

Super-Resolution-Based Inpainting

Olivier Le Meur and Christine Guillemot

University of Rennes 1, France; INRIA Rennes, France
olemeur@irisa.fr, christine.guillemot@inria.fr

Abstract. This paper introduces a new exemplar-based inpainting framework. A coarse version of the input image is first inpainted by a non-parametric patch sampling. Compared to existing approaches, some improvements have been done (e.g. filling order computation, combination of K nearest neighbours). The inpainting of a coarse version of the input image allows to reduce the computational complexity, to be less sensitive to noise and to work with the dominant orientations of image structures. From the low-resolution inpainted image, a single-image super-resolution is applied to recover the details of missing areas. Experimental results on natural images and texture synthesis demonstrate the effectiveness of the proposed method.

Keywords: exemplar-based inpainting, super-resolution.

1 Introduction

Image inpainting refers to methods which consist in filling-in missing regions (holes) in an image [1]. Existing methods can be classified into two main categories. The first category concerns diffusion-based approaches which propagate linear structures or level lines (so-called isophotes) via diffusion based on partial differential equations [1, 2] and variational methods [3]. Unfortunately, the diffusion-based methods tend to introduce some blur when the hole to be filled-in is large. The second family of approaches concerns exemplar-based methods which sample and copy best matching texture patches from the known image neighborhood [4–7]. These methods have been inspired from texture synthesis techniques [8] and are known to work well in cases of regular or repeatable textures. The first attempt to use exemplar-based techniques for object removal has been reported in [6]. Authors in [5] improve the search for similar patches by introducing an a priori rough estimate of the inpainted values using a multi-scale approach which then results in an iterative approximation of the missing regions from coarse to fine levels. The two types of methods (diffusion- and exemplar-based) can be combined efficiently, e.g. by using structure tensors to compute the priority of the patches to be filled as in [9].

Although tremendous progress has been made in the past years on inpainting, difficulties remain when the hole to be filled is large and another critical aspect is the high computational time in general required. These two problems are here addressed by considering a hierarchical approach in which a lower resolution

of the input image is first computed and inpainted using a K-NN (K Nearest Neighbours) exemplar-based method. Correspondences between the K-NN low-resolution and high-resolution patches are first learnt from the input image and stored in a dictionary. These correspondences are then used to find the missing pixels at the higher resolution following some principles used in single-image super-resolution methods.

Super-Resolution (SR) refers to the process of creating one enhanced resolution image from one or multiple input low resolution images. The two corresponding problems are then referred to as single or multiple images SR, respectively. In both cases, the problem is of estimating high frequency details which are missing in the input image(s). The proposed SR-aided inpainting method falls within the context of single-image SR on which we thus focus in this section.

The SR problem is ill-posed since multiple high-resolution images can produce the same low-resolution image. Solving the problem hence requires introducing some prior information. The prior information can be an energy functional defined on a class of images which is then used as a regularization term together with interpolation techniques [10]. This prior information can also take the form of example images or corresponding LR-HR (Low Resolution - High Resolution) pairs of patches learnt from a set of un-related training images in an external database [11] or from the input low resolution image itself [12]. This latter family of approaches is known as example-based SR methods [11]. An example-based SR method embedding K nearest neighbours found in an external patch database has also been described in [13]. Instead of constructing the LR-HR pairs of patches from a set of un-related training images in an external database, the authors in [12] extract these correspondences by searching for matches across different scales of a multi-resolution pyramid constructed from the input low-resolution image.

The proposed method thus builds upon earlier work on exemplar-based inpainting in particular on the approach proposed in [4], as well as upon earlier work on single-image exemplar-based super-resolution [12]. However, since the quality of the low-resolution inpainted image has a critical impact on the quality at the final resolution, the inpainting algorithm in [4] is first improved by considering both a linear combination of K most similar patches (K -NN) to the input patch rather than using simply the best match by template matching and K-coherence candidates as proposed in [14]. The impact of different patch priority terms on the quality of the inpainted images is also studied, leading to retain a sparsity-based priority term. In addition, a new similarity measure based on a weighted Battacharya distance is introduced. In a second step, the patches to be filled within the input HR image are processed according to a particular filling order. The algorithm thus proceeds by searching for K nearest neighbours to the input vector concatenating the known HR pixels of the patch and the pixels of the corresponding inpainted LR patch. The K -NN patches are searched in a dictionary composed of LR-HR patches extracted from the known part of the image. The similarity metric is again the weighted Battacharya metric. Similarity weights are also computed between the input and K -NN vectors formed by

the LR and known pixels of the HR patches. Finally, since the inpainted HR patches are overlapping, a seam is searched throughout the overlapping region, and the initially overlapping patches are thus pasted along this seam.

