Heritage Science

Open Access

The digital reconstruction of degraded ancient temple murals using dynamic mask generation and an extended exemplar-based region-filling algorithm

V. Rakhi Mol[*] and P. Uma Maheswari

Abstract

A mural is any piece of artwork sculpted or applied directly on a wall, ceiling or other permanent surface. This artwork symbolizes various culture's, traditions, historical events, spiritual stories, and civilizations of respective societies of ancient times. But these mural paintings are subjected to degradation either by various natural causes as well as pollution or by human beings without knowing their value. Restoring these paintings requires skilled artisans who are hard to find these days. Consequently, an efficient image restoration technique is required to meet the particular needs of the paintings. Existing in-painting algorithms largely use pixel-based textural reconstruction. The technique, however, does not work well for images with large, degraded portions. and also fails in the restoration of the structure. To resolve these drawbacks, we propose a combined technique for the textural and structural reconstruction of ancient murals. The proposed Extended Exemplar-based Region-Filling Algorithm uses a patch-based reconstruction procedure and masked images are created automatically using the Dynamic Mask Generation Algorithm. The deteriorated portions are identified by creating masks, and masks are created in such a way that degraded portions have a pixel intensity value of one and the remaining part has a value of zero, and filling is done by analyzing the surrounding pixel values of the degraded pixel. The algorithm reconstructs the structure of the paintings efficiently by generating sketches. The proposed technique reconstructs both the structure and textural information, and ensures efficient reconstructed results, compared to existing in-painting techniques. Performance is evaluated by metrics such as the Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index (SSIM).

Keywords: Image reconstruction, Image inpainting, Masked image, Digitization, Texture synthesis, Structure synthesis

Introduction

A mural painting [1] is any bit of work of art painted or sculpted straightforwardly on a wall, roof, or other perpetual surface. Indian temples are known for their rich content of heritage in various forms; especially Tamilnadu is known for its uniqueness and cultural highlights. Ancient temple murals symbolize various cultures, traditions, and civilizations of respective societies of ancient times and bring virtual screens of past events to the younger generation to learn, something that can never be learned by reading history. These paintings are hard to maintain. Ancient mural paintings are exposed to an assortment of distortions brought about by natural impacts, changes in temperature and humidity, as well as contamination and by human ignorance to preserve the monuments without knowing their value. Also, some of them have faded over time and some of them have developed cracks and dirt patches over them. in-order, to preserve and conserve the depth of knowledge and the

*Correspondence: rakhirajeev92@gmail.com
Department of CSE, College of Engineering Guindy, Chennai, India

aesthetic beauty of paintings, inscriptions, monuments, and sculptures for future generations to experience, the restoration of mural paintings is very essential.

The reconstruction of murals has been happening in the past two decades and is still going on. The artists of Tamil Virtual Academy, a part of the Tamil Nadu Archaeology Department, manually reconstructs the deteriorated temple murals by using paint and brush. But this is a time-consuming task and the availability of skilled artisans reduced while the demand for reconstruction rose. Digital inpainting has become popular with the improvement of image processing tools. The capability of these processing tools led to the availability of different methods for image processing. Image inpainting [2] has become a challenging topic of research in image processing [3] and computer vision.

This work proposes a digitized strategy to restore ancient, degraded murals. Traditional restoration with the assistance of talented artists is a tedious process. This digitized model efficiently identifies the degraded patches and restores them more fastly than conventional restoration. There are different types of degradation, though this work chiefly deals with regular or irregular holes, dirty patches, darkness, degraded edges, and noise in these ancient temple murals. The identification of degraded patches plays a critical role in their reconstruction. In this work, degraded patches are identified by manually analyzing the pixel intensity value of degraded portions seen in ancient temple murals. It seems that the degraded regions have high pixel intensity values in the range of (199,199,199) and (254,254,254). The intensity value of degraded portions is not exactly white and it has a pixel intensity value lesser than (255,255,255). In these bases, we extracted the R, G, B components and finally reached a standard that if the mean value of r,g,b components lies in between (0,30) are marked as the degraded portions. The proposed technique ensures an efficient structural and textural reconstruction of ancient temple murals with the help of digital image processing techniques and artificial intelligence for portraying the heritage and wisdom hidden in the ancient mural paintings to the public and foreign visitors. In sum, the contributions of this paper include

– Proposed a novel technique for the identification of degraded parts in the ancient murals. Automatic detection of degraded parts in the ancient temple mural is a very challenging task. The creation of masked images is very important in the synthesis of deteriorated portions. The manual creation of masks gives better result but, it is a tedious and time consuming process. In the case of regular holes it is very easy to generate masks manually but in the case of dirty patches, irregular holes and noise it may be difficult to create masks manually. The proposed Dynamic mask generation algorithm effectively identifies the degraded patches by analyzing the degraded pixel values.

– Developing a novel strategy to conserve the structural properties of ancient murals through sketch enhancements and edge conservation.

– Restoring the color value of murals in terms of brightness and illumination and producing an amplified version of the same.

– Also introduces an effective patch matching strategy for restoring the texture of the murals by analyzing the surrounding patches of the degraded regions. The filling is done based on the priority of the pixels in the degraded regions. This proposed method effectively restores noises, dirty patches, larger holes, texts seen in the ancient murals.

– Comparing it with state-of-the-art image inpainting approaches on certain murals and real images. The final results illustrate the superiority of the proposed method in terms of accuracy and efficiency.

The remainder of the paper is organized as follows. Section "Related work" describes the related work in the field. Section "Materials and methods" explains the dataset, masked image generation, and reconstruction of degraded temple murals. The metrics used for evaluation and the results obtained are provided in Section "Experimental results". Finally, the conclusion is given in Section "Conclusion".

Related work

A great deal of research has been carried out on the topic of reconstruction of paintings and research in this area is still ongoing. The image inpainting techniques are broadly classified into two categories based on the methodology used. Sequential approach and deep learning-based approach. This section summarizes the various works that have concentrated on the restoration of degraded paintings.

In the case of degraded mural paintings seen in ancient temples, manual reconstruction criteria are usually followed. The temple murals seen in ancient temples of Tamilnadu are reconstructed manually by artists of the Tamilnadu Archaeology Department, but in the case of manual reconstruction, only the structure is reconstructed, and not the texture. This is a time-consuming process and hence nowadays, digital reconstruction techniques are used.

Mask generation techniques

Most inpainting algorithms involve identifying pixels that are to be modified. Most of the techniques specify the regions prior with the use of image masks. A mask is a black and white image where the pixels with white color needs some modification while the pixels with black colour needs no change. These masks act as a hyper-parameter to the algorithm using which it decides which patch to fill next. Techniques which knows prior which pixels are damaged are called non-blind techniques. There are very less methods which can in-paint an image without specifying any masks. Such methods are called blind in-painting techniques. Automatic mask generation algorithm makes use of contours and segmentation [4] to decide which pixel needs change and in-paint them using existing in-painting algorithm. The algorithm needs the original image to make comparison with to decide which patch needs make over. Most of the in-painting algorithm [5, 6] uses manual identification of degraded parts or uses advanced computer application tools such as photoshop for creating masks.

Cracks are common in paintings. Many strategies are utilized for the identification and removal of cracks in the old mural images. Giakoumins et al. [7] presented a technique for the discovery and evacuation of cracks in paintings. They utilized a top hat transform strategy for crack identification. It is a type of grayscale morphological filter. Afterward, a crack detachment measure is done. Subsequently, painting filling is carried out by utilizing suitably altered order static filters. A two-stage restoration technique was developed by Karianakis and Maragos [8]. The first stage is the morphological feature location, here first recognizes the degraded portions by finding the boundaries, this is accomplished by utilizing the canny algorithm. The subsequent stage utilizes a Total Variation in-painting algorithm. This technique adequately restores the small degraded portions but it did not give a powerful outcome for larger degraded areas. A combined bottom-hat morphological operation and median filtering by Vaghela et al. [9] is used for the identification and removal of cracks from digitized paintings and Araujo and Hadad [10] developed some enhanced optimal filtering methods for crack detection and edge smoothing.

