LETTER

Exemplar-Based Inpainting Driven by Feature Vectors and Region Segmentation

Jinki PARK[†], Jaehwa PARK[†], Young-Bin KWON[†], *Nonmembers*, Chan-Gun LEE[†], *and* Ho-Hyun PARK[†a)], *Members*

SUMMARY A new exemplar-based inpainting method which effectively preserves global structures and textures in the restored region driven by feature vectors is presented. Exemplars that belong to the source region are segmented based on their features. To express characteristics of exemplars such as shapes of structures and smoothness of textures, the Harris corner response and the variance of pixel values are employed as a feature vector. Enhancements on restoration plausibility and processing speedup are achieved as shown in the experiments.

key words: exemplar-based inpainting, Harris corner, feature vector, region segmentation

1. Introduction

The focus of inpainting technology is to create a natural look, from a human viewpoint, when a lost or damaged portion of an image is replaced with a background area.

Criminisi et al. replaced the target region Ω (unknown area due to damage or blocking by another object) from the perimeter to the internal area [1]. This forms a square patch (e.g. Ψ_p in Fig. 1) centered at a point on the perimeter of the target region. This square patch is called the target patch. It then selects the patch most similar to the target patch from the source patches (e.g. Ψ_q, Ψ_r, and Ψ_s in Fig. 1), which are referred to as exemplars, in the source region Φ (known area) and replaces the target patch with the chosen source patch. This process continues towards the center of the target region.

Their algorithm consists of two primary functions, the priority function and the patch-selection function. The priority function determines the target patch that has to be inpainted first from all patches on the perimeter of the target region. Patches with strong structural properties such as lines and corners have higher priority. The patch-selection function searches the source patches for the exemplar most similar to the target patch. The exemplars in the source region are searched by pixel-based comparison, i.e. the SSD (sum of squared difference) between the target patch Ψ_p and a source patch Ψ_q as

$$SSD(\Psi_p, \Psi_q) = \sum_{j \in \Phi_p} |I_p(j) - I_q(j)|^2 \qquad (1)$$
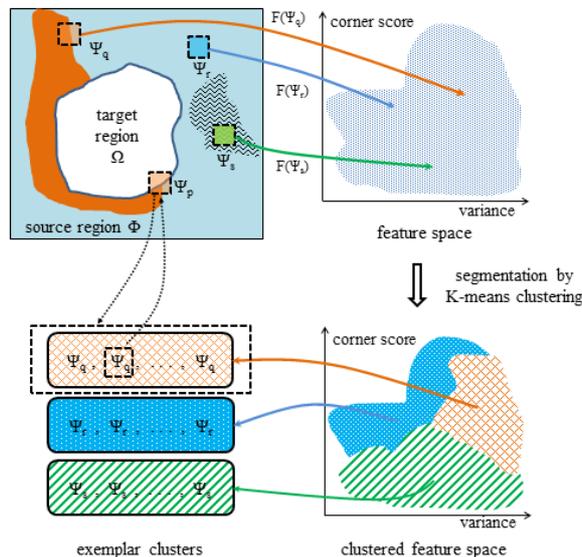
Fig. 1 Inpainting process using feature vectors and segmentation.

In Eq. (1), $I_p(j)$ and $I_q(j)$ represent the jth pixel value in patches Ψ_p and Ψ_q, respectively. Φ_p represents the set of pixels contained only in the source region among the pixels in Ψ_p.

However, human vision is considerably sensitive to structural shape and textural smoothness rather than the pixel-based differences of the exemplars. The novelty of our proposed method lies in the consideration of structural shape and textural smoothness in the design of the patch-selection function. The proposed patch-selection function exploits the Harris response to detect the global structure.

Some algorithms [2]–[4] were proposed to improve the image quality and speed of Criminisi et al.'s method [1]. Among them, Feng et al. [2] also utilized the Harris response, but it was limited to the priority function. For the selection of a source patch, they adopted the multi-resolution approach instead.

