Examplar-based Image Inpainting and Approaches to Improve the Performance

Sheetal Badgujar
K. K .Wagh Polytechnic
Chandori

N. M. Shahane
K.K.Wagh I. E. E. & R.
Nashik

ABSTRACT
Image inpainting is the process of filling in missing regions in an image. The objective of inpainting is to reconstruct the missing regions in a visually plausible way. Several algorithms are available in the literature for the same. In this paper we introduce a two different approaches to improve the performance of exemplar based Image Inpainting Algorithm. Both the algorithms are based on patch propagation by inwardly propagating the image patches from the source region into the interior of the target region patch by patch.

In the first approach of exemplar-based image inpainting a simple patch shifting scheme is used. In traditional exemplar-based inpainting errors often occur when small number of known pixels are used to present a large unknown area. The patch shifting scheme provides more meaningful target patch than traditional exemplar-based approach. In this scheme, the target patch which has known pixels less than predefined threshold would be shifted in the direction that increases the number of known pixels. This means the chance to filling-in each patch more naturally is increasing.

The second approach of exemplar-based image inpainting algorithm uses region segmentation. The method uses segmentation map to improve the performance of robust inpainting, in which a segmentation method is used to utilize spatial information in the source region. With the segmentation map, it adaptively determines patch size, and reduces search region.

General Terms

Image inpainting, image completion, exemplar-based, image segmentation

Keywords
Patch, patch shifting, threshold value, segmentation map, adaptive patch size

1. INTRODUCTION
Image inpainting is the research area in the field of image processing whose goal is to remove some objects or restore the damaged regions in a way that observers cannot notice the flaw. There are many applications of image inpainting such as photo editing, video editing, image compression and image transmission. Generally image inpainting techniques can be categorized into two approaches; Diffusion-based and Exemplar-based approaches. Diffusion-based approach is the fundamental approach in which information diffuses from known region into missing region. The problem is usually modeled by Partial Differential Equation (PDE), so sometimes it is called a PDE-based approach. Diffusion-based approach works well for non-texture image, in which the missing region must be small and thinner than the surrounding object. In the case that the missing region is large or containing texture, this approach gives a blurry result.

Exemplar-based approach is originated from the Exemplar-based texture synthesis in [1]. In that work, the texture is synthesized by copying the best match patch from the known region. However, as there are both structures and textures in natural images, directly applying Exemplar-based texture synthesis to image inpainting problem may not provide satisfactory result. Bertalmio [2] proposed to decompose the image into structural and textural images, then apply Diffusion-based inpainting to the structural image and texture synthesis to the textural image separately. The result of combining restored structural and textural image is better than restoration by only Diffusion-based inpainting or texture synthesis alone. For Exemplar-based texture synthesis to determine the fill-in order Criminisi et al. [3] introduced patch priority, which is defined by isophote direction and the known region in the target patch, Comparing with Diffusion-based inpainting, Exemplar-based approach gives a better result even in the large missing region case.

The problem of most of Exemplar-based approaches is that, in some cases, satisfied results cannot be achieved because large unknown region is filled by small number of known pixels. The patch shifting technique makes sure that the target patch always contains enough known pixels so it can efficiently compare with the candidate patch in the source region. In this way, more reliable patch is chosen to fill the unknown region.

In robust inpainting algorithm using region segmentation, segmentation map provides local texture similarity and dominant structure region. With boundary information of a segmented image map, the proposed method determines the suitable patch size and selects candidate source regions for reducing unnatural artifact.

2. EXAMPLAR-BASED APPROACH
Generally image inpainting is modeled as the problem of filling-in the missing region T, called target region, of the given image I by the information of the known region, called source region S. ∂T is the boundary between target region T and source region S. Fig.1 describes a notation diagram of an Exemplar-based inpainting algorithm, in which an image is divided into two regions S and T by boundary ∂T.