Sequential in-painting

A search algorithm using the Markov Random Field (MRF) was proposed by Ruzic and Pizurica [4] is a popular patch-based method. An MRF is a graph whose nodes represent random variables and edges represent the relation between them. MRF can be an effective technique for image denoising, image restoration and image segmentation. Manikanta et al. [11] developed an efficient method for restoring mural images. This method uses a four-stage restoration technique. The first stage distinguishes each line in the picture after which the k-means clustering method cluster the degraded parts in the image. The third stage groups every pixel and replaces each pixel with its mean value lastly takes the weighted average as the reconstructed image. Pappas et al. [12] utilized a few methods for the reproduction of painting artworks. The procedures incorporate sample mean matching, linear approximation, Iterative closest point approximation, white point transformation, and RBF approximation. Among these strategies, linear approximation and white point transformation were able to produce an efficient result compared to other techniques. These methods created worthy outcomes and they have little computational necessities.

One of the widely used methods for restoration is the patch matching technique [13]. This technique partitions the input image into some overlapping patches. Patches are rectangles of a fixed size, say 7×7 pixels; they decide the correspondence between the pixels of an image. Likewise, a hybrid strategy by consolidating the patch [14] matching method with Markov Random Field was introduced by (MRF) [15] for texture reconstruction. In the first phase, paintings go through a color contrast upgrade stage. In this stage, color values are reestablished after the Lacuna texture synthesis is done, first recognizing the degraded region and then the matching patch is distinguished by neighborhood searching. This strategy keeps up the complete shape of the paintings and avoids edge disengagement in the resultant image. An angle-aware patch matching technique [5] is used to reconstruct degraded parts. This method uses the prior knowledge for surface fitting and uses the Jaccard similarity coefficient for finding the coordinating exactness between patches.

Deep learning-based approaches

A blind image inpainting method (BICNN) based on convolutional neural networks was proposed by Cai et al. [16] which can in-paint without any prior masks. Its has cascading layers of CNN that learn the end-to-end mapping between corrupted and ground truth pairs. Sasaki et al. [17] used an encoder-decoder-based method for completing the gaps in line drawings. This can help finish the incomplete details in scanned documents. Text removal was proposed by Nakamura et al. [18], also using the CNN. Alilou and Yaghmaee [19] proposed a non-textured inpainting method using the CNN and a generalized regression neural network (GRNN). While most CNNs work similarly, the parameters and the order in which the layers are arranged matters most as to how the results are obtained. Fine-tuning the parameters and

finding the right hyperparameters also decide on the final model.

Though a lot of work is witnessed in the restoration of images. Most of the image inpainting techniques are targeted at natural and real images like faces, sceneries, etc. There is less work targeted at restoring old paintings. Also, most of these works concentrated on crack removal and reconstruction of small degraded portions. These works use manual identification of the degraded parts and fail to restore the structure and texture information if the percentage of degradation is higher. Deep-Learning based inpainting is a complex problem so for better performance it requires atleast 1000 training cases. It also requires ground truth images for one-to-one mapping. In the case of ancient temple murals, most of the paintings are highly degraded state and few of them are available for restoration, which is not sufficient to train a deep learning technique. Also, ground-truth images of degraded murals are not available. To address these

drawbacks, we propose a sequential automated method to identify and reconstruct degraded portions in ancient temple murals. The proposed technique restores both the structure and texture of degraded murals efficiently, including large, degraded portions as well.

Materials and methods
Dataset collection
The dataset consists of about 300 ancient temple murals. Most of them collected from famous temples of Tamilnadu, such as Varadharaja Perumal Temple, Brihadeeswarar temple, and so on from Kancheepuram. Most of these mural paintings are in a degraded state, so the digital acquisition facilitated with the help of high-resolution digital thermal cameras, it is depicted in Fig. 1. Capturing degraded temple murals with standard digital cameras can again introduce noise and other types of irregularities. This can be avoided to some extent, by using digital thermal cameras. A few of the degraded
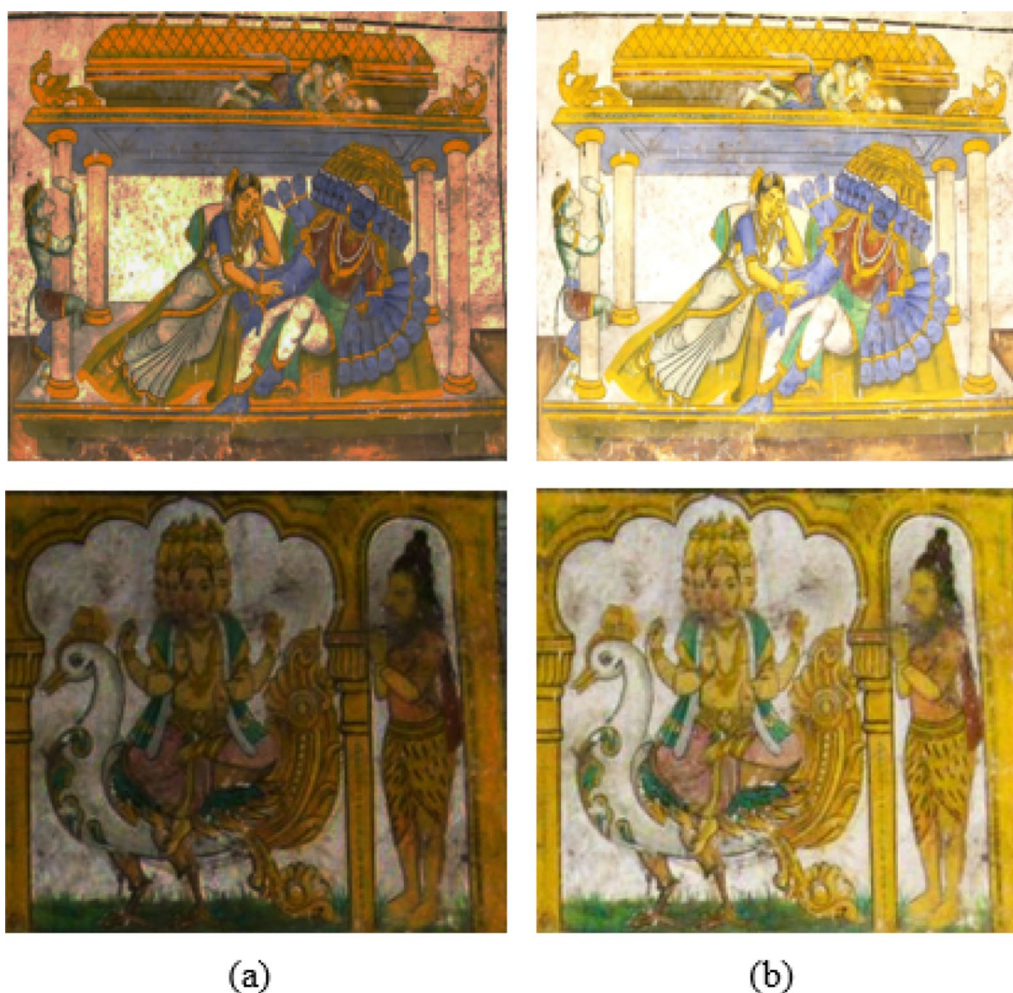
Fig. 1 a Depicts the degraded murals captured using normal digital camera and (**b**) by using digital thermal camera

and ground truth mural paintings have been collected from the official website of the Tamil virtual academy. It is an official website of the Tamil Nadu Archaeology Department. It consists of mural paintings collected from famous temples of Tamilnadu such as Meenakshi Amman Temple, Arulmigu Arunachaleswarar Temple, Sri Ranganatha Swamy Temple, Srirangam, Shri Meenakshi Amman Temple, Arulmighu Mariamman temple Samayapuram, and so on. Some of the degraded sample murals are depicted in Fig. 2.

This system aims to reconstruct degraded temple murals and archive them in their original glory for further research and common use. This work uses high-resolution murals. We have tested our system with sets of various pixel densities. Since most ancient temple murals are in a highly degraded state our testing should show a pixel density of 220 ppi(pixel per inch) has been a proper threshold. The pseudo code for super-resolution using learning method is described in Algorithm 1. Images with pixel density less than 220 ppi will look very blurred hence it should be converted to high resolution images before applying this technique Fig. 3. The block diagram of the entire system is shown in Fig. 4.

To reconstruct the degraded mural image, the input image undergoes several phases such as preprocessing, degraded patch detection, masked image creation, and finally reconstruction.

