

## ADVANCE COLLABORATIVE FILTERING FOR IMAGE INPAINTING

Gaurav Rasiya<sup>1</sup>, Rank Mayur<sup>2</sup>, Dusyantsinh Rathod<sup>3</sup>

<sup>1,3</sup>Department of Computer Engineering, SSES GI, Kadi, GTU, Gujarat, India.

<sup>2</sup>Department of Computer Engineering, INITLogic Software Solution, Gujarat, India.

**Abstract**— *Image inpainting is technique to recovering the missing or corrupted parts of an image so that the reconstructed image looks natural. Image Inpainting methods can be classified broadly into: Texture synthesis and Structure synthesis. We have chosen a combination of the above two methods combining the advantages of these algorithms because: Capable of inpainting large regions. Also some inpainting method cannot recover good result of the image. We have tried to provide a solution to this through our approach. Image Inpainting The function of this module would be to reconstruct the image based on the best exemplar approach. Non-local exemplar-based image inpainting techniques fill the target portion of the image using the best neighboring area. The exemplar based technique is work with combine both structure and texture synthesis. The method is based on patch filling process. So first find the best priority patch, and then highest priority patch will fill first. Now best match patch find from images to fill that patch. We present a bi-polar exemplar based method for filling in the missing information in an image. Which is combine like and dislike set. The method it fills the damaged region in an image, with surrounding color and texture. The algorithm is based on patch base filling process. Output result is also depends on mask that is defined by user. Using above experiment we conclude that output image quality is depend on patch size. From result we observe that if small scratch in image then small patch size gives batter results.*

**Keywords**—*Texture, Structure, patch, image inpaint technique.*

### I. INTRODUCTION

Now a day Inpainting is art of reconstructing the missing portions of images in order to make it more legible and to restore its unity. The aim is to create a approach that can remove selected portions from the image and fill the hole left behind in a visually plausible way using the background information. Today the digital world, inpainting refers to the application of algorithms to replace missing or corrupted parts of the image information and data. Inpainting is the process of reconstruct missing parts of images. This technique can be used to replace the missing blocks. Image Inpainting methods can be classified broadly into: Texture synthesis algorithm and Structure synthesis algorithm. Here chosen a combination of the two methods combining the advantages of these algorithms because: Capable of inpainting large regions. Large images may take a lot of time to be inpainted. We have tried to provide a solution to this

through our method. Image inpainting would be to reconstruct the image based on the best exemplar approach.

Applications of Image Inpainting Repairing Photographs: With age, photographs often get damaged or scratched. We can revert deterioration using inpainting. We can remove unwanted objects, text, etc from the image using image inpainting. In this method we will starts with original image and mask image as a input. Now mask image define the object or scratch to be removed from the original image. The user can select object to be removed from image because it depends on the appropriate choice of user.so give the reconstructed image as a output. Survey of the paper for first, Simultaneous Structure and Texture Image Inpainting [2] Digital image inpainting is currently hot issues of the computer graphics. The applications of image inpainting are such as increase image resolution, image restoration and object removal. This paper algorithm has divided three main building blocks: Image decomposition, image (structure synthesis) inpainting, and texture synthesis. Second, Region Filling and Object Removal by Exemplar-Based Image Inpainting [3] a new algorithm is proposed for removing large objects from digital images. The challenge is to fill in the hole that is left behind in a visually possible way. Once, this problem has been addressed by two classes of algorithms: One is texture synthesis algorithms which generates large image regions from sample textures and second is inpainting techniques for filling in small image loopholes. Third, A Non local-means Approach To Exemplar-based Inpainting [4]. There are so many image inpainting techniques available exemplar-based inpainting one of them. This paper presents a new approach for the use of nonlocal-means to solve the problem of image inpainting. In traditional inpainting techniques, only local information around the target regions is used to fill in the missing information, which is insufficient in many cases. Fourth, Exemplar-Based Image Inpainting with collaborative Filtering [5]. This paper proposes a collaborative filtering approach is used to enhance the exemplar-based propagation to obtain ideal inpainting results. The approach works on pixel level information, while many exemplar-based propagation algorithms focus on patch level information. Object removal and stain image recovering are carried out to evaluate the proposed approach. This is the plural non-local exemplar based propagation. Inspired by collaborative filtering, an effective unknown pixel value prediction method is proposed in exemplar-based propagation for image inpainting. The method utilizes pixels information between exemplar patches and within patch to provide better results

in pixel prediction. Fifth, Slope One Predictors for Online Rating-Based Collaborative Filtering [6] Rating-based collaborative filtering is the process of predicting how a user would rate a given item from other user ratings. We propose three related slope one schemes with predictors of the form  $f(x) = x + b$ , which compute the average difference between the ratings of one item and another for users who rated both. The basic SLOPE ONE scheme is recommended as a new reference scheme for advance collaborative filtering.