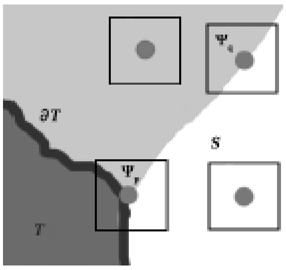

Fig.1 Notation diagram for exemplar-based inpainting

In Exemplar-based image inpainting, in order to fill the target patch Ψ_p , which is centered at pixel p and partially within T, the best match patch $\Psi_{\hat{q}}$, which is centered at $\hat{q}$, is chosen from the source region. Then the intensities of Ψ_p, in the target region are filled by copying from the corresponding pixels of $\Psi_{\hat{q}}$. The order of selecting target patch intensively affects the restored result as the example shown in [3]. For a natural looking result, the edges should be continued which means the patch which contains high structural information should be filled first. With this principle, patch priority P is introduced [3]. It is determined by the magnitude of the isophote direction and the known pixel in the patch. On each iteration, the target patch which has the highest patch priority is filled. Mathematically, patch priority is defined as

$$P(p) = C(p) \cdot D(p) \qquad (1)$$

The confident term C(p) and data term D(p) are defined as follows;

$$C(p) = \frac{\sum_{q \in \Psi p \cap S} C(q)}{|\Psi p|} \qquad (2)$$

$$D(p) = \frac{|\nabla I \frac{\perp}{np} \cdot np|}{255} \qquad (3)$$

where $|\Psi_p|$ is the area of Ψ_p, np is the normal vector of the front ∂T, $I\frac{\perp}{np}$ is the isophote at p and α is the normalizing factor which equals 255 for 8-bit grey-scale image. The confident term C shows the ratio of known pixels which surround the center of the target patch. The data term D shows the strength of the edge at the target patch. The process of Exemplar-based approach can be described as follows. Firstly, the confidence term is initialized by assigning to C(p) = 0 for ∀p ∈ T and C(p) = 1 for ∀p ∈ S.

Then the following process is repeated until the filling front ∂T = ∅.

1. Identify the filling front ∂T.
2. Compute patch priorities of all the patches whose center align on filling front ∂T.
3. Chose the patch Ψ_p which has the maximum patch priority.
4. Find the best match patch $\Psi_{\hat{q}}$ of Ψ_p from the source region S.
5. Copy data from $\Psi_{\hat{q}}$ to Ψ_p for ∀p ∈ $\Psi_p \cap$ T
6. Update C(q) for ∀p ∈ $\Psi_p \cap$ T

Note that, the best match patch $\Psi_{\hat{q}}$ in step 4 is the patch which minimizes the Sum of Squared Differences (SSD) between itself and Ψ_p in known region. SSD is defined as follows;

$$d(\Psi_p,\Psi_q) = \sum_{(i,j)} \left| I\left(p_{(i,\ j)}\right) - I\left(q_{(i,\ j)}\right)\right|^2,$$
$$(\forall\, p_{(i,j)} \in \Psi_p \text{ and } \forall q_{(i,j)} \in \Psi_q). \qquad (4)$$

3. PATCH SHIFTING

As we discussed previously that the best result of Exemplar-based approach may not be achieved because in some cases the target patch has not enough known pixels for a meaningful representation. This situation can occur and ruin the final result, although the number of known pixels is a parameter to consider the patch priority. In Fig.2, it obviously seems that the target patches on the right column would produce the better result than the target patches on the left column. In this paper, we introduce an easy but efficient approach to modify the target patch in the way that it always contains enough known pixels to produce more reliable result. Our idea is to shift the target patch to the known region in the case that there are not enough known pixels in that patch.