In summary, the proposed method further advances the state-of-the-art in exemplar-based inpainting methods by proposing:

- a new framework which combines inpainting and super-resolution in a two-step approach improving the trade-off between quality and complexity;
- improvements concerning the use of priority terms, the set of candidates (K-NN and K-coherence candidates) and distance metrics.

The paper is organized as follows. In Section II, the new framework of the proposed inpainting method is presented. Section III elaborates the proposed exemplar-based inpainting and presents experimental results in comparison with those produced by with state-of-the-art methods. In Section IV, the details of the SR-aided inpainting method are introduced. Section V presents the performance of the proposed method as well as comparisons with state-of-the-art methods. Finally, we conclude this work in Section V.

2 Algorithm Overview

Image completion of large missing regions is a challenging task. As presented in the previous section, there are a number of solutions to tackle the inpainting problem. In this paper, we propose a new inpainting method using a single-image SR algorithm. In the following sections, we briefly present the main ideas of this paper and the reasons why the proposed method is new and innovative.

2.1 Motivations

The proposed method is composed of two main and sequential operations.

The first one is a non-parametric patch sampling method used to fill-in missing regions. However, rather than filling in missing regions at the original resolution, the inpainting algorithm is applied on a coarse version of the input picture. There are several reasons for performing the inpainting on a low-resolution image. First, the coarse version of the input picture could be compared to a gist [15] representing dominant and important structures. Performing the inpainting of this coarse version is much easier since the inpainting would be less contingent on local singularities (local orientation for instance) or even noise. Second, as the picture to inpaint is smaller than the original one, the computational time to inpaint it is significantly reduced compared to the one necessary to inpaint the full resolution image.

The second operation is run on the output of the first step. Its goal is to enhance the resolution and the subjective quality of the inpainted areas. We use a single-image SR approach. Given a low-resolution input image, which is the result of the first inpainting step, we recover its high-resolution using a set of training examples, which are taken from the known part of the input picture.

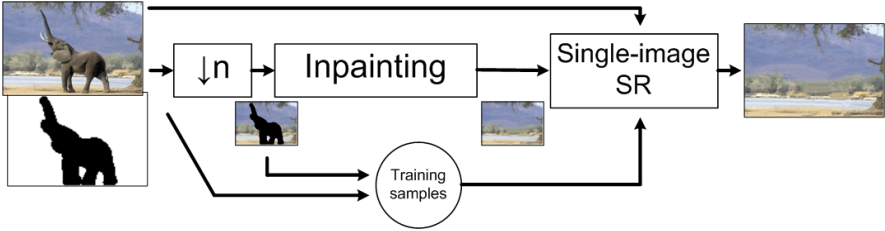


Fig. 1. The framework of the proposed method

This new method is generic since there is no constraint on both the number and the type of inpainting methods used in the first pass. The better the inpainting of low-resolution images, the better the final result should be. Regarding the number of methods, one could imagine using different settings (patch size, search windows etc) or methods to fill-in the low-resolution images and to fuse results. We believe that it would increase the robustness and the visual relevance of inpainting, as recently proposed by Bugeau et al. [16]. They have indeed shown that the best inpainting results are obtained by a combination of three different methods. In this paper, due to the limited space, we will not consider the possibility to combine the results of different inpainting algorithms, but rather focus on the proposed SR-aided inpainting framework.

2.2 Principle

Figure 1 illustrates the main concept underlying the proposed method. The two main components are the inpainting and the super-resolution algorithms. More specifically, the following steps are performed:

1. a low-resolution image is first built from the original picture;
2. an inpainting algorithm is applied to fill-in the holes of the low-resolution picture;
3. the quality of the inpainted regions is improved by using a single-image SR method.

3 Exemplar-Based Inpainting of Low-Resolution Images

This section presents the inpainting method which is used in this paper to fill in the low-resolution images. This is an adaptation of the Criminisi et al. [4] method. The influence of different priority terms on the quality of the inpainted images is first studied. A similarity metric based on a weighted Bhattacharya distance is proposed. The resulting inpainting algorithm is compared against two state-of-the-art methods. The first one is also based on a non-parametric patch sampling (Patch Match, [7]) whereas the second one is based on partial derivatives equations [2]. We have chosen these two methods because of their relevance

and because the code is available. The proposed exemplar-based method follows the two classical steps as described in [4]: the filling order computation and the texture synthesis. These are described in the next sections.

