

Multiple Object Removal Using Exemplar-Based Image Inpainting

Prof. Kawade M.D.

Information Technology,
LS KBJ's SNJB's COE,
Chandwad,Nashik
madhuri3620@gmail.com

Prof. Ufade A.S.

Electronics Engg.
MET's,IOE,Nashik,
India.
amitufade13@gmail.com

Abstract— The art of filling missing data in an image is known as image inpainting. The main purpose of inpainting process is to reconstruct missing regions in a visually plausible way. The propose system suggest a new algorithm for multiple object removal from images using adaptive patch selection, the implemented algorithm select unwanted part manually i.e target region. Further to improve the performance of inpainting, here image is divide into source region and target region and then searches the best matching source patch using surrounding pixel information of target region. The proposed system is useful for multiple object inpainting and gives better result with less Execution time. The performance analysis is done using evaluation parameter like RSME, PSNR. And Execution time.

Keywords- Image Inpainting, Object remove, Target region, Patch selection, Exemplar based inpainting

I. INTRODUCTION

The people are already preserving their visual works carefully. Now, because the popularity of computers, people often keep their arts, pictures, even old films or cartoons in their computers. By the great power of modern computers, we can inpaint digitalized images. Image inpainting is a technique to restore/complete the area of a removed object which is manually selected by the users. Image inpainting fills the missing or damaged region in an image utilizing spatial information of neighbouring region [1, 2]. Digital Inpainting is a term introduced in [3]. It alludes to how to perform inpainting digitally through image processing in some sense. Thereby also automating the process and reducing the interaction required by the user. Ultimately, the only interaction required by the user is the selection of the region of the image to be inpainted. When we capture images there may be some unwanted object that comes in between, hence there is need of software that can efficiently remove these unwanted objects from the image.



(a) Original image



(b) Inpainted image

Figure 1.1- Remove unwanted object from digital photograph.

In above Fig.1.1 suppose we capture Elephants image using digital camera, but we want only big Elephant. For that purpose a software will be require. The Inpainting method that were incorporated in commercial software also (i.e. Photo wipe, Adobe photo shop) [3], but non-expert users are not familiar with the Inpainting concept. Extensive research has been carried out in the field of digital inpainting , such that currently, a large number and variety of algorithms exist. Depending on the approach they take towards image completion, digital inpainting algorithms can be grouped into two main categories. Partial differential equation (PDE) based algorithms, like the ones proposed in [4] fill in missing regions in an image by extending lines of equal luminance values, from the source region into the target region, via diffusion. The main drawback of this type of inpainting algorithms consists of introducing post- inpainting blur artifacts, that become more visible when

larger areas are inpainted. The second category comprises exemplar-based inpainting algorithms [2] [3] [11] [13]. Methods in this category try to overcome the drawback exhibited by PDE based techniques, by reconstructing large missing image regions from sample textures.

Image inpainting can be divided into two basic approaches. Texture oriented-mainly use for remove large image gap & Structure oriented-for small image gap. The main purpose of inpainting process is to reconstruct missing regions in a visually plausible way. The propose system suggest a new algorithm for multiple object removal from images using adaptive patch selection, the implemented algorithm select unwanted part manually i.e. target region. Further to improve the performance of inpainting, here image is divide into source region and target region and then searches the best matching source patch using surrounding pixel information of target region. The proposed system is useful for multiple object inpainting and gives better result with less Execution time.

II. EXEMPLAR BASED INPAINTING

PDE based inpainting algorithms are not sufficient for faithfully reconstructing textured images, nor images with large missing areas. Thus, when inpainting is done with an image restoration purpose in mind, more complex techniques are required, as paintings are composed of both structures (i.e. primal sketches) and textures (i.e. regions with homogeneous patterns). Because of this characteristic of paintings (and natural images, in a more general manner), a technique that is strictly designed for texture synthesis will not perform well, either. Exemplar-based inpainting methods can overcome this drawback, being able to provide reasonably good quality results, even for large gaps, by combining the isophote driven inpainting with texture synthesis [19]. The idea behind exemplar-based image completion refers to the use of a set of image blocks (or exemplars) that can be extracted either from the actual image that needs to be inpainted, or from another image that belongs to a set of representative images. However, it is most common, in the context of inpainting, to fill in the gap with the best matching parts of the same image [10]. The order in which the gap is filled in has a great importance, as it may contribute to the minimization of artifacts.