Preprocessing
Preprocessing aims to address the irregularities in the image, which enhance the quality of the picture information and suppress the undesirable disturbance or upgrade some image features significant for further processing. Mural images are subjected to a lot of noise and distortions either by various natural resources or by human beings. Also, in some cases, the acquisition of images also
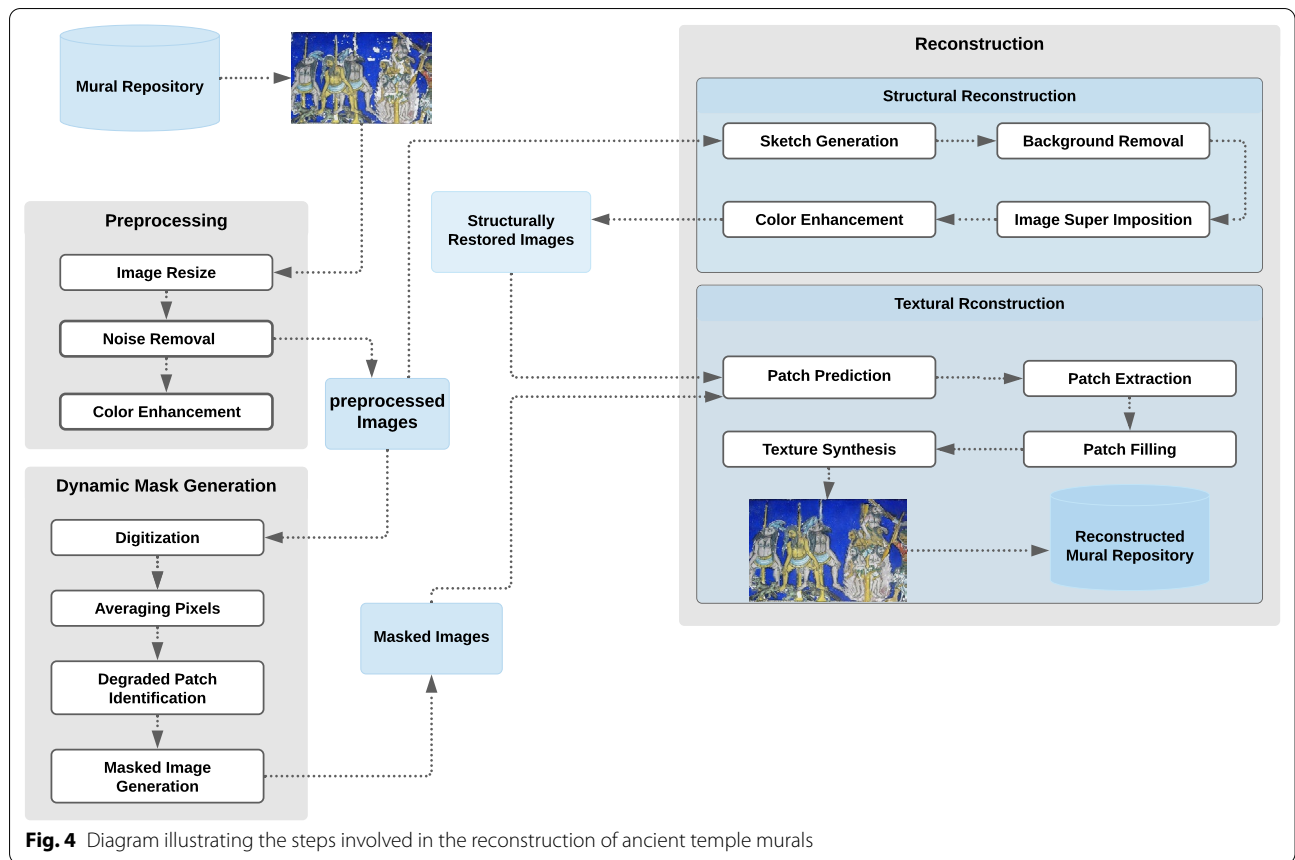

Fig. 2 Sample Temple Mural Images

Fig. 3 a Represents low-resolution mural and **b** represents high-resolution mural

Fig. 4 Diagram illustrating the steps involved in the reconstruction of ancient temple murals

introduces noises and disturbance. So, the mural image must undergo some cleaning process for acquiring a better final output. Preprocessing includes image resize, noise removal, and image enhancement. Mural images are of different sizes. So, it is necessary to convert the input parameters into a similar data distribution for the ease of processing it. This method converting the murals into a unique fixed size of (256 * 256).

Algorithm 1: Super-resolution of degraded temple murals

Input	: Low Resolution Degraded Murals $Mural_i, i \in \{1, 2, 3, ..m\}$
Output	: High Resolution Murals , $HRmural_i$, $i \in \{1, 2, ..., m\}$
Methodology:	Super-resolution of degraded murals

1   **for** *int i in* 1 *to* m **do**
2   Resize each image $Mural_i$ using bicubic Interpolation.
3   Learning Phase:
4   Training using existing high-resolution images
5   Patch extraction:
6   Extract overlapping dense patch.
7   Represent each patch as a high dimensional vector.
8   Non Linear Mapping:
9   Mapping into another higher dimensional vector.
10   Restoration:
11   Average the high Resolution patches.
12   Testing Phase:
13   Pass the randomly chosen low resolution mural image $Mural_i$ to produce the high-resolution $HRmural_i$ image.
14   **end**

After resizing noise removal is carried out. This phase mainly concentrates on the removal of irregularities and disturbances present in the image. Noise [20] is actually a slight variation in the pixel intensity of an image. These changes occurred at the time of acquisition or were introduced at the time of transmission. This type of noise and irregularities strongly affects the quality of the image and hide useful information regarding the image; so it is essential to reduce this noise, in order to preserve the essential information about the image. In this work, we are mainly concern with gaussian noise, and its removal is done using bilateral filter [21]. It is a non-linear filter, which replaces the intensity of every pixel with a weighted normal of intensity values from the encompassing pixels. This filter gives a better result compared to the Gaussian and median filter and it also preserves sharp edges. It is an edge-preserving and noise-reducing smoothing filter. It is represented by Eq. 1. Figure 5 depicts a sample noise removed mural using bilateral filter.

$$I_{filtered}(y) = \frac{1}{W_p} \sum_{y_i \in \Omega} I(y_i) f_r(\|I(y_i) - I(y)\|) g_s(\|y_i - y\|)$$
(1)

and normalization term W_p is given by Eq. (2)

$$W_p = \sum_{y_i \in \Omega} f_r(\|I(y_i) - I(y)\|) g_s(\|y_i - y\|)$$
(2)

where

$I_{filtered}$ is the noise removed image
I is the original input image to the bilateral filter
y are the current pixel coordinates to be filtered
Ω is the window focused in y, so $y_i \in \Omega$ is another pixel
f_r is the span kernel for smoothing differences in intensities
g_s is the domain kernel for smoothing differences in coordinates.

Then noise-removed murals are subjected to the normalization phase; the purpose of normalization is to enhance the image in terms of brightness and contrast estimations of each pixel in the picture. Brightness refers to the overall lightness or darkness of the image. Increasing the brightness makes each pixel in the image get brighter. Increasing the contrast value makes the bright areas appear as light and dark regions look more obscure. Adjustment of brightness and contrast values helps to distinguish an object from other objects and the background. Here the brightness and contrast value is adjusted to produce an enhanced image to identify the deteriorated portions smoothly.