3.1 Patch Priority and Filling Order

The filling order computation defines a measure of priority for each patch in order to distinguish the structures from the textures. Classically, a high priority indicates the presence of structure. The priority of a patch centered on p is just given by a data term (the confidence term proposed in [4] is not used here since it does not bring about any improvement). Three different data terms have been tested: gradient-based priority [4], tensor-based [9] and sparsity-based [17].

The sparsity-based priority has been proposed recently by Xu et al. [17]. In a search window, a template matching is performed between the current patch ψ_p and neighbouring patches ψ_{p,p_j} that belong to the known part of the image. By using a non-local means approach [18], a similarity weight w_{p,p_j} (i.e. proportional to the similarity between the two patches centered on p and p_j) is computed for each pair of patches. The sparsity term is defined as:

$$D(\mathbf{p}) = \|\mathbf{w}_p\|_2 \times \sqrt{\frac{|N_s(\mathbf{p})|}{|N(\mathbf{p})|}} \quad (1)$$

where N_s and N represent the number of valid patches (having all its pixels known) and the total number of candidates in the search window. When $\|\mathbf{w}_p\|_2$ is high, it means larger sparsity whereas a small value indicates that the current input patch can be efficiently predicted by many candidates. As illustrated in figure 2, the sparsity-based priority is more robust and visually improves the



Fig. 2. Inpainting of LR pictures with different gradient-based priority [4] (first row), tensor-based priority [9] (second row) and sparsity-based priority [17] (third row).

final result compared to the gradient and tensor-based priority. In the following, we adopt this method to compute the filling order.

3.2 Texture Synthesis

The filling process starts with the patch having the highest priority. Two sets of candidates are used to fill in the unknown part of the current patch. A first set is composed of the K most similar patches located in a local neighbourhood centered on the current patch. They are combined by using a non-local means approach [18]. The weighting factors are classically defined as follows:

$$w_{p,p_j} = \exp\left(-\frac{d(\psi_p, \psi_{p,p_j})}{h}\right) \tag{2}$$

where $d()$ is a metric indicating the similarity between patches, and h is a decay factor. These weights are then normalized as $w_{p,p_j} / \sum_k w_{p,p_k}$. The number of neighbours is adapted locally so that the similarity of chosen neighbours lies within a range $(1 + \alpha) \times d_{min}$, where d_{min} is the distance between the current patch and its closest neighbour, α is equal to 0.75.

As mentioned by [8], a major problem of local neighbourhood search is its tendency to get stuck at a particular place in the sample image and to produce verbatim copying. This kind of regions is often called garbage region. This problem can be addressed by introducing some constraints in terms of spatial coherence. The idea is based on the fact that patches that are neighbours in the input image should be also neighbours in the output image [14, 19]. Figure 3 a) illustrates this process. With a 8-connexity neighbourhood centered on the current patch (noted C on figure 3 a)), K_i patches are used as candidates and compared to the best candidate obtained by the local neighbourhood search. Figure 3 b) and c) show the influence of the k-coherence method on the quality of the low-resolution inpainted image. The use of k-coherence candidates improves locally the quality on many parts of the pictures.

Concerning the similarity measure, we have considered two metrics: the classical Sum of Squared Differences (d_{SSD}) and a weighted Bhattacharya distance as the one proposed in [16] ($d_{(SSD,BC)}$). The last metric is defined as follows:

$$d_{(SSD,BC)}(\psi_p, \psi_{p,p_j}) = d_{SSD}(\psi_p, \psi_{p,p_j}) \times (1 + d_{BC}(\psi_p, \psi_{p,p_j})) \tag{3}$$

where, $d_{BC}(\psi_p, \psi_{p,p_j})$ is a modified version of the classical Bhattacharya distance as described in [16] ($d_{BC}(\psi_p, \psi_{p,p_j}) = \sqrt{1 - \sum_k \sqrt{p_1(k)p_2(k)}}$ where p_1 and p_2 represent the histograms of patches ψ_p, ψ_{p,p_j} , respectively). This is not exactly the same formulation as in [16]: indeed Bugeau et al. directly multiply the SSD distance with d_{BC} . This presents a drawback: for two patches having the same distribution, no matter how the rotation the value d_{BC} is null, leading to a null distance $d_{(SSD,BC)}$. With the proposed metric, the distance is equal to the SSD distance.