III. PROPOSED SYSTEM

To inpaint image we used multiple size patches (3X3, 5X5, 7X7). Results show the accuracy of proposed system. Multiple object inpainting method removes the all drawbacks of traditional system. It improves the speed and quality of image inpainting. The proposed system is based on multiple object selection and multiple patches filling module. Following Fig 3.1 show the input and output image of proposed system.



Figure 1.2:Input and output of proposed system

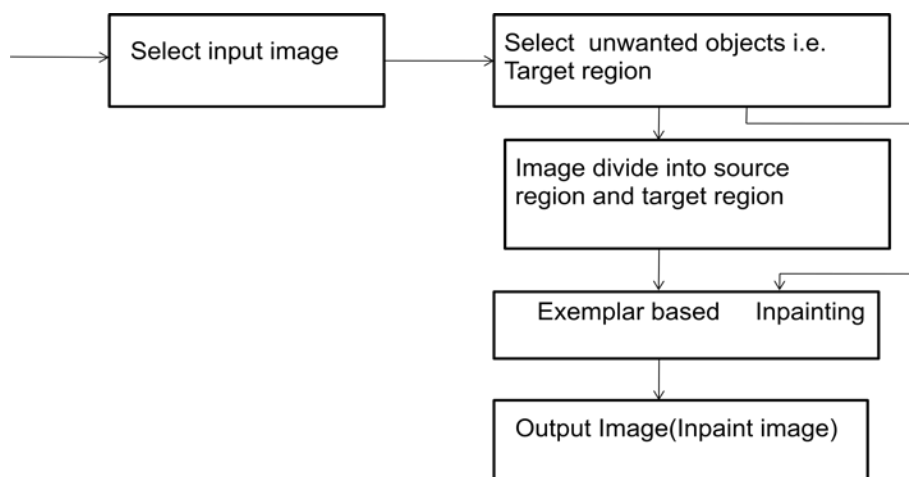


Figure 1.3: Overall working of proposed system.

A. Main Three Steps-

1. Select target object-
User manually select the target region.
2. Identify the Source region and Target region
3. Inpainting
1. *Select target object(T) i.e. Multiple object selection*

In the first multiple object selection module, systems work is depend on selecting multiple object from single input image, to do this action we implement one method to select unwanted part or damage part manually using user interaction

In this system we select multiple objects at a time which allow user to inpaint multiple objects at the same time. For this we used array data structure to store the information of multiple location points and area of the selected patches.

$$S = I - T \quad (1)$$

Where I is a input image to be inpaint, T is a target region and S is a Source region.

2. *Identify the Source region and Target region*

Target region indicate by white(255) color and background using black

3. *Inpainting*

In this step are two modules, first is patch selection and second is patch filling. In the patch selection method we calculate the parameter value using patch selection function.

- *Patch selection*
 - i. Calculate the average value of each patch,

$$C = \frac{1}{n} \sum_{i=1}^n x(r, g, b) i \quad (3)$$

Where c is target color , n is number of colored pixel. r,g,b is color value of ith pixel

- ii. Calculate the distance of each patch with each other

To select patch we first calculate boundary block of the target region. After that we select one block from the target region and calculate the distance for each surrounding block. To calculate this distance we used following formula.

$$\sum_{i=0}^{dn} (d_i - d_{i+1}) \quad (4)$$

Where d is the distance of each block pixel and n is number of blocks

- *Patch Filling*

In the second part we fill the target region with selected patch. To do this we select block with minimum distance value and replace target block with that block. We repeat this process for each block in the target region. We used following formula to fill the patch.

$$\sum_{i=1}^n \sum_{j=1}^m \text{Set(patch)} i, j \quad (5)$$

Where n is width and m is height of image patch is block having minimum distance

IV. PERFORMANCE ANALYSIS

Several experiments have been done to check the effects of proposed method. The platform used for the experiments, is equipped with Intel Core i3 CPU and 3GB memory running on Windows 7 and 2.6GHz clock speed. The experimental results are obtained using .Net C#. RGB color model is used. The patch size is 3X3,5X5,9X9 etc.To verify the effectiveness of the proposed variance approach and the improvement in speed, we performed tests on several images and compared.The algorithms have been tested for a variety of different cases such as PSNR,RMSE, Execution time etc.