2. Background

2.1 Harris Corner

The Harris algorithm detects corners in images [5]. Given a two-dimensional image I, it creates the following matrix M at a point (x, y) according to the movement of a local

window:

$$M = \sum_{u,v} w_{u,v} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \quad (2)$$

$$\text{where} \quad I_x = \frac{\partial I(x,y)}{\partial x}, \quad I_y = \frac{\partial I(x,y)}{\partial y},$$

$$w = \text{pixel weights in a local window}$$

Then, the Harris corner score is defined as

$$h(x,y) = Det(M) - kTr(M)^2 \quad (3)$$

$$\text{where} \quad Det(M) = AB - C^2, \; Tr(M) = A + B$$

In Eq. (3), the constant k is generally set to 0.04. If a region in an image indicates a high Harris corner score, it has strong structural characteristics such as corners and edges. In contrast, if it has values near zero, it implies a planar area.

2.2 Variance of Pixel Values

One approach to describe a region is to quantify its texture. The variance of pixel values is a statistical method for expressing the smoothness of texture. The variance σ_p^2 and mean m_p of a patch Ψ_p are defined as

$$\sigma_p^2 = \sum_{j \in \Psi_p} (I_p(j) - m_p)^2 \big/ |\Psi_p| \; \text{ where } \; m_p = \sum_{j \in \Psi_p} I_p(j) \big/ |\Psi_p|$$

$$(4)$$

The variance represents the combination of the structure and texture of an exemplar. If an exemplar has low variance, it is considered as smooth texture. Conversely, a high variance implies a rough texture or structure such as an edge or corner.

3. Feature Vectors and Region Segmentation

It has been known that an inpainting result is most successful when the inpainting starts from a region where the structural characteristics are strongest on the boundary of the target region. Because [1] and [2] perform in this manner, we will use the priority function of [1] or [2]. The focus of the proposed approach is the patch-selection function.

During the patch-selection phase of [1], [2], SSD is the only metric for selecting a patch as shown in Eq. (1). Generally the SSD works well but it does not always choose the best patch. An exemplar in the source region occasionally has the minimum SSD even though it has different structure and texture from the target patch. This can lead to unnatural inpainting. Therefore, structure and texture must be considered in the patch-selection phase, too.

In the proposed approach, a combination of the Harris corner score and variance of pixel values is used as a pre-screening metric before the SSD comparison. Structural and textural features of all the exemplars are calculated in advance and segmented into several clusters. The global structure and texture of an exemplar Ψ_q are represented by a feature vector $\mathbf{F}$ as follows:

$$\mathbf{F}(\Psi_q) = (|h|, \sigma_q^2) \quad \text{where } |h| = \text{Harris corner score}, \quad (5)$$

$$\text{and } \sigma_q^2 = \text{variance of pixel values}$$

Figure 1 presents the overall inpainting process using feature vector segmentation. Feature vectors $\mathbf{F}$ are extracted for exemplars in the source region, $\Psi_q, \Psi_r, \Psi_s \in \Phi$, and mapped into a feature space. The feature space consists of corner score axis and variance axis that represent $|h|$ and σ_q^2 of $\mathbf{F}$, respectively.

It is acceptable that two exemplars which have a short distance in the feature space have similar characteristics in terms of structure and texture. Based on this assumption, groups of similar exemplars can be generated using a clustering algorithm. As a result, the feature space is segmented into several regions and exemplars are clustered in the feature space. In our implementation, the K-means clustering was adopted [6].

The feature vector of the target patch Ψ_p is also extracted and mapped into a clustered feature space. When selecting an exemplar from the source region, only exemplars in the same cluster as the target patch are compared by the SSD. Searching only exemplars having similar structure and texture to the target patch can avoid the risk of SSD error.

Figure 2 shows an example to form exemplar clusters and to select an exemplar from the clusters using a real image. Figure 2 *a* illustrates a case where we set the inpainted area (green) in the original image. Figure 2 *b* shows the cluster areas roughly in the feature space. Note that the exemplars are divided into four groups since we set K to 4 for K-means clustering.