Fig. 5 **a** The original RGB image and **b** indicates the image obtained after bilateral filtering

---

Algorithm 2: Preprocessing of Degraded Temple Murals

Input	: Degraded Mural Images $Mural_i, i \in \{1, 2, 3, ..m\}$
Output	: Preprocessed Mural Images , $Preprocessed_i, i \in \{1, 2, ..., m\}$
Methodology:	Preprocessing of Degraded Temple Mural Images

1 **for** *int i* **in** 1 **to** *m* **do**
2   convert each image $Mural_i$ into YCbCr format.
3      $NewMural_i \leftarrow Mural_i$
4   Resizing $NewMural_i$ into benchmark values.
5      $Resized\_M_i \leftarrow NewMural_i$
6   // Noise Removal.
7      $nw\_Mural_i \leftarrow Resized\_M_i$
8      $blocks \leftarrow divide(nw\_Mural_i, 16)$
9   **for** *int j* **in** 1 **to** 1200*blocks* **do**
10          $block \leftarrow getblock(j)$
11          $index \leftarrow find(block == 0)$
12          $no\_zerobk \leftarrow size(index)$
13   **if** $no\_zerobk > 0$ **then**
14              $average \leftarrow AVG(block)$
15              $new\_Muralim \leftarrow replace(0, average, index)$
16   **end**
17   **end**
18   Enhancing brightness of each image $new\_Muralim$
19      $Preprocessed_i \leftarrow new\_Muralim$
20 **end**

Dynamic mask generation

The automatic detection of degraded parts in a mural image is a challenging task. Most of the previous works [6] manually identify the degraded portions. To detect the degraded portions automatically, this work uses an averaging pixel algorithm. The preprocessed mural image is given as the input to this algorithm. The first phase of this algorithm is the digitization of the input image. Image digitization is the process that converts a 2D image into its numerical representation. Digitization is carried out in two steps, sampling and quantization. In the sampling step, the analog image is converted into discrete values; that is, after sampling, the image is represented in the form of a matrix. For example, if an image I has a height of M and width of N, then after sampling, the image I is represented in the form of M*N matrix. Quantization is associated with the values in the generated matrix. It defines the intensity values corresponding to every pixel in the image; that is, the high-intensity area has high pixel values and the low-intensity area has lower pixel values. After digitization, the RGB component of each pixel of an image is extracted. If I is the image, after digitization, the image I can be represented by Eq. 3.

$$I(i,j) = \sum_{i=0}^{m} \sum_{j=0}^{n} f(i,j) \qquad (3)$$

where $I(i, j)$ is the image and $f(i, j)$ represents the intensity value at the point(i,j).

The next step is the extraction of r,g,b components of each pixel in image I; This algorithm uses the get pixel() function for extracting pixel values and it is stored in a list, so the 0th position indicates the r-value, 1st position indicates g value and the 2nd position corresponds to the b value. In this way, all the pixels with their r, g, b components are extracted and next is the analysis of these extracted pixel values. Generally, the degraded portion has different pixel intensity values. Here in the case of temple mural images, first, we manually analyzed the pixel values of the degraded portions. Generally, these degraded portions are part of a rock, ceiling, or wall. The manual analysis of degraded pixel intensity values shows a particular similarity; that is, it has high-intensity values that are in the range of (199,199,199) and (254,254,254) and the degraded portion does not have a pixel value of (255,255,255). It means the degraded portion is not white, it has some intensity value less than the intensity value of white color. The next step of this algorithm is the subtraction of r, g, b components of each pixel from 255 and obtaining a new r, g, b value; that is, the new r, g, b values of the image(I), is given in Eq. 4.

$$
\begin{aligned}
r_{new} &= 255 - r \\
g_{new} &= 255 - g \\
b_{new} &= 255 - b
\end{aligned}
\qquad (4)
$$

That is, each pixel of the image is subtracted from (255,255,255) and the average of the new r,g,b values is calculated using Eq. 5 for further processing.

$$avg = \frac{(r_{new} + g_{new} + b_{new})}{3} \qquad (5)$$

The algorithm works, based on the assumption that the average of the degraded pixel value is greater than zero and less than 30 (Fig. 6). Then we stored the start pixel
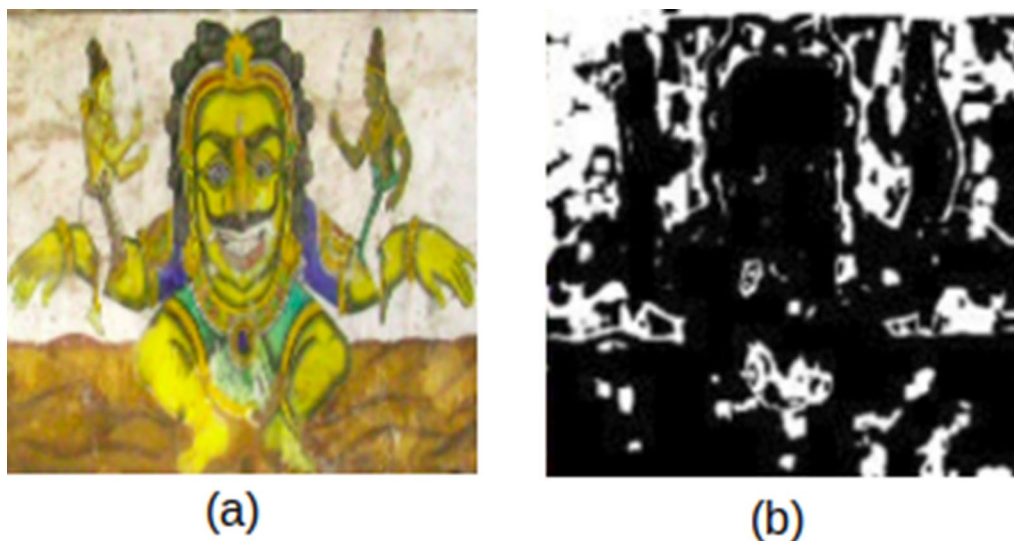
Fig. 6 The Degraded mural Images and **b** represent it masked image (**a**)

value and the end pixel value of the particular stream of a pixel that satisfied the above condition in an array and finally iterated the array and converted all the pixels in between the particular start and end pixel into white. The procedure is repeated for all the start and end pixel values, and all the remaining pixels in the image are set in black. Finally, the masked image is stored in a folder for further processing. Some masked images are depicted below Fig. 6.

Reconstruction of degraded temple murals using the extended exemplar-based region-filling algorithm

Exemplar-based algorithms [5, 6] mainly reconstruct the degraded portion by finding the most suitable exemplar, but sometimes it results in an incorrect texture filling when similar patches or exemplars are not available. Criminsci algorithm mainly focuses on removal of objects from an images. Objects are selected manually and masked images are created with the help of image processing tools. This technique does not concentrate on the enhancement of structure and colour. Also, this technique does not consider larger or irregular holes, dirty patches and noise. Also, this algorithm fails in reconstructing the structure of the image. To solve these issues, the proposed work uses sketch-based reconstruction criteria for structural restoration and an exemplar-based technique for textural restoration. This method identifies similar patches based on the confidence and priority value of patches. A patch is a small (generally rectangular) piece of an image. Due to the smaller size, image-processing algorithms are easier to operate on sub-images (patch) rather than operating on the entire image itself. The region-filling algorithm carries out reconstruction by assigning confidence and priority values for each pixel. In this algorithm, we are treating color images and so, each pixel in the mural images is associated with a color value or empty if it is not filled; this color value is considered as the confidence value and priority is the temporary value assigned during the algorithm. Based on the two terms, the algorithm finds the best match.

Algorithm 3: Masked Image Generation

Input   : Preprocessed Mural Images $Preprocessed_i, i \in \{1, 2, 3, ..m\}$
Output   : Masked Images,$Masked_j, j \in \{1, 2, ..., m\}$
Methodology: Automatic Mask generation using Averaging pixel