## II. EXEMPLAR BASED COLLEBORATIVE FILTERING METHOD

In the proposed algorithm, we use nonlocal image information from multiple samples within the image. The contribution of each sample to the reconstruction of a target pixel is determined using a weighted similarity function and aggregated to form the missing information. Also weight served to favor frequently occurring pixel value over infrequent pixel value; we will now consider favoring another kind of especially relevant value pattern. We accomplish this by divided the prediction into two set. Using the weighted slope one algorithm, we derive one set from users aT(above threshold) and another set that users bT(below threshold). Given a pixel value scale, say from 0 to 10, it might seem reasonable to use the middle of the scale, 5, as the threshold and to say that items rated above 5 are above threshold and those rated below 5 are not.

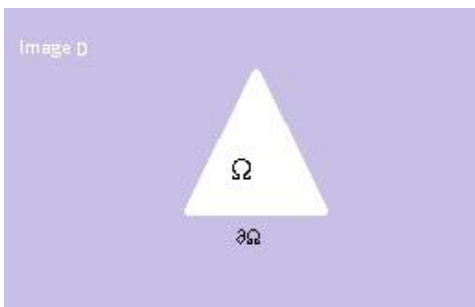


Fig.1: Principle of inpainting

### A. Exemplar Based Slope One Method [5]

**Region to be inpainted ( $\Omega$ ):** The part of the image from where the object is to be removed is known as target region or inpainting domain  $\Omega$ .

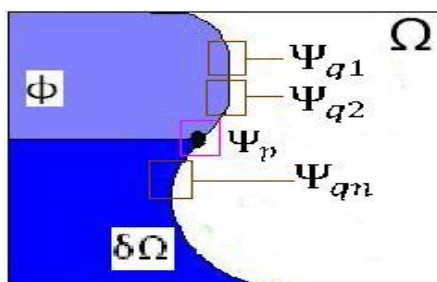


Fig. 2: Exemplar based inpainting

**Source Region ( $I - \Omega$ ):**  $\phi$  the sources region is entire image minus the target region

**Boundary of the target region:**  $\delta\Omega$ : inpainting process start from their target region. So it each iteration we have to find boundary of the target region. Now all the parameter specifying in image, the region- filling proceeds automatically and the algorithm iterates following three steps until all pixels have been filled.

### Step1: Calculating Patch Prioritizing.

Our method is use a strategy of best-first filling that depends on the priority values that are assigned to each patch on the fill front. The calculating of priority is biased toward those patches which are in two parts first, is the continuation of strong edges and second, is surrounded by high-confidence pixels. Given a patch  $\Psi_p$  centered at the point  $p$  for some  $p \in \delta\Omega$  (see Fig), we define its priority as the product of two terms.

$$P(p) = C(p) * D(p) \quad (1.0)$$

Now  $c(p)$  is the confidence term,  $D(p)$  is the data term, they are defined as below:

$$c(p) = \sum_{q \in \Psi_p \cap \phi} C(q) / |\Psi_p|, \quad D(p) = |\nabla I_p^\perp \cdot n_p| / \alpha$$

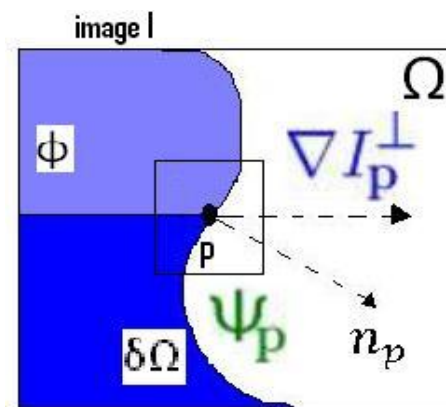


Fig. 3: Term use in Inpainting

### Step 2: Searching Exemplar Patch.

The priority is finding for all patches on boundary then  $\Psi_p$  is highest priority for filling this patch first. Then find  $\Psi_q$  search patch which is similar to  $\Psi_p$ . the exemplar patches are selected from the sum of squared difference (SSD) measure between  $\Psi_p$  and a patch  $\Psi_q$  in  $\Omega$  as