4.1 Test images

For evaluating the performance of the inpainting algorithm six different images (varying in size and resolution, as seen in Table 1.1), have been considered. These images, labelled as parachoots,Birds,Parrots,Cartoons,Fish respectively. Fig. 4.1 shows the five deteriorated images, with the inpainting domain marked in white. As it can be observed, the size of the region to be inpainted differs from image to image, varying from narrow gaps to larger gaps. The test images include natural scenes and full color photographs of complex textures



Figure 1.4: Test images

Table 1.1: Resolution, Dimensions and size of images considered for inpainting evaluation

| Image | Resolution(dpi) | Dimensions | Size(KB) |
|-------------|-----------------|------------|----------|
| Para shoots | 96 | 512x512 | 130 |
| Birds | 94 | 512x512 | 69.4 |
| Parrots | 96 | 512x512 | 38.8 |
| Cartoons | 94 | 512x512 | 98.8 |
| Fish | 94 | 512x512 | 105 |

4.2 Comparison with Jino Lee Approach[7]

We have an image with two parashoots. One parashoots in the image is the unwanted parashoot that we now want to remove. The performance of our algorithm is compared with the Jino Lee[7] existing methods in terms of the inpainting quality(i.e.RMSE,PSNR) and time.The value of RMSE, PSNR and execution time store in following Table 1.2 and Table 1.3 for five different images, Proposed system PSNR value is high so it restores the target region more closely to the true pixels also it required less time than the old system.



Figure 1.5: Comparison with Jino Lee approach. (a) Image to be inpainted. (b) Result using our algorithm. (c) Result using our implementation of Jino Lee approach[13]

Table 1.2: Comparison of Old system[7] and Proposed system using Time(in Milliseconds)

| Image Name | Old system | Proposed system |
|-------------|------------|-----------------|
| Para shoots | 933 | 195 |
| Birds | 2159 | 143 |
| Parrots | 1358 | 154 |
| Cartoons | 1849 | 184 |
| Fish | 2750 | 116 |

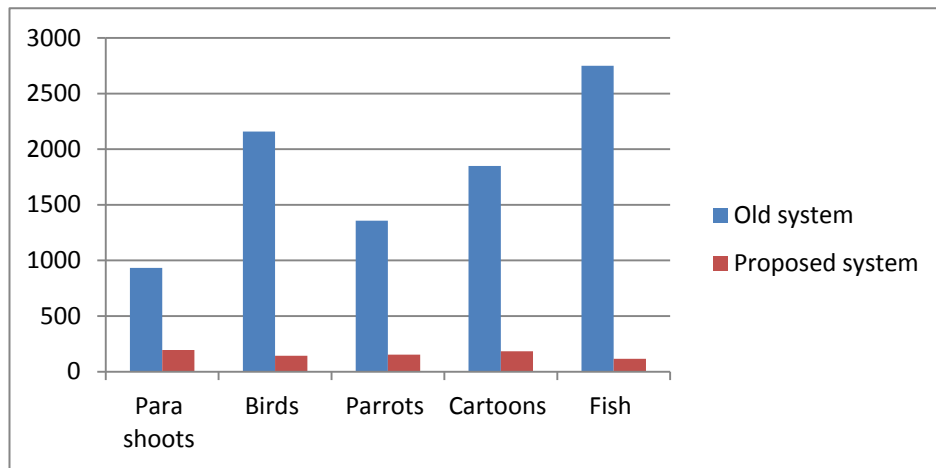


Figure 1.6: Graph show the proposed system required less time

Table 1.2: Comparison of Old system[7] and Proposed system using RMSE and PSNR

| Image Name | Old System | | Proposed System | |
|-------------|------------|---------|-----------------|---------|
| | RMSE | PSNR | RMSE | PSNR |
| Para shoots | 6.252 | 32.2105 | 0.7118 | 51.0834 |
| Birds | 4.2849 | 35.492 | 0.9555 | 48.5264 |
| Parrots | 7.8374 | 30.2474 | 0.8544 | 49.4981 |
| Cartoons | 6.1155 | 32.4022 | 1.0678 | 47.5611 |
| Fish | 6.4247 | 31.9738 | 0.6546 | 51.8111 |