1 $flag \leftarrow true$, start=[],end=[],$mask\_im = []$
2 **for** $int\ i\ in\ 1\ to\ m$ **do**
3 $\quad$ **for** $int\ x\ in\ 0\ to\ width(Preprocessed_i)$ **do**
4 $\quad\quad$ **for** $int\ y\ in\ 0\ to\ height(Preprocessed_i)$ **do**
5 $\quad\quad\quad$ $px \leftarrow getpixel(x, y)$
6 $\quad\quad\quad$ $px = [r, g, b]$
7 $\quad\quad\quad$ $r \leftarrow px[0]$
8 $\quad\quad\quad$ $g \leftarrow px[1]$
9 $\quad\quad\quad$ $b \leftarrow px[2]$
10 $\quad\quad\quad$ $r_{new} = 255 - r$
11 $\quad\quad\quad$ $g_{new} = 255 - g$
12 $\quad\quad\quad$ $b_{new} = 255 - b$
13 $\quad\quad\quad$ $avg = \frac{r_{new} + g_{new} + b_{new}}{3}$
14 $\quad\quad\quad$ **if** $avg > 0\ and < 30$ **then**
15 $\quad\quad\quad\quad$ **if** $flag = true$ **then**
16 $\quad\quad\quad\quad\quad$ $start = (x, y)$
17 $\quad\quad\quad\quad\quad$ $end = (x, y)$
18 $\quad\quad\quad\quad\quad$ $flag \leftarrow false$
19 $\quad\quad\quad\quad$ **end**
20 $\quad\quad\quad\quad$ **else**
21 $\quad\quad\quad\quad\quad$ $end = (x, y)$
22 $\quad\quad\quad\quad$ **end**
23 $\quad\quad\quad$ **end**
24 $\quad\quad\quad$ **else if** $flag == false$ **then**
25 $\quad\quad\quad\quad$ $start.append(start)$
26 $\quad\quad\quad\quad$ $end.append(end)$
27 $\quad\quad\quad\quad$ $flag \leftarrow true$
28 $\quad\quad\quad$ **end**
29 $\quad\quad\quad$ **else**
30 $\quad\quad\quad\quad$ $Preprocessed_i.putpixel(x, y) = (0, 0, 0)$
31 $\quad\quad\quad$ **end**
32 $\quad\quad$ **end**
33 $\quad$ **end**
34 $\quad$ **for** $int\ j\ in\ 0\ to\ size(start)$ **do**
35 $\quad\quad$ **for** $int\ y\ in\ start[j+1]\ to\ end[j+1]$ **do**
36 $\quad\quad\quad$ $Preprocessed_i.putpixel(start[j], y) = (255, 255, 255)$
37 $\quad\quad$ **end**
38 $\quad$ **end**
39 $\quad$ $masked\_i \leftarrow Preprocessed_i$
40 **end**

Structural reconstruction

Structural reconstruction is carried out by enhancing sketches corresponding to each mural, M_i. During sketch generation, each mural M_i is converted into gray-scale image M_g. Then the generated gray-scaled images (M_g) are inverted by subtracting the image from 255. Inversion is done in such a way that dark areas in the input image become light and light areas become dark, inverted image M_{inv} is depicted by Eq. 6. Then the inverted image M_{inv} is smoothed using the Gaussian function. Then the smoothed image (M_s) is inverted to obtain the inverted blurred images($M_{s inv}$). Finally, the inverted blurred images are blended with the gray-scaled images(M_g) to obtain the sketches(M_{sk}) corresponding to each mural. In-order to enhance the sketches color dodge blend mode is applied, which separates the bottom layer from the inverted top layer. So that the edges in the images are sharpened and we have blurred the image so that the edges in the images are highlighted. The enhanced sketch images M_{sk}'s background is converted into transparent and it is superimposed with the degraded murals M_i to generate the structurally restored murals. The generated images are color enhanced for further processing. Figure 7 shows the images after structural reconstruction. This phase only restoring the edges and contours not the texture. So, structurally restored murals are fed as the input to textural reconstruction for getting fully restored murals.

$$M_{inv}(i, j) = 255 - M_g(i, j) \tag{6}$$
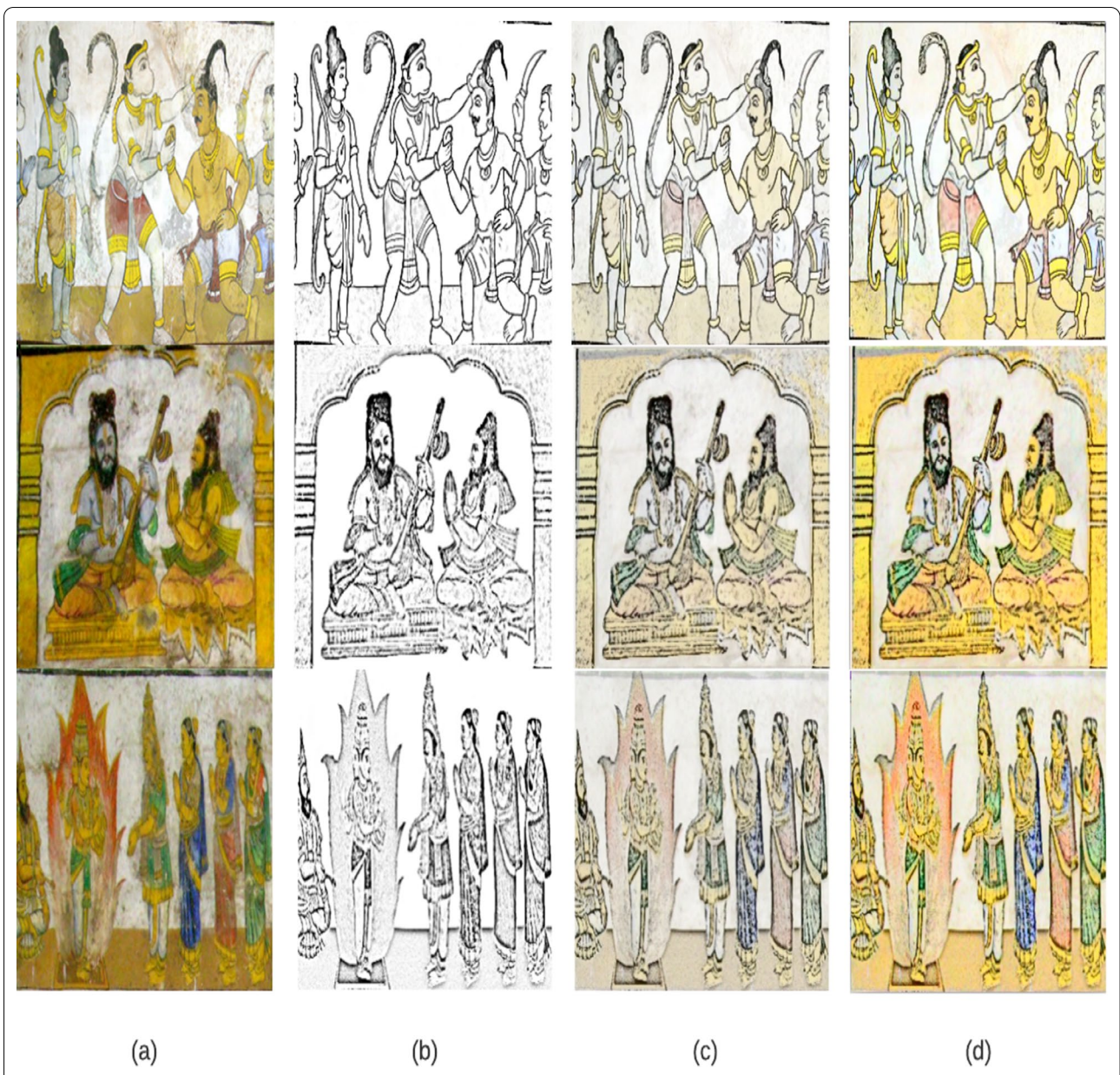
Fig. 7 **a** The original degraded paintings and **b** depicts the corresponding sketches generated and **c** structurally reconstructed images also, **d** depicts the color enhanced version of structurally reconstructed image

---

Algorithm 4: Structural Reconstruction of Degraded Temple Murals

Input	: Preprocessed Mural Images $Preprocessed_i, i \in \{1, 2, 3, ..m\}$
Output	: Restored Mural Images , $Restored_i, i \in \{1, 2, ..., m\}$
Methodology:	Structural Reconstruction of Degraded Temple Mural Images

1 **for** *int i* **in** 1 *to m* **do**
2   convert each RGB image $Preprocessed_i$ into gray-scale format.
3      $Gray_i \leftarrow Preprocessed_i$
4   Invert the gray-scale images $Gray_i$ to get a negative Inv_i.
5      $Inv_i(x, y) \leftarrow 255 - Gray_i(x, y)$
6   Apply a Gaussian blur to the inverted gray scale images
7      $b_i \leftarrow Inv_i$
8   The Blurred images b_i are inverted
9      $Binv_i(x, y) \leftarrow 255 - b_i(x, y)$
10   Blend the gray-scale image $Gray_i$ with the blurred negative $Binv_i$.
11     $sketch_i \leftarrow cv2.divide(Gray_i, Binv_i, 256.0)$
12   Convert $sketch_i$ background to transparent
13   Superimpose $sketch_i$ with the $Preprocessed_i$
14     $Restored_i \leftarrow cv2.addWeighted(sketch_i, 0.3, Preprocessed_i, 0.7, 0)$
15   Enhancing the color of each $Restored_i$ image
16 **end**