Cluster 1 is expected that exemplars in this cluster represent flat features which have little structure and texture,
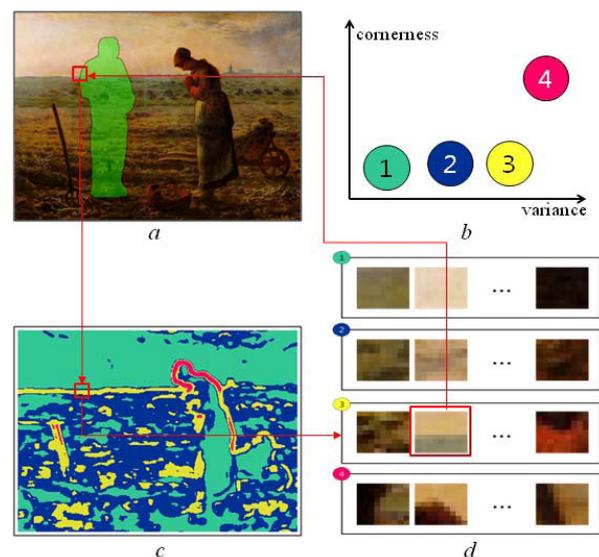
Fig. 2 Process of formation of exemplar clusters and exemplar searching: *a* Image with inpainting domain, *b* Clusters in feature space, *c* Cluster map, *d* Exemplar clusters.

whereas cluster 4 may have noticeable structural characteristics. Figure 2 c shows a cluster map, i.e., mapping the clusters into the original image. Figure 2 d shows some exemplars in each cluster.

The red lines show the process to estimate an exemplar inpainted from the exemplar clusters. As shown in Fig. 2 a, a target patch is first selected among pixels on the boundary of the inpainting domain according to the algorithm of [1]. In Fig. 2 c, the cluster to which the target patch belongs is calculated using the feature vector and cluster map. We select an exemplar with the minimum SSD among the exemplars in the cluster in Fig. 2 d that the target patch belongs to, then fill the inpainting area in Fig. 2 a with the chosen exemplar.

The feature vector clustering provides an additional benefit. Examining exemplars in only one cluster reduces the search time significantly compared to the methods of [1] and [2] that search all the exemplars in the source region.

The feature vector inpainting consists of the following two algorithms. Algorithm 1 describes the initial phase that creates the exemplar clusters for the source patches using feature vectors. Algorithm 2 describes the process of selecting a source patch and replacing the target patch.

Algorithm 1: Creation of exemplar clusters
1: divide the source region Φ into patches $\Psi_{q1}, \ldots, \Psi_{qN}$;
2: for each Ψ_{qi}, $1 \le i \le$ N
3:     calculate Harris corner score $|h|$ using Eq. (3)
        and variance σ_{qi}^2 using Eq. (4);
4:     generate feature vector $F(\Psi_{qi})$ using Eq. (5);
5: segment $F(\Psi_{q1}), \ldots, F(\Psi_{qN})$ into K clusters $C_1, \ldots, C_K$
    using K-means clustering;

Algorithm 2: Inpainting for a target patch
1: select a target patch Ψ_p on the boundary of target
    region Ω using the priority function of [1] or [2];
2: calculate Harris corner score $|h|$ using Eq. (3),
    and variance σ_p^2 using Eq. (4) for Ψ_p;
3: generate feature vector $F(\Psi_p)$ using Eq. (5);
4: C_k = the cluster that Ψ_p belongs to based on K-means
    clustering;
5: for each $\Psi_i \in C_k$, $1 \le i \le |C_k|$ ($|C_k|$ is the number of
               patches in cluster C_k)
6:     calculate $SSD(\Psi_p, \Psi_i)$ using Eq. (1);
7: select a patch Ψ_q such that
    $SSD(\Psi_p, \Psi_q) = \min\{SSD(\Psi_p, \Psi_i), 1 \le i \le |C_k|\}$;
8: replace Ψ_p with Ψ_q;