#### 1. Picking Exemplars

The exemplar patches are selected from the sum of squared difference (SSD) measure between  $\Psi_p$  and a patch  $\Psi_q$  in  $\Omega$  as

$$SSD(\Psi_p, \Psi_q) = \sum_{i=1}^M \mu_i (\Psi_p(i) - \Psi_q(i))^2 \quad (1.1)$$

Where,  $M$  is the size of the patch,  $\mu_i$  is pixel mask function with value (0-unknown region, 1-known region) and  $\Psi_p(i)$  and  $\Psi_q(i)$  are the pixel value in respective patches. An patch  $\Psi_q$  is a patch with the lower SSD value. Which is declare as

$$\Psi_q = \min_{\Psi_q \in \Phi} SSD(\Psi_p, \Psi_q) \quad (1.2)$$

This equation 2.1 gives similar to  $\Psi_p$  the minimum SSD value. Here takes color value of two pixels for difference SSD method

2. Synthesis and propagation

Assuming n best exemplar patches  $\Psi_{qi}$  (i=1,2,...,n) are picked out, The weight of n best non-local exemplar patches as

$$w(\Psi_{qi}) = e^{-\frac{SSD(\Psi_p, \Psi_{qi})}{h}} \quad (1.3)$$

A normalized linear combination coefficient is defined as  $\alpha_i$

$$\alpha_i = \frac{w(\Psi_{qi})}{\sum_{j=1}^n w(\Psi_{qj})} \quad (1.4)$$

At last,  $\Psi_q$  is expressed by synthesizing the n best non-local exemplar patches  $\Psi_{qi}$  as

$$\Psi_q(x) = \sum_{i=1}^n \alpha_i \cdot \Psi_{qi}(x) \quad (1.5)$$

Where, x is pixel position in the patch. After finding  $\Psi_q$  fill target region of patch  $\Psi_p$  with synthesized patch  $\Psi_q$ . This approach propagates missing pixels in  $\Psi_p$  with counterpart pixels in the synthesized exemplar patch  $\Psi_q = \sum \alpha_i \Psi_{qi}$ . However, propagated missing pixels in  $\Psi_p$  may not be well integrated with known pixels in  $\Psi_p$ .

Step 3: Fill Target Patch.

Once, the best match patch  $\Psi_q$  is finding fill the pixels in  $\Psi_p$  using this patch. In patch  $\Psi_p$  update only those pixels which are of target region, pixels of the source region are remain as it is. At last completion of these three steps then update boundary with updated target region and again continue these three steps until all the pixels in the target region not fill. The mean deviation value between known pixel j and missing pixel k in  $\Psi_p$  is

$$dev_{j,k} = \sum_{i=1}^n \alpha_i \cdot (p_i^k - p_i^j) \quad (1.6)$$

Where,  $\alpha_i$  is defined by Eq.(3.1).and The pixel value of missing pixel k is obtained by

$$p_0^k = \frac{\sum_{j=1}^m \mu_j \cdot (p_0^j + dev_{j,k})}{\sum_{j=1}^m \mu_j} \quad (1.7)$$

Update  $\mu_k$  to 1 and then fill another unknown pixel in  $\Psi_p$  by the same method until there is no unknown pixel in  $\Psi_p$ .

B. Bi-polar Collaborative Filtering Method

In our approach we used Bi-polar collaborative Filtering Method to fill target region

$p_0^1$	$p_0^2$	...	$p_0^m$
$p_0^{m+1}$	$p_0^{m+2}$	...	$p_0^{2m}$
...	...	...	...
$p_0^{m^2-m+1}$	$p_0^{m^2-m+2}$	...	$p_0^{m^2}$

Table. 1: Pixel Notation in a Serialized Patch  $\Psi_p$

Assuming the k-th pixel in  $\Psi_p$  is unknown(marked by \* in Table2), pixel arrays of  $\Psi_p$  and  $\Psi_{qi}$  (i=1,2,3....n) are arranged into a matrix as Table2.