Textural reconstruction

Initially, we applied the Laplacian filter on the masked images so that it makes the black region negative and then these negative regions were filtered out. After this step, we superimpose the structurally reconstructed degraded mural images with their corresponding masked images. After superimposing, the degraded portions remain unfilled. Every pixel in a degraded image has a color value and a confidence value. Based on these data values, the filling priority is calculated; that is, every pixel in the degraded image is assigned a temporary priority value which determines the order in which the degraded pixels are to be filled. The priority calculation is biased toward those patches which (i)are on the continuation of strong edges and(ii) are surrounded by high-confidence pixels. Suppose *I* (see Fig. 8) is the degraded image to be reconstructed, Ω is the degraded portion that is the target region, Φ is the source region and $\delta\Omega$ indicates the target contour. Suppose for a patch, p_k is centered at the point k for some $k \in \delta\Omega$, then priority of *k* is given by Eq. 7, and

Fig. 8 Notation diagram

$C(k)$ and $D(k)$ represents the confidence and data value of pixel *k*. Eqs. 8 and 9 give the data and confidence value of *k*,

$$P(k) = C(k)D(k) \tag{7}$$

$$C(k) = \frac{\sum_{q \in P_k \cap (1-\Omega)} C(q)}{|P_k|} \tag{8}$$

$$D(k) = \frac{|\Delta I_k^{\perp} \cdot n_k|}{\alpha} \tag{9}$$

where $|P_k|$ is the area of P_k, α is a standardization factor (*e.g.*, $\alpha = 255$), n_k is a unit vector symmetrical to the front $\delta\Omega$ in the point *k* and $\perp$ is the symmetrical operator. The priority $P(k)$ is evaluated for each border patch, with distinct patches for every pixel on the restriction of the target region($\delta\Omega$). During the initialization $C(k) = 0 \forall, k \in \Omega$ and $C(k) = 1 \forall, k \in (1 - \Omega)$.

The confidence term gives reliable information regarding exemplars of a pixel, for example, $C(k)$ gives reliable information regarding the surrounding pixels *k*. The aim is to fill those patches whose a segment of the pixels are so far filled, that is, pixels in the target contour and patches that incorporate corners and thin tendrils of the objective area will, in general, be filled first. As the filling procedure continues, the pixels in the external layer of the objective area have the highest confidence values and therefore it is filled first. The pixels in the center of the target region($\delta\Omega$) will have lesser confidence esteem, so it is filled later. Then for each pixel in the degraded region, we considered neighboring pixels [22] in eight directions and the degraded pixel was filled with the highest priority valued pixel. The same procedure goes inward until all the pixels in the degraded portions are to be filled. When
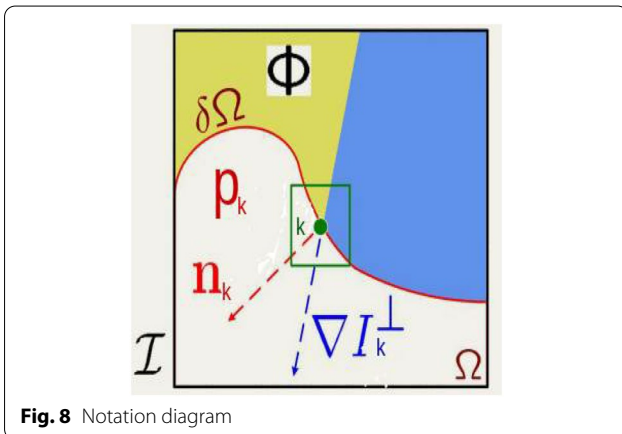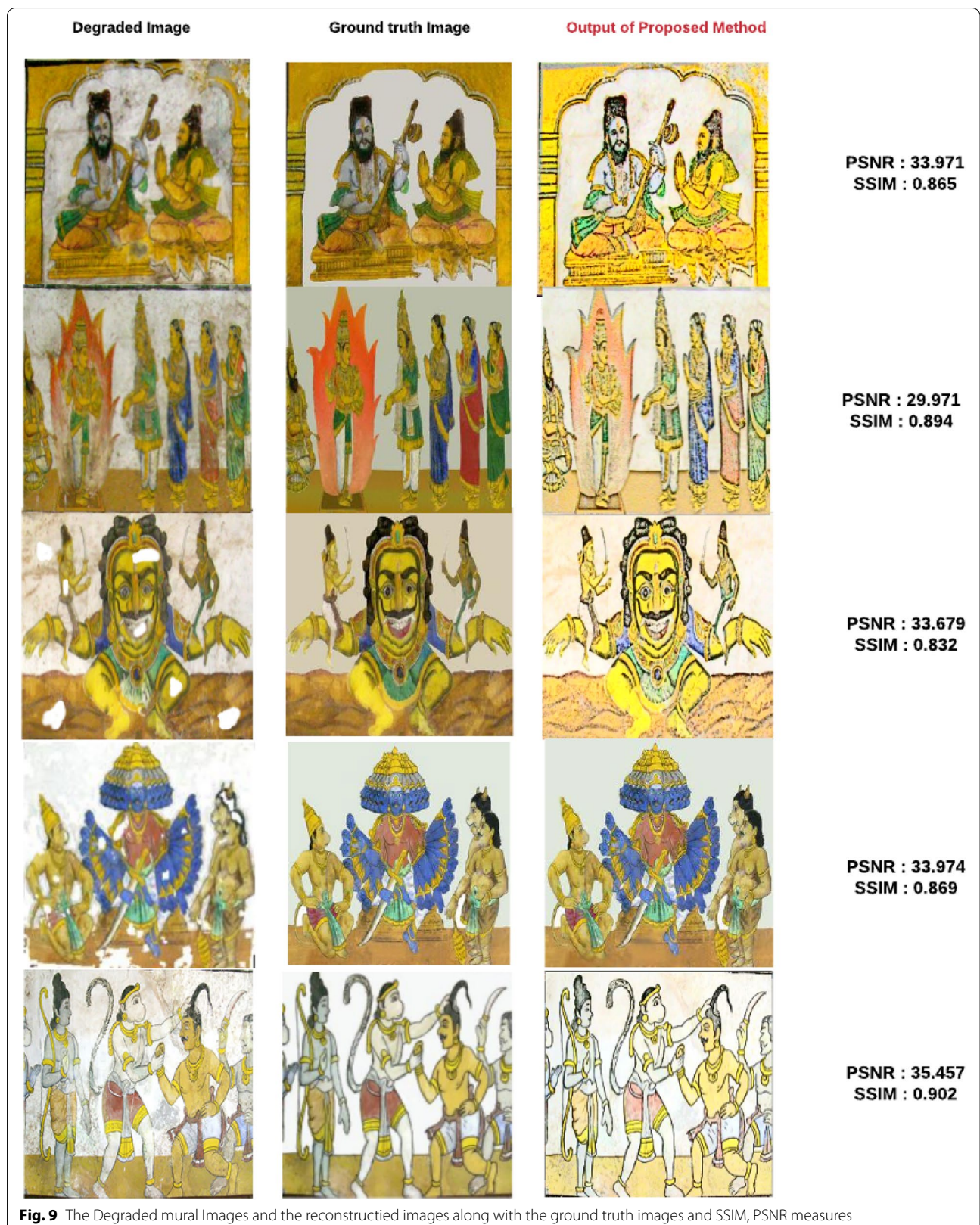
Fig. 9 The Degraded mural Images and the reconstructied images along with the ground truth images and SSIM, PSNR measures
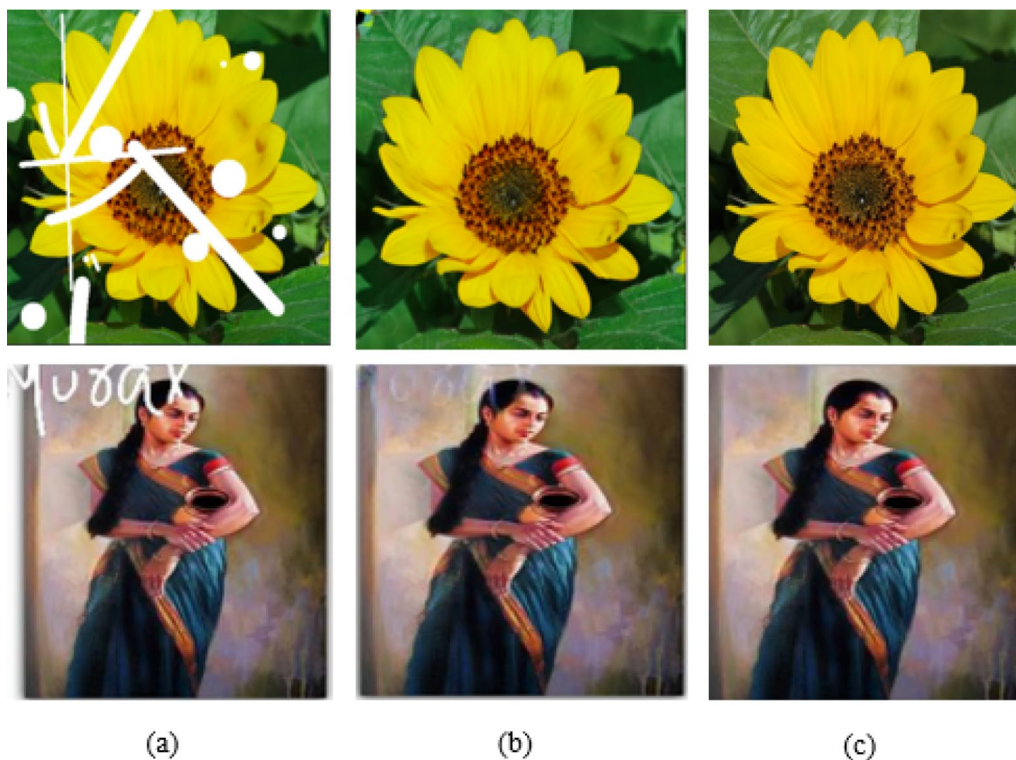
Fig. 10 a The Manually Degraded mural Images and **b** represents the reconstructed image using proposed technique and **c** the original mural image