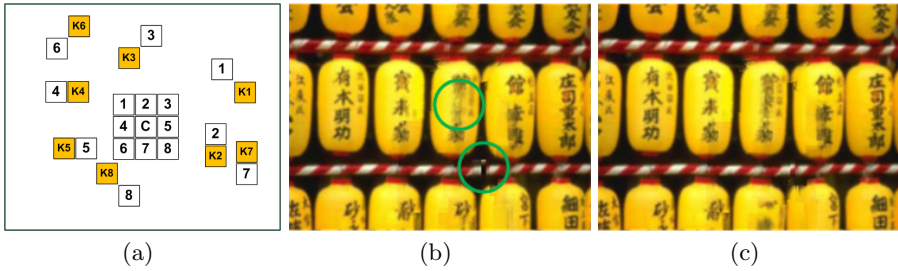


Fig. 3. (a) K-coherence algorithm: candidates shown in orange are used as predictors; (b) and (c) are inpainted images when the k-coherence method is disabled or enabled respectively. Green circles stress the major differences between pictures (b) and (c).

3.3 Other State-of-the-Art Methods

To compare the proposed inpainting method to existing ones, we have chosen methods for which either the source code or an executable file are available.

Criminisi et al. Method: Criminisi et al. [4] proposed to guide the filling process with a priority term based on edge strength. A modified version of the Matlab software available on <http://www.cc.gatech.edu/~sooraj/inpainting/> is used to perform the inpainting.

Diffusion-Based Method: as introduced earlier, diffusion-based methods propagate the structures into missing regions. For comparison purposes, we use in this paper the approach proposed in [2].

Patch Match: Published in 2009 [7], the PatchMatch method is a fast algorithm for computing dense approximate nearest neighbour correspondences between patches of two image regions. This algorithm is available in Adobe Photoshop CS5 and works well. We will systematically compare our results to Patch Match's ones.

Note that we have also tested a method called tensor completion [20]. However, when the missing area is too large, the inpainting quality is low. We then put aside this method.

3.4 Comparison between Proposed and State-of-the-Art Methods

Figure 4 (e) and (f) illustrate the performance of the proposed method on various low-resolution pictures. These pictures are downsampled versions of the original one (the down sampling factor is equal to 4 in both directions for (e) and equal to 2 for (f)). Results from [7] and [2] are also depicted. As expected, the diffusion-based approach retrieves the main structures of the scene (except for the third and fifth pictures). However, it tends to smooth the textured regions. Results obtained by the proposed and the Patch Match methods are comparable although artefacts are not the same (for Patch Match: a man is duplicated (see second row) and some grass appears on the rock (last row); for the proposed method (e): the picture on the third row presents more artefacts).

