outcome often requires multiple iterations.

To evaluate the model's applicability, this study analyzed areas with suboptimal virtual restoration results, as shown in Figure 17. The main failure scenarios are:

- (1) Excessive texture generation: In large-scale peeling areas lacking contextual reference, the model occasionally generates redundant textures inconsistent with the surrounding murals.
- (2) Color discontinuity: When prompt constraints are insufficient, or the noise-reduction intensity is set too high, the edges of the repaired areas may exhibit a non-smooth color transition relative to the original image.

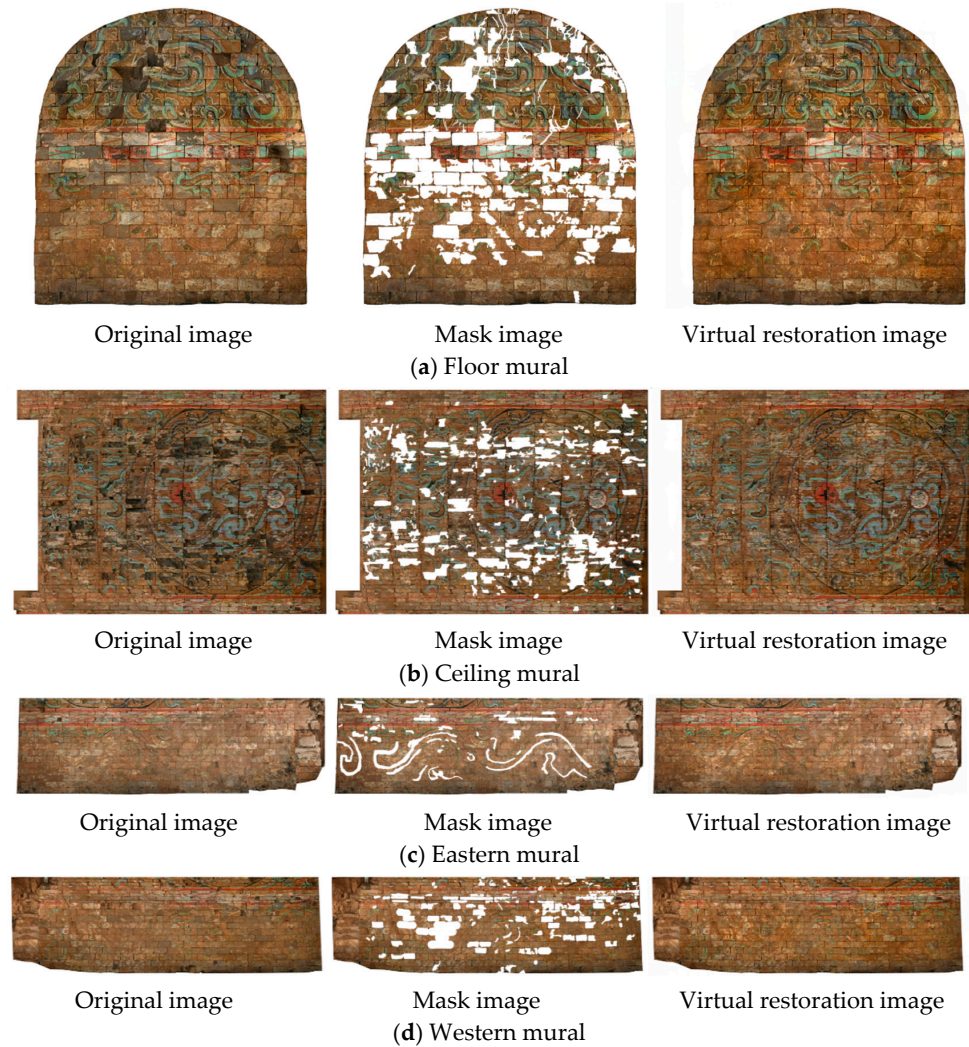


Figure 16. Virtual restoration result image.



Figure 17. Cases of model repair failure and defects.

In practical operations, these failure cases usually require manual intervention for correction, such as adjusting noise-reduction parameters or re-generating the seed.

4.2.4. Integrated System Design

(1) Overall design and workflow mechanism

Based on the verification of the proposed models in the previous studies, this study develops an integrated digital restoration workflow prototype to assist in mural-protection decision-making. The system is developed in Python 3.8. Each module is designed independently. Moreover, the core recognition and virtual restoration models are locally deployed and can be efficiently invoked via official interfaces.