Pixe l	1	2	...	k	...	M	Threshol d Value
$\Psi_p$	$p_0^1$	$p_0^2$	...	$p_0^k *$	...	$p_0^M$	-
$\Psi_{q1}$	$p_1^1$	$p_1^2$	...	$p_1^k$	...	$p_1^M$	$\bar{\Psi}_{q1}$
$\Psi_{q2}$	$p_2^1$	$p_2^2$	...	$p_2^k$	...	$p_2^M$	$\bar{\Psi}_{q2}$
...	...	...	...	...	...	...	...
$\Psi_{qn}$	$p_n^1$	$p_n^2$	...	$p_n^k$	...	$p_n^M$	$\bar{\Psi}_{qn}$

Table. 2: Pixel Arrays of  $\Psi_p$  and  $\Psi_{qi}$

Pick up the same no of row of all patch  $\Psi_p$  and  $\Psi_{qi}$  ( i = 1,2,3,4...n ).

Fing the average (Thresold) of row of sample patches (  $\bar{\Psi}_{q1}, \bar{\Psi}_{q2}, \bar{\Psi}_{q3} \dots \bar{\Psi}_{qn}$  ).

Calculate all row in two set (  $\Psi_{q1}^{aT}, \Psi_{q1}^{bT}$  ), (  $\Psi_{q2}^{aT}, \Psi_{q2}^{bT}$  ), ... (  $\Psi_{qn}^{aT}, \Psi_{qn}^{bT}$  ).  $\Psi_{q1}^{aT}$  consider all pixel above threshold( $\bar{\Psi}_{q1}$ ) and  $\Psi_{q1}^{bT}$  consider all pixel below threshold ( $\bar{\Psi}_{q1}$ ).

$P_0^j$  is the value of j-th pixel in  $\Psi_p$  and  $P_i^j$  ( i != 0) is the value of j-th pixel in  $\Psi_{qi}$  the mean deviation value between know pixel j and missing pixel k in  $\Psi_p$  for both the set is

$$dev_{j,k}^{aT} = \sum_{i=1, p_i^j \in \Psi_{qi}^{aT}}^n (\alpha_i) (P_i^k - P_i^j), \quad (1.8)$$

$$dev_{j,k}^{bT} = \sum_{i=1, p_i^j \in \Psi_{qi}^{bT}}^n (\alpha_i) (P_i^k - P_i^j) \quad (1.9)$$

Where,  $\alpha_i$  is defined by Eq. (1.4). The pixel value of missing pixel k is obtained by

$$P_0^k = \frac{\sum_{j=1}^m (\mu_j) ((P_0^j + dev_{j,k}^{aT}) \cdot c_j^{aT} + (P_0^j + dev_{j,k}^{bT}) \cdot c_j^{bT})}{\sum_{j=1}^m \mu_j} \quad (1.10)$$







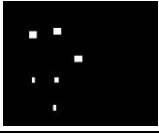

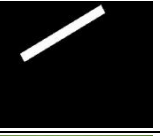






Where the weight  $c_j^{aT} = n(\Psi_{qj}^{aT})$  is number of element in  $\Psi_{qj}^{aT}$  and  $c_j^{bT} = n(\Psi_{qj}^{bT})$  is number of element in  $\Psi_{qj}^{bT}$ , (j=1, 2 ... n).

Update  $\mu_k$  to 1 and then fill another unknown pixel in  $\Psi_p$  by the same method until there is no unknown pixel in  $\Psi_p$

**Algorithm**








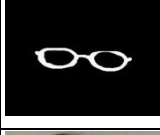




Step of the Algorithm:

1. Define mask ( $\mu$ ) value for the entire pixel. If consider pixel is on target area then set it to 0 otherwise set 1.
2. Calculate boundary ( $\delta\Omega$ ) of the Target area.
3. For all the boundary pixels, create fix size of patch and calculate its priority.
4. Find highest priority patch ( $\Psi_p$ ).
5. Using Sum of Square (SOS) of method to find best n number of sample patch ( $\Psi_{qi}$  ex:  $i=1, 2, 3\dots$ ).
6. Calculate the weight of all sample patches and find its normalized linear combination coefficient ( $a_i$ ).
7. To fill the  $\Psi_p$  patch, fill one by one pixel.
8. suppose z in know pixel in  $\Psi_p$ , Take Z-th pixel's row of all sample patches and find averages of all known pixel of that rows, divided pixels in two set, In first set all above averages pixel consider call that set (aT) and set with below averages pixel call (bT) set.
9. Find mean deviation value between known pixel j and missing pixel Z for both aT and bT set.
10. Both mean deviation values add to know pixel value of  $\Psi_p$  patch and find new value of Pixel Z using average of know pixel value of  $\Psi_p$  patch.
11. Update Z-th pixel of patch  $\Psi_p$  with new value and set mark variable to 1 (like) go to step 8 until all pixel of  $\Psi_p$  patch fill.
12. go to step2 until all the pixel of target area fill.