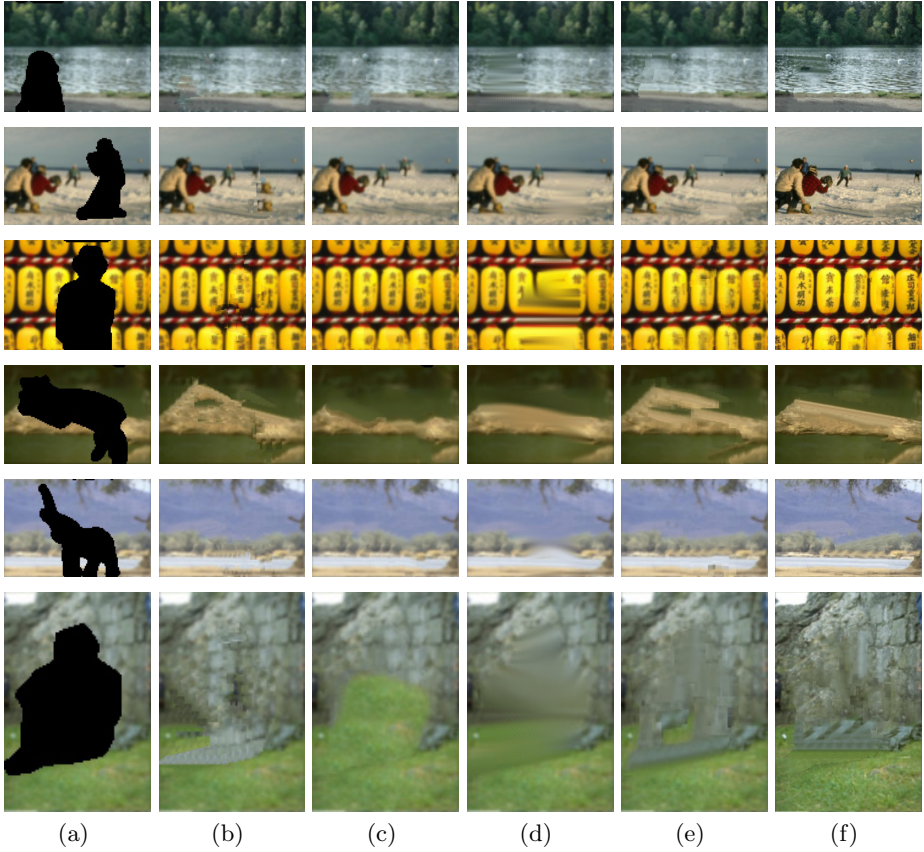


Fig. 4. (a) low-resolution pictures with missing areas in black; (b) Criminisi et al.'s results; (c) Patch Match results; (d) Diffusion-based results; (e) Proposed method (the down sampling factor is set to $n = 4$; patch size is 11×11); (f) Proposed method with $n = 2$ and patch's size of 15×15

4 Super-Resolution Algorithm

Once the inpainting of the low-resolution picture is completed, a single-image super-resolution approach is used to reconstruct the high resolution of the image. The idea is to use the low-resolution inpainted areas in order to guide the texture synthesis at the higher resolution. As in [11], the problem is to find a patch of higher-resolution from a database of examples. The main steps, illustrated in figure 5 are described below:

1. Dictionary building: it consists of the correspondences between low and high resolution image patches. The unique constraint is that the high-resolution patches have to be valid, i.e. entirely composed of known pixels. In the proposed approach, high-resolution and valid patches are evenly extracted from

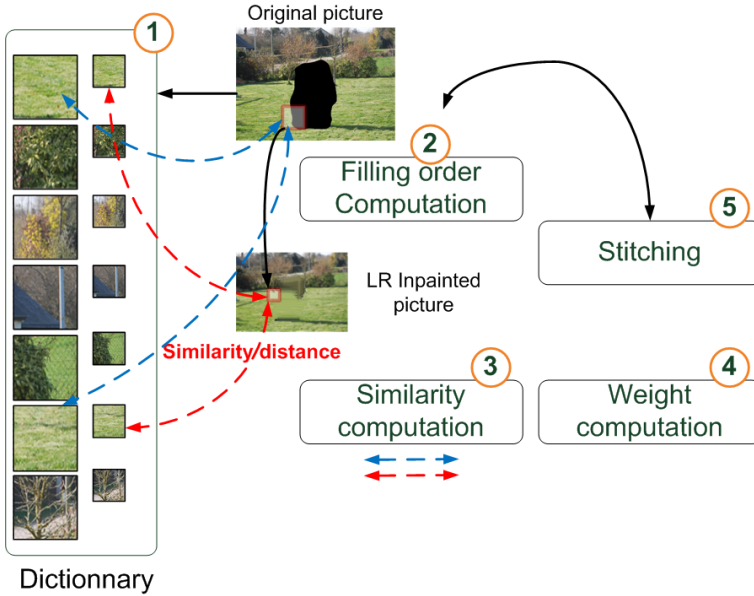


Fig. 5. Flowchart of the super-resolution algorithm. The missing parts of the red block is filled by a linear combination of K HR-candidates (green arrows). The weights are computed using the similarity distance between LR and HR patches (green and red arrows, respectively). The top image represents the original image with the missing areas whereas the bottom one is the result of the low-resolution inpainting.

the known part of the image. The size of the dictionary is a user-parameter which might influence the overall speed/quality trade-off. An array is used to store the spatial coordinates of HR patches (\mathcal{D}^{HR}). Those of LR patches are simply deduced by using the decimation factor;