The system's standard workflow comprises four functional modules and data-transfer paths:

- (1) System Initialization Module: This module initializes the model and extracts raw data. Users place the images to be processed in the designated folder. Subsequently, the system retrieves the original images from the database or the specified path.
- (2) Damage Area Identification Module: This module calls the local prediction model to cut, fully automatically label the images, detect the damage area, and output the identified result images and binary mask files.
- (3) Damage Intelligent Repair Module: Seamless connection is achieved through direct data transfer between modules. Once the model finishes annotation, the binary mask file and the original image are jointly transferred to this module. By importing the workflow JSON file to call the repair model, feature reconstruction is carried out only at the damaged positions of the original image, based on the mask and preset parameters.
- (4) Repair Result Output Module: This module integrates the identification and repair results and finally outputs the virtually repaired image in the specified format.

Regarding hardware specifications and execution efficiency, due to the intensive tensor calculations involved in generating diffusion models, the system demands a high-end video-memory configuration (recommended minimum: 4 GB). With the hardware configuration used in this study, it usually takes 30 min to complete the recognition and virtual restoration of a single image.

(2) Influence mechanisms of different injury types on generation and repair

This system primarily focuses on the three most representative types of mural damage: cracks, peeling, and brick damage. For linear damage, such as cracks that disrupt the continuity of the mural patterns, the virtual restoration process focuses on inferring geometric structures.

Specifically, the generative model performs linear interpolation and structural alignment based on the edge information at both ends of the cracks. Surface damage, such as peeling and brick damage, typically entails large-scale color and texture loss. Therefore, the virtual restoration process centers on synthesizing semantic textures. In this case, the generative model refers to the style features of non-damaged areas to complete the missing parts naturally.

This system adopts an aggregation strategy. Considering the actual requirements of mural protection, it prioritizes dealing with common damages that significantly impact visual presentation. However, for extremely minor or uncommon special damages, the system does not perform automatic identification or intervention.

(3) Spread and limitations of incorrect segmentation

In the system, segmentation errors are directly transferred to the virtual-restoration stage. At present, the research has not comprehensively considered the secondary correction

of segmentation errors. Specifically, the repair stage is unable to automatically identify and rectify the error masks originating from the segmentation stage.

This method represents a current limitation of the research. We have addressed this in the conclusion and outlook section. Future research could focus on this aspect.

(4) Coupling relationship between detection and generation

The two demonstrate a complementary and collaborative relationship, jointly forming a complete mural-restoration system. Neither can be omitted. The segmentation results guide the virtual restoration work to be carried out only in the damaged areas.

This method prevents any impact on the undamaged parts of the original image, thereby improving the accuracy and speed of virtual restoration.

(5) Semi-automated human-machine collaborative interaction mode of the system

In practical applications, this system adopts a semi-automated human-machine collaborative architecture. During system initialization and damage identification, the operations are fully automated.

However, in the virtual repair stage, users need to manually adjust the repair parameters to achieve better results. Due to the inherent randomness in the output of the diffusion model, the system automatically generates multiple candidate repair outcomes. Then, users are required to choose the highest-quality candidate from these options.

(6) Generalization ability of the workflow and deployment in new domains

Regarding the system's migration capability in a completely new scenario, the process and architecture design of this system are repeatable. However, due to the objective laws of the data-driven model, this system cannot directly achieve the desired effect without additional fine-tuning on a completely new and distinct mural dataset.

The core AI models integrated into the system, namely, the DeepLabv3 segmentation network and the specific mural LoRA model, lack strong cross-domain generalization capabilities. To deploy the system for a new mural project, it is essential to gather new data, re-label it, and carry out a certain degree of transfer fine-tuning on these two models to adapt to the features of the new scenario. This represents a common hurdle that contemporary deep learning-based visual systems have to overcome.

Furthermore, it is crucial to note explicitly that, without careful consideration, the current results cannot be simply extrapolated to other scenarios. The reported performance metrics rely significantly on the specific features of the current dataset. To verify the wider applicability of the proposed method, extensive additional validation using independent datasets is necessary in future research.

5. Conclusions and Prospects

This research mainly focuses on the detection of mural damage and digital restoration work. Through the integration of deep learning and diffusion models, a preliminary exploration has been conducted across the entire workflow, from automatic damage identification to local virtual restoration. The main conclusions are as follows:

- (1) The semantic segmentation model can identify various types of mural damage. Test results show that the model demonstrates preliminary feasibility in distinguishing between minor cracks and large-scale peeling, but classification accuracy in complex backgrounds still needs improvement.
- (2) A visual restoration process was established based on ComfyUI and SD, and fine-tuning was performed using LoRA to adapt the model to the specific style of murals. Experimental results demonstrated the potential of generative models for virtual restoration, enabling the generation of textures with similar tones.