4. Experiments

We conducted experiments to find the best K value in K-means clustering and to compare the proposed method of feature vector and region segmentation (FVRS) with two previous exemplar-based methods [1], [2]. For all the images, the patch size was set to 9×9 pixels. Due to space limitation, we present only three of the experimental results. Figure 3 shows inpainting results for bungee jump, cactus and beach images. The K values to generate clustered feature maps for FVRS were varied among 2, 4 and 6. The
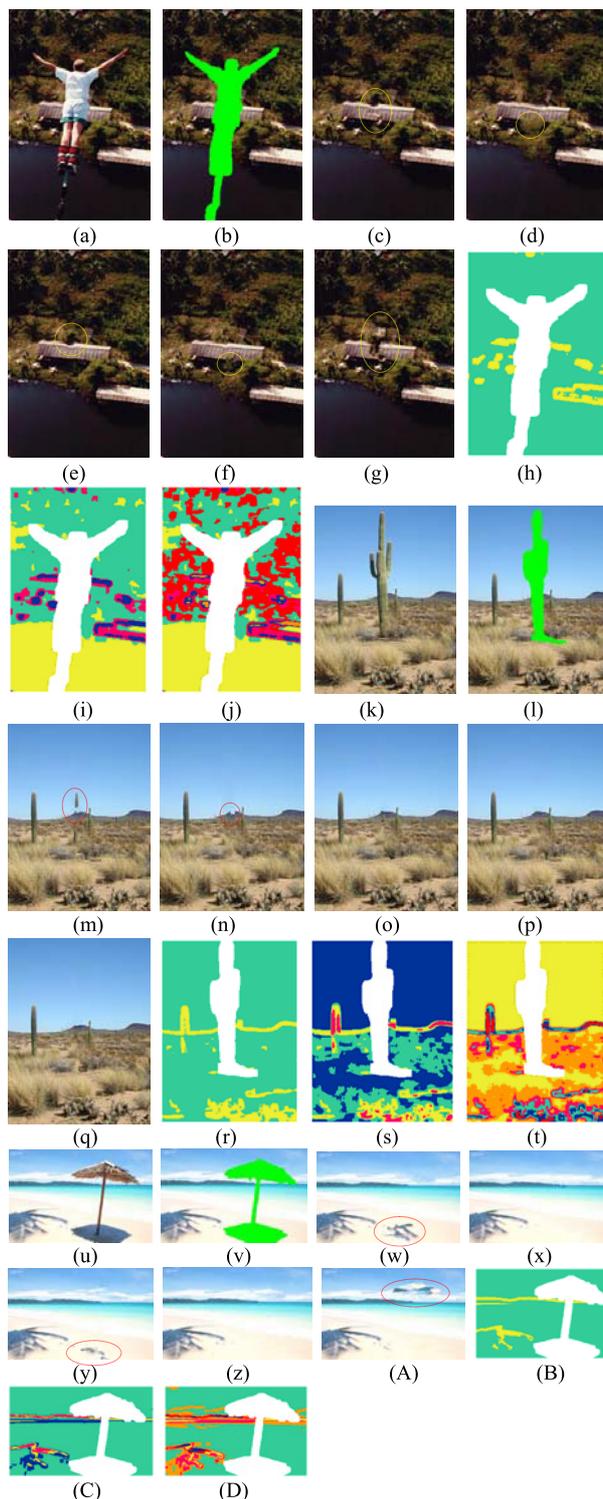
Fig. 3 Experimental results using bunjee jump, cactus and beach images: (a), (k), (u) Original images, (b), (l), (v) Images with unknown region, (c), (m), (w) Results of [1], (d), (n), (x) Results of [2], (e), (o), (y) Results of FVRS with K = 2, (f), (p), (z) Results of FVRS with K = 4, (g), (q), (A) Results of FVRS with K = 6, (h), (r), (B) Clustered feature maps (K = 2), (i), (s), (C) Clustered feature maps (K = 4), (j), (t), (D) Clustered feature maps (K = 6).

Table 1   Execution time of inpainting for three images.