2. Filling order of the HR picture: the computation of the filling order is similar to the one described in Section 3. It is computed on the HR picture with the sparsity-based method. The filling process starts with the patch ψ_p^{HR} having the highest priority. This improves the quality of the inpainted picture compared to a raster-scan filling order;
3. For the LR patch corresponding to the HR patch having the highest priority, its K -NN in the inpainted images of lower resolution are sought. The number of neighbours is computed as described in the previous section. The similarity metric is also the same as previous;
4. Weights w_{p,p_j} are calculated by using a non-local means method as if we would like to perform a linear combination of these neighbours. However, the similarity distance used to compute the weights is composed of two terms: the first one is classical since this is the distance between the current LR patch and its LR neighbours, noted $d(\psi_p^{LR}, \psi_{p,p_j}^{LR})$. The second term is the distance between the known parts of the HR patch ψ_p^{HR} and the HR patches

corresponding to the LR neighbours of ψ_p^{LR} . Say differently, the similarity distance is the distance between two vectors composed of both pixels of LR and HR patches. The use of pixel values of HR patches allows to constraint the nearest neighbour search of LR patches.

5. A HR candidate is finally deduced by using a linear combination of HR patches with the weights previously computed:

$$\psi_p^{HR} = \sum_{p_j \in \mathcal{D}^{HR}} w_{p,p_j} \times \psi_{p,p_j} \quad (4)$$

with the usual conditions $0 \leq w_{p,p_j} \leq 1$, and $\sum_k w_{p,p_k} = 1$.

6. **Stitching:** the HR patch is then pasted into the missing areas. However, as an overlap with the already synthesized areas is possible, a seam cutting the overlapped regions is determined to further enhance the patch blending. The minimum error boundary cut [21] is used to find a seam for which the two patches match best. The similarity measure is the Euclidean distance between all pixel values in the overlapping region. More complex metrics have been tested but they do not substantially improve the final quality. At most four overlapping cases (Left, Right, Top and Bottom) can be encountered. There are sequentially treated in the aforementioned order. The stitching algorithm is only used when all pixel values in the overlapping region are known or already synthesized. Otherwise, the stitching is disabled.

After the filling of the current patch, priority value is recomputed and the aforementioned steps are iterated while there exist unknown areas.

5 Experimental Results

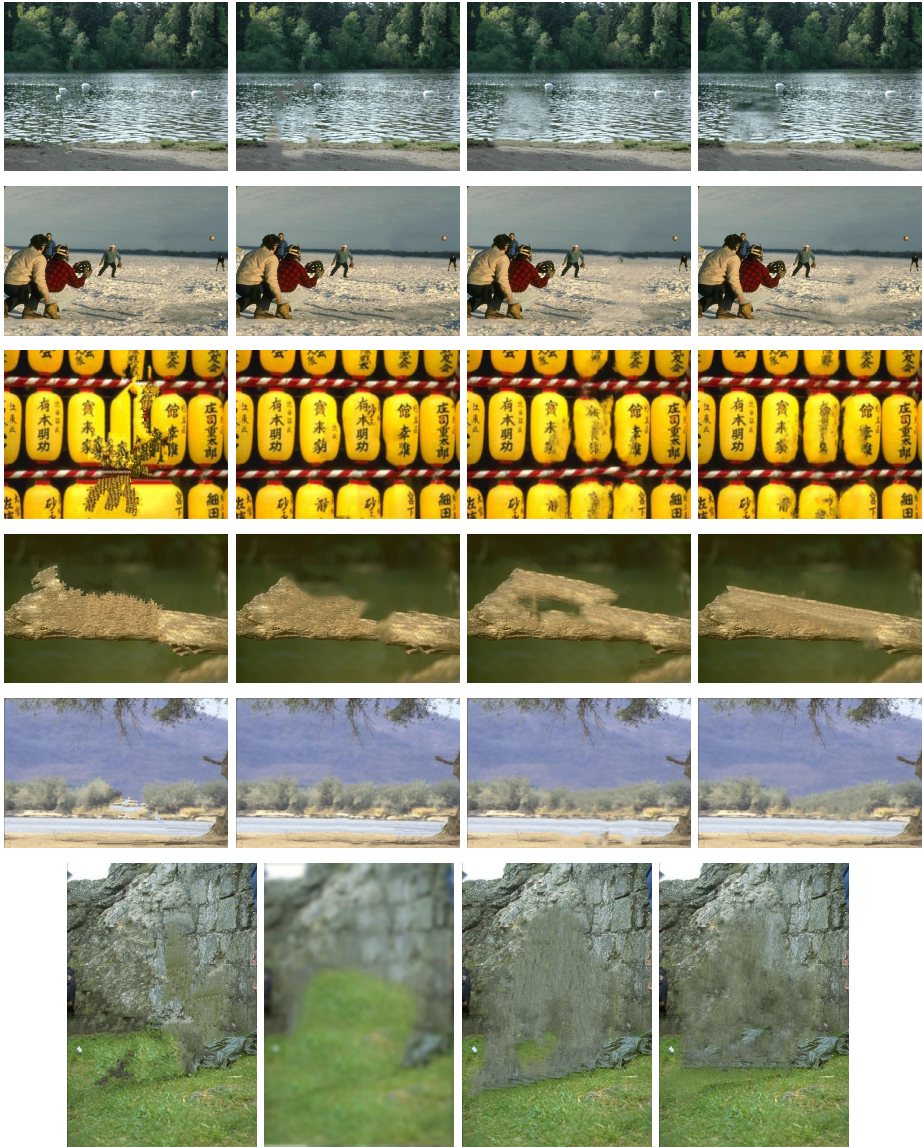
In order to assess the performance of the proposed approach, the parameters of the algorithm are kept constant for the tests presented in this paper.