- (3) Automatic recognition and local repair have been successfully integrated into a single process, thus forming a feasible, comprehensive workflow. These achievements not only provide heritage protection experts with a tool for assessing damage conditions and formulating repair plans but also offer a framework for cultural research.

The study focuses on ancient tomb mural heritage, a crucial medium for research in architecture, cultural heritage, history, and conservation. As a result, the scope of this study is strictly confined to architectural cultural heritage and historical building conservation. While the study is rooted in architectural heritage conservation, its methodological contribution is centered on an AI-driven workflow. This research explores how computer vision can be integrated into the documentation phase of architectural pathology.

This study's starting point, application scenarios, and ultimate goal all closely revolve around identifying architectural relic damage, conserving mural architectural heritage preventively, and protecting the masonry tomb architecture itself. The research core lies in "damage characterization" and "virtual restoration effect prediction". In actual heritage-protection projects, clearly defining the restoration goals (i.e., the visual and physical states to be achieved after restoration) is a prerequisite for formulating intervention plans.

This study combines the methods of mural damage identification and digital restoration. Utilizing the assessment of building damage morphology and computer vision technology, this approach is applicable to the identification and digital restoration of mural damage affixed to common building structures, such as bricks and concrete. Specifically, this approach is suitable for protecting murals in architectural heritage sites under poor preservation conditions and with complex damage types.

The intelligent recognition and virtual restoration system developed in this study has significant core application value:

- (1) Damage quantification and initial intervention reference: The high-precision damage mask generated by DeepLabv3 can serve as a digital record of the surface damage situation. By analyzing the spatial distribution of indicators such as linear cracks or masonry loss, this system provides intuitive morphological references, which can help experts identify high-risk areas and coordinate further professional structural inspections.
- (2) Digital restoration simulation for decision support: The virtual restoration based on Stable Diffusion and LoRA provides a visual simulation environment that adheres to the "minimum intervention" principle. Before any irreversible physical restoration is carried out, this workflow allows the protection team to intuitively preview the possible results under different virtual restoration intensities, thereby providing an auxiliary reference basis for reducing the risks brought about by excessive intervention.

Although certain progress has been made, there are still some issues that need to be addressed in future studies.

- (1) Although this study improved the evaluation method by introducing the NIQE and MUSIQ indicators, which, to some extent, enhanced the objectivity of the assessment, it should be recognized that the repair quality of the system remains highly reliant on manual parameter adjustment and result screening, demonstrating a prominent semi-automated characteristic.
- (2) The lack of professional verification based on cultural heritage experts is an important limitation of this study. Future research will focus on establishing a protection evaluation mechanism that integrates expert opinions to verify the scientificity and applicability of this technology in real cultural heritage protection projects.
- (3) The output of diffusion models inherently exhibits variability. Thus, repeated attempts are needed to achieve optimal results. In future research, architecture could be optimized to make the virtual restoration process more stable and automated.

- (4) Future research should focus on enhancing the generalization ability of the model on various independent datasets. It should also concentrate on verifying the model's generalization ability in complex real-world scenarios. A more comprehensive analysis of the model's uncertainty should be conducted together with larger-scale verification.
- (5) During the virtual-restoration stage, the segmentation error has not been corrected. In the future, a feedback mechanism should be introduced to evaluate the quality of the virtual restoration stage. This, in turn, will guide further optimization of the segmentation mask in a reverse direction.
- (6) The limitations of the dataset remain the main drawback of the research. Future work should focus on conducting robustness analysis across multiple dimensions and performing external validation on different datasets.

Author Contributions: Conceptualization, C.R. and W.T.; methodology, C.R. and W.T.; software, D.Y.; validation, D.Y.; formal analysis, D.Y. and W.T.; investigation, C.R.; data curation, C.R. and W.T.; writing—original draft preparation, C.R. and D.Y.; writing—review and editing, C.R. and W.T.; visualization, D.Y. and W.T.; supervision, Y.T. and Q.W.; project administration, Q.W. and P.W.; funding acquisition, Y.T. and P.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the State Key Laboratory of Green Building (Project No. LSZZ-Y202611); the Shaanxi Natural Science Basic Research Program (2025SYS-SYSZD-068); the National Natural Science Foundation of China (52578378); the National Natural Science Foundation Project (52578245); and the Shaanxi Natural Science Basic Research Program (2025JC-YBMS-464).