	Result 1 (dot Stain Recovery)	Result 2 (Scratch Removal)	Result 3 (Scratch Removal)
<b>Original Image</b>	 (414 x 325)	 (391 x 384)	 (269 x 201)
<b>Inpaint Image</b>	 (patch size: 9x9)	 (patch size: 13x13)	 (patch size: 3x3)
<b>Mask Image</b>			
<b>Slope One[5]</b>	 (PSNR=36.0076)	 (PSNR= 39.8572)	 (PSNR= 35.6035)
<b>Our approach</b>	 (PSNR=37.5236)	 (PSNR=39.8663)	 (PSNR=35.8884)

**III. EXPERIMENT RESULT**

We chose six images to compare performance of different relating approaches. In Table1 shows the experimental results. First row is show the original image. Second row is show the inpaint image. Third row is show the mask image. Fourth row is show the different resulted image of Collaborative slope one method and Fifth row is show the different resulted image of our approach. Here, we can calculate the peak signal-noise ratio (PSNR) to measure the quality of inpainting. PSNR is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. In experiment of different image recovering method, to show Table1 got lower PSNR value of slop one method compare to our approach, so our approach performs better than slope one. In experiment of text, dot, scratch and object removing in our approach obtained a higher PSNR value than slope one. While in our approach can still increase PSNR value. The results illustrate that Bi-polar collaborative filtering algorithms can improve performance of inpainting algorithms.

	Result 4 (Object Removal)	Result 5 (Object Removal)	Result 6 (Character Removal)
<b>Original Image</b>	 (168 x 105)	 (198 x 237)	 (200 x 134)
<b>Inpaint Image</b>	 (patch size: 3x3)	 (patch size: 9x9)	 (patch size: 3x3)
<b>Mask Image</b>			
<b>Slope One[5]</b>	 (PSNR=31.4831)	 (PSNR=27.4439)	 (PSNR= 35.5360)

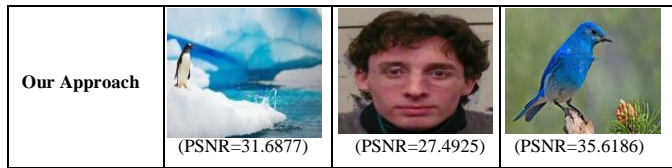


Table. 3: Experiment results.



Graph1. PSNR Comparison

Six different result are created using two method is show in above graph. We can see that our approach PSNR value still increase then slop one method.

#### IV. CONCLUSION

Here we conclude that the Bipolar Advance collaborative filtering method is better the Slope one method. In this dissertation work, we present a bi-polar advance exemplar based method for filling in the missing information in an image. Which is combining above and below threshold set? The method it fills the damaged region in an image, with surrounding colour and texture. The algorithm is based on patch base filling process. Output result is also depends on mask that is defined by user. Using above experiment we conclude that output image quality is depend on patch size. From result we observe that if small scratch in image then small patch size gives batter results. Image inpainting is technique to fill missing region from an image. It removes scratch, dot, line and object from an image in naturally. For filling the part of image, it uses information from the neighbouring area.

#### REFERENCES

- [1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. "Image inpainting." in Proc. SIGGRAPH, 2000, pp. 417–424.
- [2] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher. "Simultaneous structure and texture image inpainting," in Proc. Conf. Comp. Vision Pat-tern Rec., Madison, WI, 2003.
- [3] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. "Region filling and object removal by exemplar-based

image inpainting". IEEE Trans. Image Process, vol. 13, pp.1200–1212, 2004.

- [4] Alexander Wong<sup>1</sup> and Jeff Orchard<sup>2</sup>. "A nonlocal-means approach to exemplar-based inpainting." in Proc. the IEEE Int. Conf. Image Processing, 2008.
- [5] Xinran Wu, Wei Zeng, Zhenzhou Li. Exemplar-Based Image Inpainting with Collaborative Filtering." IEEE Sixth International Conf. on Image and Graphics, 2011.
- [6] Daniel Lemire, Anna Maclachlan. " Slope One Predictors for Online Rating-Based Collaborative Filtering" January 9, 2005.
- [7] T. Chan and J. Shen. "Mathematical models for local non-texture inpaintings." SIAM J. on Appl. Math, vol. 62, pp. 1019–1043, 2002