5.1 Implementation Details and Parameters

Reproducible Research: It is possible to reproduce results by using the executable software, the masks and pictures available on authors' web page.

Parameters: Two versions of the proposed method are evaluated. One uses a down sampling factor of 4 in both directions (the patch size is equal to 5×5) whereas this factor is set to 2 for the second version (the patch size is equal to 7×7). For both versions, the size of the dictionary is the same and can contain at most 6000 patches evenly distributed over the picture. The LR patch size is 3×3 and the HR patch size is 15×15 .

Line Front Feathering: in spite of the use of stitching method, the front line which is the border between known and unknown areas can still be visible. It is possible to hide this transition by feathering the pixel values across this seam. A Gaussian kernel is used to perform the filtering.



(a) Criminisi (b) PatchMatch (c) Proposed (d) Proposed

Fig. 6. Comparison of the proposed method with state-of-the-art approaches: (a) Criminisi et al. [4]; (b) Patch Match [7]; (c) proposed method ($n = 4$), ($n = 2$) (f)

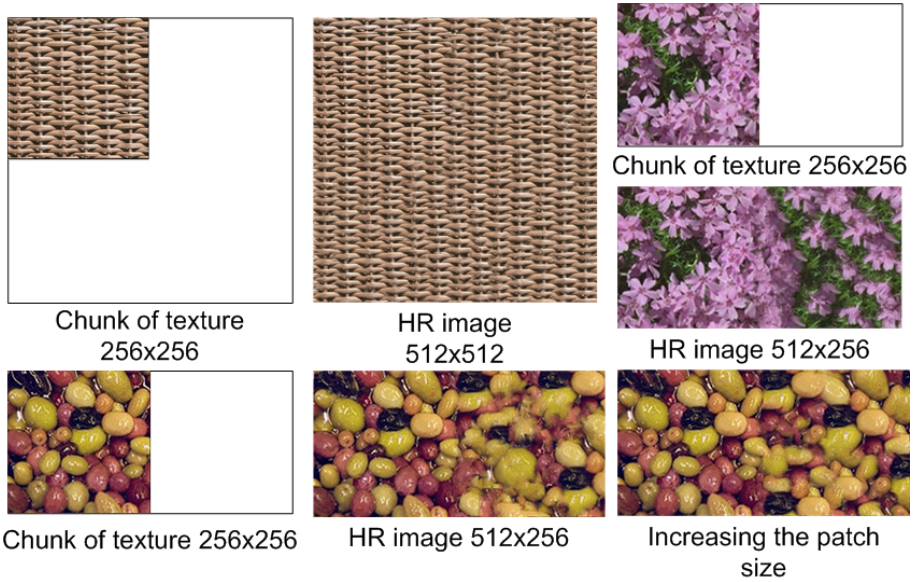


Fig. 7. Texture synthesis (down sampling factor $n = 2$)



Fig. 8. For these pictures, a large missing region has been inpainted (masks are not intentionally given but are available as supplementary materials)

5.2 Comparison with State-of-the-Art Methods

Figure 6 illustrates the comparison between the proposed methods and state-of-the-art methods. The proposed method (for both settings (e) and (f)) provides similar results to Patch Match and visually outperforms Criminisi's approach. Figure 8 gives further results. For these examples, a large missing area has been filled in. Figure 7 presents additional results for texture synthesis. A small chunk of texture (in the example 256×256) was placed into the upper left corner of an empty image. Figure 7 illustrates the performance of the proposed method for these kinds of texture. For deterministic textures, results are very good. For stochastic ones, some artefacts are visible. However, increasing the patch size would cope with these artefacts, as illustrated on the bottom-right of figure 7. The running time on a 3 GHz CPU is less than one minute for pictures having a resolution of 512×512 .