all priorities on the fill front($\delta\Omega$) have been processed, the patch P'_k with the most elevated is found. We at that point fill it with information extricated from the source region Φ. We proliferate picture texture by direct examination of the source area. That is, we search in the source area for the patch which is most similar to P_k. For example, the isophote of the patch P_k is identified by calculating the minimal distanced patches, it is given by Eq. 10.

$$P_{q'} = arg \min_{P_q \in \Phi} d(P_{k'}, P_q) \tag{10}$$

The minimal distance between two patches is the sum of the squared differences(SSD) of the already filled pixels in the two patches. After identifying the perfect exemplar of the source region($P_{q'}$), the intensity of each pixel k in the target patch(P_k) to be filled with k' where $k' \in P_{k' \cap \Omega}$. After the filling process, the confidence values are updated based on Eq. 11.

$$C(k) = C(k') \forall k \in P'_k \cap \Omega \tag{11}$$

Likewise, all the pixels in the target regions are replaced with their corresponding exemplar. This method provides a better structure and texture reconstruction. This method reconstructs the degraded portions based on purely the priority of surrounding pixels, bringing about a natural combination measure that eliminates the

danger of broken-structure artifacts. Also, the gradient-based guidance will spread solid edges, and blocky and misalignment antiques are reduced. This algorithm does not utilize any explicit or implicit segmentation at any stage and provides a better result compared to other exemplar-based in-painting algorithms.

Figure 9(a) depicts some of the degraded temple murals & Fig. 9(b) represents its corresponding reconstructed image result. Figure 10 represents the test outcome of proposed technique on manually degraded images.

Experimental results
Performance evaluation metrics
The performance of the proposed system is examined using matrices like the Mean Squared Error, Peak Signal-to-Noise Ratio, and Structural Similarity Index.

Mean squared error
Mean Squared Error/Mean Squared Deviation is the normal of the squared error that is used as the loss function for least squares regression. MSE is the entirety of the square of the difference between the envisioned and real objective factors, partitioned by the number of data points. MSE is determined from the formula in Eq. 12. The MSE metric assesses how well data fits a regression trend line using individual error terms. The lesser value

of MSE indicates a better fit and smaller magnitudes of error. The Root Mean Squared Error(RMSE) is calculated by applying the square root to the MSE value as in Eq. 13.

$$MSE = \frac{1}{m*n} \sum_{k=1}^{m} \sum_{l=1}^{n} [I'(k,l) - I(k,l)]^2 \tag{12}$$

$$RMSE = \sqrt{MSE} \tag{13}$$

Peak signal-to-noise ratio

Peak Signal to Noise Ratio is the proportion between the most elevated possible intensity of a signal and the intensity of corrupting noise that influences the devotion of its depiction. The higher assessment of PSNR meant better restoration quality. Since many signals have an extremely wide powerful range, PSNR is normally communicated regarding the logarithmic decibel scale. It is determined from the formula given in Eq. 14.

$$PSNR = 20 \log_{10} \frac{255}{RMSE} \tag{14}$$

Structural similarity index

The Structural Similarity Index (SSIM) is a perceptual metric that estimates the picture quality corruption produced during data compression or transmission. It is a full reference metric that needs two pictures from a similar picture catch - a reference picture and a prepared one. The measurement predominantly considers the likeness between the images. The SSIM index is determined on various windows of an image. The measure between two windows i and j of fundamental size M × M is given in Eq. 15.

$$SSIM(i,j) = \frac{(2\mu_i\mu_j + c1)(2\sigma_{ij} + c2)}{(\mu_i{}^2 + \mu_j{}^2 + c1)(\sigma_i{}^2\sigma_j{}^2 + c2)} \tag{15}$$

Algorithm 5: Reconstruction of Degraded Mural Images

Input	: Structurally restored Mural Images $Restored_i, i \in \{1, 2, 3, ..m\}$, Image Mask $masked_i, i \in \{1, 2, 3, ..m\}$
Output	: Reconstructed Images ,$Recon_j, j \in \{1, 2, ..., m\}$
Methodology:	Reconstruction of Images using Exemplar Based Region Filling

1 Initialize Data, Confidence, $Priority \leftarrow None$
2 **for** *int i* **in** 1 **to** m **do**
3     $img \leftarrow Restored_i$
4     $mask \leftarrow masked_i\ patch\_size \leftarrow 9$
5     $functionInpaint(img, mask, patch\_size)$
6     $unmask \leftarrow MaskRegionRemoval()$
7   **if** $im.size() == unmask.size()$ **then**
8 | Reconstruction is not required
9   **end**
10   **else**
11 | return Height, Width Size Attributes of img
12   **end**
13   $InverseMask \leftarrow 1 - mask$
14   $RGBinverse \leftarrow toRGB(InverseMask)$
15   $EImage \leftarrow img * RGBinverse$
16   $ConfidenceandDataValueInitialization()$
17   **if** $missing\_region$ **then**
18 | $Confidence \leftarrow 0, mask \leftarrow 1$
19   **end**
20   **else**
21 | $Confidence \leftarrow 0, mask \leftarrow 0, Data \leftarrow 0$
22   **end**
23   $FindingFrontRegion()$
24   $Front \leftarrow ApplyLaplacianfilter$
25   $PriorityUpdation()$
26   $Priority \leftarrow Confidence * Data$
27   $Confidence \leftarrow OldConfidence$
28   **if** *pixel px* **in** *Front_Region* **then**
29   **for** *every pixel px* **do**
30         $patch \leftarrow GetPatchInformation$
31         $NewConfidence \leftarrow AreaofPatch$
32         $Normal \leftarrow Normalmatrix$
33         $Gradient \leftarrow HighestGradientValue(Gradient\_Matrix)$
34         $NormalGradient \leftarrow Normal * Gradient$
35         $Data \leftarrow \sqrt{NormalGradient}$
36         $Target \leftarrow highestprioritypixel(img)$
37         $Patch \leftarrow Findsourcepatch(Target)$
38         $Recon\_img \leftarrow$ **Update** $image(Target, Patch)$
39   **end**
40   **end**
41   **Return** $Recon\_Img$
42 **end**

Table 1 Performance comparision of proposed method with other existing techniques

Authors	Methodology	SSIM	PSNR
Pappas et al. [12]	linear approximation and white point transformation	26.039	0.820
Manikanta et al. [11]	K-Means Clustering and Pattern Matching	26.681	0.835
Papyan et al. [14]	Multi scale patch matching technique	30.329	0.867
Chanda et al. [13]	Patch matching Technique	29.215	0.881
Araujo et al. [10]	Optimal Filtering	27.345	0.762
Ruzic et al. [4]	Markov Random Field Algorithm	30.345	0.870
Zhang et al. [5]	Angle-aware Patch Matching Technique	32.345	0.893
Proposed method	Extended Exemplar based Region Filling Algorithm	33.495	0.932

Table 2 Time complexity comparison of the proposed system with other existing techniques

Figure	Percentage of degradation (%)	Time measured in seconds		
		Patch-matching technique [13]	Standard exemplar based technique [6]	Proposed method
5th image in Fig. 8	12.60	18.98	12	8.6
1st image in Fig. 8	18.79	38	33.57	30.14
3rd image in Fig. 8	28	78	76	73.98
2nd image in Fig. 8	22.76	69.56	64	61.58

Results and discussion

Figure 9 shows that the proposed system provides efficient and effective reconstruction results in terms of structure, texture, and color. Table 1 compares the proposed system with existing techniques [4, 5, 10]. The proposed system shows better PSNR and SSIM values than other state-of-the- art methods. Efficient preprocessing and the digital identification of degraded parts, have improved the efficiency of the system.