Inpainting method	Execution time (sec)		
	bungee	cactus	beach
Criminisi et al [1]	25.52	17.87	33.61
Fang et al [2]	16.08	13.19	20.63
FVRS with K=2	22.11	13.24	31.04
FVRS with K=4	13.56	9.05	28.19
FVRS with K=6	10.26	7.54	22.54

unnatural inpainting results are marked with yellow or red ellipses.

Note that the images surrounded by the yellow or red ellipse in Fig. 3 (c), (m) and (w), which are results of [1], are very unnatural. Figure 3 (d), (n) and (x), results of [2], are better than those of [1]; but unnatural inpaintings are still visible. These unnatural results are a consequence of SSD error during the patch-selection process. As shown in Fig. 3 (e), (o) and (y), the results of FVRS with K = 2 perform better than [1]; but Fig. 3 (e) and (y) still show similar appearance to [1] because K is small yet.

Viewing Fig. 3 (f), (p) and (z), which are results of FVRS with K = 4, we observe that the connectivity of the structure and texture is significantly improved. Figure 3 (g), (q) and (A) are results of FVRS with K = 6. Images of (g) and (A) are much worse than those with K = 4. This is because as K increases, dividing clusters are too fine grained and some patches in the boundary area between two clusters may be incorrectly assigned. The case with K = 4 shows the best results in bungee jump and beach images while results with K = 2, 4 and 6 are similar in cactus image. According to our experimental results for other images, the best K varied with the types of images; but in most cases, the experiments with K = 4 showed the best results. For the priority function of FVRS, the method of [1] was used in this experiment. However, other methods including [2] could also be used.

Table 1 summarizes the execution time for Fig. 3. The method of [1] requires a significant amount of time because it compares all the exemplars during the patch-selection process. The method of [2] provides better execution time than [1] by the help of multi-resolution. For the images of bungee jump and cactus, FVRS consumed the least execution time as expected. As K increases, the execution time decreases because the search space for patch selection shrinks. Recall that our method reduces overall execution time significantly by examining only exemplars in one cluster. However, it required longer execution time for beach image than the method of [2] because a large number of exemplars are skewed into one cluster from which most of target patches

are selected.

Although FVRS requires some overhead to generate and cluster the feature vectors in the initial phase, it is minor compared to the patch-selection time. For example, Algorithm 1 spends only 0.55s during the total execution time (9.05s) for the cactus image with K = 4. The majority of the time (8.50s) is consumed by Algorithm 2.

5. Conclusion

This study presented a novel scheme to improve the performance of exemplar-based inpainting. The FVRS inpainting preserved the global structure and texture of an image using feature vectors and clustering. The feature vector was represented by a combination of the Harris response and variance of pixel values. The experiments showed that the proposed method improved the visual result as well as the execution time compared to previous exemplar-based methods.

In our experiments, examining the best match cluster for the patch-selection turns out to be sufficient. However, if the visual result of the inpainting is not satisfactory, the candidates can be expanded to adjacent clusters in the clustered feature space at the expense of the execution time.

References

[1] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," IEEE Trans. Image Process., vol.13, no.9, pp.1200–1212, 2004.
[2] C.-W. Fang and J.-J. Lien, "Rapid image completion system using multiresolution patch-based directional and nondirectional approaches," IEEE Trans. Image Process., vol.18, no.12, pp.2769–2779, 2009.
[3] T. Shibata, A. Iketani, and S. Senda, "Fast and structure-preserving image inpainting based on probabilistic structure estimation," IEICE Trans. Inf. & Syst., vol.E95-D, no.7, pp.1731–1739, July 2012.
[4] B. Kim, J. Kim, and J. So, "Pixel and patch reordering for fast patch selection in exemplar-based image inpainting," IEICE Trans. Inf. & Syst., vol.E96-D, no.12, pp.2892–2895, Dec. 2013.
[5] C. Harris and M. Stephens, "A combined corner and edge detector," 4th Alvey Vision Conf., pp.147–151, 1988.
[6] J.B. MacQueen, "Some methods for classification and analysis of multivariate observations," Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability, pp.281–297, 1967.