6 Conclusion

In this paper we have introduced a new inpainting framework which combines non-parametric patch sampling method with a super-resolution method. We first propose an extension of a well-known exemplar-based method (improvements are sparsity-based priority, K-coherence candidates and a similarity metric adapted from [16]) and compare it to existing methods. Then, a super-resolution method is used to recover a high resolution version. This framework is interesting for different reasons. First the results obtained are within the state-of-the-art for a moderate complexity. Beyond this first point which demonstrates the effectiveness of the proposed method, this framework can be improved. For instance, one interesting avenue of future work would be to perform several inpainting of the low-resolution images and to fuse them by using a global objective function. First, different kinds of inpainting methods (patch-based or PDE-based) could be used to fill-in the missing areas of a low-resolution image. Second, for a given inpainting method, one can envision to fill-in the missing areas by using different settings e.g. for the patch size in order to better handle a variety of textures and to better approach the texture element sizes. Finally, we believe that the proposed framework will be appropriate for video completion. This application is indeed very time-consuming. The use of the proposed framework could dramatically reduce the computational time.

References

1. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: SIG-GRAPH 2000 (2000)
2. Tschumperlé, D., Deriche, R.: Vector-valued image regularization with pdes: a common framework for different applications. *IEEE Trans. on PAMI* 27, 506–517 (2005)
3. Chan, T., Shen, J.: Variational restoration of non-flat image features: models and algorithms. *SIAM J. Appl. Math.* 61, 1338–1361 (2001)
4. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. on Image Processing* 13, 1200–1212 (2004)
5. Drori, I., Cohen-Or, D., Yeshurun, H.: Fragment-based image completion. *ACM Trans. Graph.* 22, 303–312 (2003)
6. Harrison, P.: A non-hierarchical procedure for re-synthesis of complex texture. In: *Proc. Int. Conf. Central Europe Comp. Graphics, Visua. and Comp. Vision* (2001)
7. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 28 (2009)
8. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: *International Conference on Computer Vision*, pp. 1033–1038 (1999)
9. Le Meur, O., Gautier, J., Guillemot, C.: Exemplar-based inpainting based on local geometry. In: *ICIP* (2011)
10. Dai, S., Han, M., Xu, W., Wu, Y., Gong, Y., Katsaggelos, A.: Softcuts: a soft edge smoothness prior for color image super-resolution. *IEEE Trans. on Image Processing* 18, 969–981 (2009)

11. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. *IEEE Computer Graphics and Applications* 22, 56–65 (2002)
12. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: 2009 IEEE 12th International Conference on Computer Vision (ICCV), vol. 10, pp. 349–356 (2009)
13. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: *Computer Vision and Pattern Recognition*, vol. I, pp. 275–282 (2004)
14. Ashikhmin, M.: Synthesizing natural textures. In: *I3D 2001* (2001)
15. Oliva, A., Torralba, A.: Building the gist of a scene: the role of global image features in recognition. *Progress in Brain Research: Visual Perception* 155, 23–36 (2006)
16. Bugeau, A., Bertalmío, M., Caselles, V., Sapiro, G.: A comprehensive framework for image inpainting. *IEEE Trans. on Image Processing* 19, 2634–2644 (2010)
17. Xu, Z., Sun, J.: Image inpainting by patch propagation using patch sparsity. *IEEE TIP* 19, 1153–1165 (2010)
18. Wexler, Y., Shechtman, E., Irani, M.: Space-time video completion. In: *Computer Vision and Pattern Recognition, CVPR* (2004)
19. Tong, X., Zhang, J., Liu, L., Wang, X., Guo, B., Shum, H.Y.: Synthesis of bidirectional texture functions on arbitrary surfaces. In: *SIGGRAPH 2002*, pp. 6635–6672 (2002)
20. Liu, J., Musialski, P., Wonka, P., Ye, J.: Tensor completion for estimating missing values in visual data. In: *International Conference on Computer Vision*, pp. 2114–2121 (2009)
21. Efros, A.A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: *SIGGRAPH*, pp. 341–346 (2001)