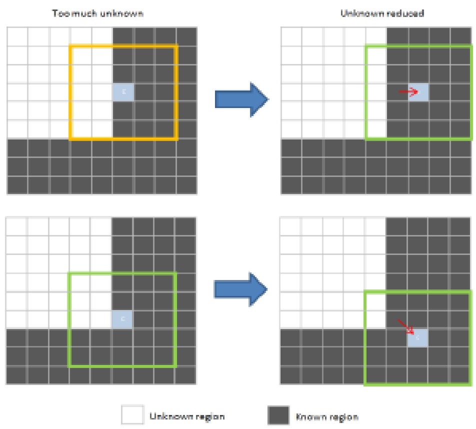

Fig.2 The idea of patch shifting

As shown in the first row of Fig. 2, if 15 known pixels (60% of the patch size) is not enough for the criteria then the target patch is shifted to the right as shown on the right. In this case, we gain 5 more known pixels (20% of the patch size). For more clear understanding, let us consider the second row of Fig.2. If known pixels must be more than 76% of the patch size in each target patch, the target patch should be shifted one pixel right and one pixel down as shown in the bottom right of Fig.2. After shifting, we gain 4 more pixels (16% of patch size).

In practice, we can find the vertical shift S_v and horizontal shift S_h of the patch by

$$S_v(p) = \sum_{n=-1}^{1} \sum_{m=-1}^{1} \psi\,(i+m, j+n) M_V(m+2, n+2)$$

$$S_h(p) = \sum_{n=-1}^{1} \sum_{m=-1}^{1} \psi\,(i+m, j+n) M_h(m+2, n+2) \qquad (5)$$

where

$$M_v = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \qquad M_h = \begin{bmatrix} +1 & 0 & -1 \\ +1 & 0 & -1 \\ +1 & 0 & -1 \end{bmatrix} \qquad (6)$$

ψ is a binary image whose pixel is 0 at known pixel and 1 at unknown pixel, and p = (i, j) is the center of the target patch.
On Exemplar-based inpainting, patch shifting is applied to the target patch with maximum priority whose number of known pixels is less than the predetermined threshold. The target patch repeatedly shifts until the number of known pixel is more than threshold. Then, the best matched patch of the shifted target patch is searched. The next target patch with lower maximum priority is chosen and does the patch shifting again if necessary. These processes are repeated until satisfied target patch is found. Note that, to maintain the advantages of patch priority, we apply patch shifting to only limited number of target patches. The target patch which has too low priority may cause discontinuity in reconstructed edges. If there is no satisfied patch, the shifted patch of the maximum priority patch is chosen.

4. ROBUST EXAMPLAR-BASED INPAINTING

4.1 Region segmentation

A Graph-based region segmentation algorithm [4] is used to perform segmentation. An initial graph G = (V, E) is refined as a segmentation map M that provides significant structural information of I, where V represents initial vertex set and E denotes the corresponding set of edges. A segmentation map M is a set of properly refined regions through iterative merging process. In each merging step, components of vertices C_k and C_{k+1} are merged into one segment if the difference between two components is smaller than internal difference of two components.

First, the segmentation algorithm is used to produce an initial segmentation map. Next, we merge segments in T of the initial segmentation map into one segment and then assign a new label that indicates the target region. The segmentation map in the robust Exemplar-based inpainting method performs two functions: as an indicator of T and as selection criteria of patch size and candidate source regions.

Fig. 3 shows examples of segmentation map M with three input images. Segmentation maps are labeled with gray-scale values [0, 255]. We update labels of an initial segmentation map to classify target region. We set the label of user-defined target region to 255 (white). Also, if there are source regions that have values of 255, we change the label to an unused gray-scale value. Figs. 3(a), 3(c), and 3(e) are three input images. Figs. 3(b), 3(d), and 3(f) are final segmentation results of Figs. 3(a), 3(c), and 3(e), respectively, in which target regions have values of 255. Moreover, source regions are divided and represented as gray scale values according to their local texture similarities