Data Availability Statement: The data presented in this study is available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Jia, M.; Hu, J.; Yang, Z.; Liu, W.; Qi, J.; Chen, B. Automatic Restoration of Dunhuang Murals and Process Visualization Method Based on Deep Learning. *Appl. Sci.* **2025**, *15*, 1422. [[CrossRef](#)]
2. Xu, Y.; Li, Y.; Zheng, X.; Zhang, Q. Computer-Vision and Machine-Learning-Based Seismic Damage Assessment of Reinforced Concrete Structures. *Buildings* **2023**, *13*, 1258. [[CrossRef](#)]
3. He, X.; Xu, M.; Zhang, H.; Zhang, B.; Su, B. An exploratory study of the deterioration mechanism of ancient wall-paintings based on thermal and moisture expansion property analysis. *J. Archaeol. Sci.* **2014**, *42*, 194–200. [[CrossRef](#)]
4. Cather, S. (Ed.) *The Conservation of Wall Paintings: Proceedings of a Symposium Organized by the Courtauld Institute of Art and the Getty Conservation Institute, London, UK, 13–16 July 1987*; The Getty Conservation Institute: Marina del Rey, CA, USA, 1991.
5. Liu, P. Mural heritage protection and digital restoration based on T-UNet and attention mechanism. *Proc. Inst. Civ. Eng. Eng. Hist. Herit.* **2025**, *179*, 23–37. [[CrossRef](#)]
6. Haji Sadeghi, N.; Azizi-Bondarabadi, H.; Correia, M. Preventive Conservation of Vernacular Adobe Architecture at Seismic Risk: The Case Study of a World Heritage Historical City. *Buildings* **2025**, *15*, 134. [[CrossRef](#)]
7. Lazaridis, P.C.; Kavvadias, I.E.; Demertzis, K.; Iliadis, L.; Vasiliadis, L.K. Structural Damage Prediction of a Reinforced Concrete Frame under Single and Multiple Seismic Events Using Machine Learning Algorithms. *Appl. Sci.* **2022**, *12*, 3845. [[CrossRef](#)]
8. Abd-Elhamed, A.; Alkhatib, S.; Abdelfattah, A.M.H. Prediction of Blast-Induced Structural Response and Associated Damage Using Machine Learning. *Buildings* **2022**, *12*, 2093. [[CrossRef](#)]
9. Wu, L.; Zhang, L.; Shi, J.; Zhang, Y.; Wan, J. Damage detection of grotto murals based on lightweight neural network. *Comput. Electr. Eng.* **2022**, *102*, 108237. [[CrossRef](#)]
10. Chen, Y.; Zhang, A.; Shi, J.; Gao, F.; Guo, J.; Wang, R. Paint Loss Detection and Segmentation Based on YOLO: An Improved Model for Ancient Murals and Color Paintings. *Heritage* **2025**, *8*, 136. [[CrossRef](#)]
11. Wang, Y.; Xiao, M.; Hu, Y.; Yan, J.; Zhu, Z. Research on Restoration of Murals Based on Diffusion Model and Transformer. *Comput. Mater. Contin.* **2024**, *80*, 4433–4449. [[CrossRef](#)]
12. Tang, X.; Sui, Y.; Sun, K.; Xiang, L. DiffInpaint: Line Drawing Guided Murals Restoration with Diffusion Model. *Measurement* **2026**, *258*, 119223. [[CrossRef](#)]

13. Ryu, S.; Lee, J.; Kang, Y.J. Flexural Behavior of Concrete-Filled Steel Tube Beams with Corrugated Webs. *Buildings* **2023**, *13*, 317. [[CrossRef](#)]
14. Hacıfendioğlu, K.; Altunışık, A.C.; Abdioğlu, T. Deep Learning-Based Automated Detection of Cracks in Historical Masonry Structures. *Buildings* **2023**, *13*, 3113. [[CrossRef](#)]
15. Swathi, B.; Rao, D.B.J. Automated image inpainting for historical artifact restoration using hybridisation of transfer learning with deep generative models. *Sci. Rep.* **2026**, *16*, 4810. [[CrossRef](#)] [[PubMed](#)]
16. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [[CrossRef](#)] [[PubMed](#)]
17. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv* **2018**, arXiv:1802.02611. [[CrossRef](#)]
18. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [[CrossRef](#)]
19. Meng, C.; He, Y.; Song, Y.; Song, J.; Wu, J.; Zhu, J.-Y.; Ermon, S. SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. *arXiv* **2021**, arXiv:2108.01073. [[CrossRef](#)]
20. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-Resolution Image Synthesis with Latent Diffusion Models. *arXiv* **2021**, arXiv:2112.10752. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.