Most of the existing reconstruction algorithms use front-propagation algorithms for reconstruction. But in the proposed system confidence values are propagated along with the front region, not color esteems themselves, which are examined from the source region. So, the proposed methods maintain sharp edges, and fine information regarding the mural images is preserved; it also reduces diffusion-related blur. The existing reconstruction algorithms use manual identification of degraded parts, but the proposed system uses an automatic identification of degraded parts without any human interference. In the case of structure and texture propagation, existing algorithms introduce blur in the reconstructed region due to the use of diffusion-based propagation. However, the proposed system accomplishes the simultaneous proliferation of structure and surface inside the chose target region effectively and it does not use diffusion-based propagation, so this method avoids blurring of propagated structures, thus making

the algorithm more computationally efficient. This algorithm succeeded in filling the degraded regions seen in ancient temple murals without any explicit and implicit segmentation. Also, it performs well at the limit between two unique surfaces. This method effectively reconstructs both structure and texture. This system identifies the degraded portions automatically without any human interference and it automatically generates the masked images for each degraded one. These properties of the proposed system give better reconstruction results in terms of structure, texture, and color. These properties of the proposed system give a better reconstruction of degraded ancient temple murals. Being hard to find the computational complexity of the proposed algorithm hypothetically, we examine its complexity in the worst and the best cases. Let P_s be the total number of pixels of the source region Φ, and let $n * n$ be the size of the patch. Then the complexity of the algorithm is calculated as $\mathcal{O}(Q * P_s)$ where Q takes value of n^2 in worst case and 1 in best case. The time complexity of the proposed technique is less compared to other state-of-art methods and is depicted in Table 2.

Conclusion

This paper proposes an automated system for the identification and reconstruction of degraded parts seen in the ancient temple murals. This model overcomes the traditional and specific challenges posed by existing

reconstruction criteria. This is achieved by a combination of algorithms such as Dynamic mask creation and reconstruction by extended exemplar-based region filling algorithm. Our algorithm employs an averaging pixel-based analysis to identify the degraded parts and an exemplar-based region-filling technique to reconstruct and determine the order in which the target region will be filled. Textural reconstruction is done by analyzing the confidence and priority values of pixels in the degraded regions, and the structure restored by generating sketches and analyzing edges. Every pixel in an image maintains confidence and priority esteem, which along with picture isophotes, impacts their fill priority. The proposed method propagates both linear structure and texture in the objective region, thereby preserving the structure, texture, fine information on the image, and its sharp edges. It outperforms earlier reconstruction algorithms in terms of the quality of the reconstructed images and computational productivity. Also, this method does not use any segmentation technique and provides a better area filling to dodge over-shooting artifacts. The patch-based reconstruction criteria achieve high efficiency by an accurate synthesis of texture and precise propagation of linear structures. The main disadvantage of this system is that it is a time-consuming process. It does not give efficient results in the reconstruction of areas for which comparable patches do not exist. In the future, we can extend the proposed system by using emerging artificial intelligence such as deep learning technologies for overcoming the disadvantage of this system and also for color enhancement and super-resolution of murals. Furthermore, we are investigating more ground truth data to enhance the performance of the system.

References
1. Colta IP. The conservation of a painting. case study: Cornel minişan, "landscape from caransebeş" (oil on canvas, 20th century). Procedia Chem. 8:72–77, 2013. Youth in the Conservation of CUltural Heritage, YOCOCU 2012.
2. Rakhimol V, Maheswari PU. A survey on restoration of paintings. In: 2020 International Conference on Communication and Signal Processing (ICCSP). 2020; pp. 102–108.
3. Prabhu P. Digital image processing techniques-a survey. Golden Res Thoughts. 2016;5:05.
4. Ruzic T, Pizurica A. Context-aware patch-based image inpainting using Markov Random Field Modeling. In IEEE Trans Image Process. 2015;24(1):444–56.
5. Ji H, Zhang N, Wang G, Liu L. Exemplar-based image inpainting using angle-aware patch matching. EURASIP J Image Video Process, 2019.
6. Criminisi A, Perez P, Toyama K. Region Filling and Object Removal by Exemplar-Based Image Inpainting. In: IEEE Trans Image Process. 2004;13(9).
7. Giakoumis I, Nikolaidis N, Pitas I. Digital image processing techniques for the detection and removal of cracks in digitized paintings. IEEE Trans Image Process. 2006;15(1):178–88.
8. Karianakis N, Maragos P. An integrated system for digital restoration of prehistoric theran wall paintings. 2013 18th International Conference on Digital Signal Processing (DSP). 2013; pp. 1–6.
9. Vaghela KT, Patel NM. Automatic crack detection and inpainting. Int J Comput Sci Eng (IJCSE). 2014;3(6):271–275.
10. Araujo ADA, Hadad RM. Enhancement of sketch contours on paintings infrared photographs by optimal filtering. In: Proceedings II Workshop on Cybernetic Vision. 1996; pp. 231–236.
11. Kumar KMP, Kumar M, Bhargav BVS, Ghorai M. Digital restoration of deteriorated mural images. In: 2014 Fifth International Conference on Signal and Image Processing, 2014; pp. 36–41.
12. Pappas M, Pitas I. Digital color restoration of old paintings.IEEE Trans Image Process. 2000;9(2):291–4.
13. Chanda B, Ratra D, Mounica BLS. Virtual restoration of old mural paintings using patch matching technique. Third International Conference on Emerging Applications of Information Technology, 2012; pp. 299–302.
14. Papyan V, Elad M. Multi-scale patch-based image restoration. IEEE Trans Image Process. 2016;25(1):249–61.
15. Pei SC, Zeng, Chong Y, Chang CH. Virtual restoration of ancient chinese paintings using color contrast enhancement and lacuna texture synthesis. IEEE Trans Image Process Publication IEEE Signal Process Soc. 2004;13(3):416–29.
16. Cai N, Su Z, Lin Z, Wang H, Yang Z, Ling BWK. Blind inpainting using the fully convolutional neural network. In Visual Comput. 2017;33(2):249–61.
17. Sasaki K, Iizuka S, Simo-Serra E, Ishikawa H. Joint gap detection and inpainting of line drawings. In: IEEE Conference on Computer Vision and Pattern Recognition, 2017; pp. 5725–5733.
18. Nakamura T, Zhu A, Yanai K, Uchida S. Scene text eraser. In: 14th IAPR InternationalConference on Document Analysis and Recognition (ICDAR), vol, 1. IEEE, 2017; pp. 832-837.
19. Alilou VK, Yaghmaee F. Application of GRNN neural network in non-texture image inpainting and restoration. Pattern Recogn Lett. 2015;62:24–31.
20. Jain Goswami. A comparative study of various image restoration techniques with different types of blur. Int J Res Comput Appl Robotics. 2015;3(11):54–6.

21. Patil PD, Kumbhar AD. Bilateral filter for image denoising. IEEE ICGCIoT. 2016;299–302.

22. Zeng Y, Gong Y. Nearest neighbor based digital restoration of damaged ancient chinese paintings. In: 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), 2018; pp. 1–5.

Publisher's Note

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