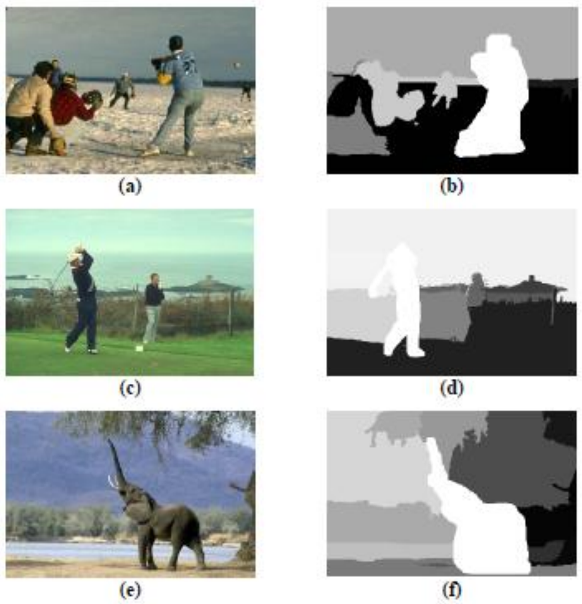

Fig. 3. Segmentation results. (a) Baseball (256×170), (b) segmentation map of (a), (c) Golfer (481×322), (d) segmentation map of (c), (e) Elephant (384×256), (f) segmentation map of (e).

4.2 Robust exemplar inpainting

Considering local texture similarity using a segmentation map, the method fills efficiently the target region with patches in source regions. It can adaptively choose the patch size between 9×9 and 17×17 using segment information in the target patch. An image can be separated into several regions depending on texture similarity while dominant structures are identified as boundaries of adjacent segments. Using segmentation map M, an input image I can be separated into several regions, which is expressed as

$$I = \cup_{i=1}^{N} Ri \qquad (7)$$

where R_i represents i^{th} segment of I and N denotes the total number of segments. A chosen target patch Ψ_p , belongs to at least one segment R_i. This method, simply define selection rules for suitable patch size and candidate search region with segmentation result. First, the patch size is adaptively selected as follows. When the current patch is located on the segment boundaries, a default window size (9×9) is used. On the other hand, when the current patch belongs to a single segment R_i, it increase the size of the patch while $\Psi_p \subset R_i$. The maximum window size of patches can be set to 17×17 to achieve high quality results. Next, to prevent undesirable source patch selection, the method restrict search region using adjacent segments only.

It is assumed that an image is grouped according to texture similarity, thus search area is restricted to adjacent neighboring regions. This method searches corresponding candidate source regions that contain target region. With this approach, the computation time and error propagation can be reduced.

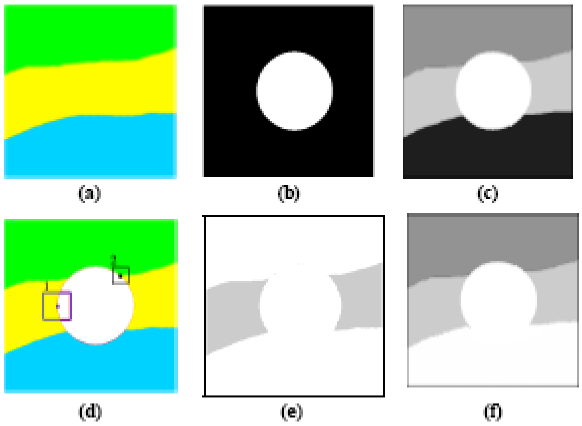

Fig. 4 Adaptive patch size and candidate source regions (a) input image (300 x300)(b) target image (c) segmentation map (d) two target patches that belong to different segments (e) candidate source region for patch 1 (f) candidate source region for patch 2

Fig.4 describes the inpainting method using a synthesized image. Fig. 4(a) is a synthesized input image and Fig.4(b) is a target image. Fig. 4(c) is a segmentation map constructed from an input image and a target region, in which segmentation map contains the target region (white region). In Fig.4(d), the adaptive patch size selection is shown, where target patch 1 belongs to only a part of a single segment. Therefore, the size of target patch 1 is determined by patch size selection rules that are previously mentioned. Because target patch 2 is located on the segment boundaries, the patch size of target patch 2 is fixed to 9×9. There can be the local similarity of an input image through the region segmentation step. This can help to reduce search area for every target patch. Figs. 4(e) and 4(f) show candidate source regions (gray regions) for target patches 1 and 2, respectively. Similar to patch size selection step, the method reduces search area according to the position of the target patch. An image in Fig. 4(c) is divided into four regions according to segmentation map *M* and target region *T*. Two target patches 1 and 2 belong to different segments as shown in Fig. 4(d). The method searches source regions that have the same label in the target patch.