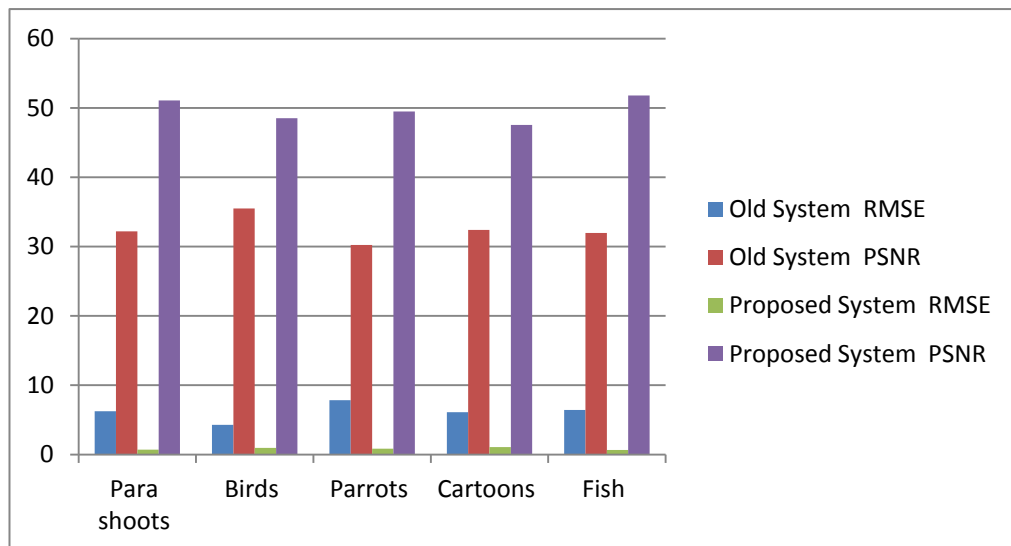


Figure 1.7: Graph show the proposed system have less degradation than old system

V.CONCLUSION

The Work proposes for multiple object remove using exemplar based inpainting method. It improves robustness of inpainting results, which is influenced by patch selection function. We are dealing with the image processing into Patch rather than pixels. The working with pixels increase extra overhead, thus to solve this problem of overhead we use patch processing with user defined patch size. User can select multiple patch size according to his requirements. The enhanced inpainting method includes: Modifying the patch filling process by selecting matching patch using distance based method. In our system PSNR value is high so it restores the target region more closely to the true pixels also it required less time than the old system. Single object removal required less time than multiple object removal. In future try to remove unwanted object from video.

REFERENCES

- [1] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-Stokes, fluid dynamics, and image and video inpainting," in Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 355–362, Kauai, HI, Dec. 2001.
- [2] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," IEEE Trans. Image Processing, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.
- [3] W.-H. Cheng, C.-W. Hsieh, S.-K. Lin, C.-W. Wang, and J.-L. Wu, "Robust algorithm for exemplar-based image inpainting," in Proc. Int. Conf. Computer Graphics, Imaging and Vision 2005, Beijing, China, Jul. 2005, pp. 64–69.
- [4] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," In Proc. ACM Conf. Comp. Graphics (SIGGRAPH), pp. 417–424, New Orleans, LU, Jul 2000.
- [5] A. Efros and T. Leung. "Texture synthesis by non-parametric sampling", In Proc. ICCV, pp. 1033–1038, Kerkyra, Greece, Sep 1999.
- [6] Criminisi, A, Perez, P., & Toyama, K., " Object Removal by Exemplar-Based Inpainting", Appears in the Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition
- [7] Jino Lee, Dong-kyu Lee, Rae-Hong park, "Robust Exemplar-Based Inpainting Algorithm Using Region Segmentation", IEEE Trans. Image Processing, vol. 98, no. 3063, Feb. 2012
- [8] Anupam, Pulkit Goyal, Sapan Diwakar, "Fast and Enhanced Algorithm for Exemplar Based Image Inpainting", 2010 Fourth Pacific-Rim Symposium on Image and Video Technology, Information Technology IIT Allahabad, 2010.