It finds pixel $\hat{P}$ with the maximum priority and thus the most similar source patch, $\Psi\hat{q}$ where $\hat{q}$ is the center point of the patch. It searches the candidate source region to find a patch with the minimum distance from the patch, $\hat{p}$ i.e.

$$\Psi\hat{q} = \arg\ \min_{\Psi_q \in S} d(\Psi\hat{p}, \Psi q) \qquad (8)$$

Distance $d(\Psi\hat{p}, \Psi\hat{q})$ is defined as the sum of squared differences, which is expressed

$$d(\Psi\hat{p}, \Psi\hat{q}) = \sqrt{\frac{1}{np}\sum_i^{np}\left(\left|\left|C\hat{p} - Cq\right|\right|^2 + ||G\hat{p} - Gq||^2\right.} \qquad (9)$$

where np is the number of pixels in a patch and C is the color vector and G is the image gradient vector. A target patch Ψp is updated by a selected source patch, Ψq

$$\Psi\hat{p}(r) = \begin{cases} \Psi\hat{q} & \forall r \in \Psi\hat{p} \cap T \\ \frac{\Psi\hat{p}(r) + \Psi\hat{q}(s)}{2} & \forall r \in \Psi\hat{p} \cap S \end{cases} \qquad (10)$$

Where r and s are pixels in the target patch and co-located pixel in the source patch, respectively. The method updates whole pixels in the target patch $\Psi\hat{p}$, thus pixels in $\Psi\hat{p}$ that belong to source region and co-located pixels in $\Psi\hat{q}$ are overlapped. It fill overlapped region with the average of target and source pixels. Then, its confidence term and boundary ∂T are updated. Following the region update rule [3], segment labels are introduced in the target segment. Therefore, it can infer the suitable patch size and candidate source region for the target region after a number of iterations.

5. CONCLUSION

Exemplar-based image inpainting gives better results as compare to PDE-based image inpainting. The performance of the exemplar-based image inpainting can be improved either by patch shifting scheme or by using segmentation map.

The best matched patch should be found using target patch which has sufficient known pixels. The target patch is checked for the availability of known pixels before finding the best match patch. If more known pixels are needed, the patch shifting scheme is applied.

The other method utilizes structure and texture information using a segmentation map. The structure and texture information are used to determine appropriate patch size and candidate source region. With this approach, we can reduce the number of iterations and error propagation caused by incorrect matching of source patch

6. REFERENCES

[1] A. Efros and T. Leung, “Texture synthesis by non-parametric sampling,” in International Conference on Computer Vision, 1999, pp. 1033–1038.

[2] M. Bertalm´ ıo, L. A. Vese, G. Sapiro, and S. Osher, “Simultaneous structure and texture image inpainting,” IEEE Transactions on Image Processing, vol. 12, no. 8, pp.882–889, 2003.

[3] A. Criminisi, P. Pérez, and K. Toyama, “Region filling and object removal by exemplar-based image inpainting,” IEEE Transactions on Image Processing, vol. 13, no. 9,pp. 1200–1212, 2004.

[4] P. F. Felzemszwalb and D. P. Huttenlocher, “Efficient graph-based image segmentation,” Int. J. Computer Vision, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[5] Jino Lee, Dong-Kyu Lee and Rae-Hong Park “Robust Exemplar-Based Inpainting Algorithm Using Region Segmentation” IEEE Transactions on Consumer Electronics, Vol. 58, No. 2, May 2012.

[6] J. Sun, L. Yuan, J. Jia, and H.Y. Shum, “Image completion with structure propagation,” *ACM Trans. Graphics*, vol. 24, no. 3, pp. 861–868, Jul. 2005.

[7] 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*, pp. 64–69, Beijing, China, Jul. 2005.

[8] Sarawut Tae-o-sot and Akinori Nishihara , “ Exemplar based image inpainting with patch shifting scheme”, 2011 17th International Conference on Digital signal Processing (DSP).