Image Inpainting Based on Local Optimisation

Jun Zhou Antonio Robles-Kelly
Canberra Research Lab, NICTA, Australia
{jun.zhou, antonio.robles-kelly}@nicta.com.au

Abstract

In this paper, we tackle the problem of image inpainting which aims at removing objects from an image or repairing damaged pictures by replacing the missing regions using the information in the rest of the scene. The image inpainting method proposed here builds on an exemplar-based perspective so as to improve the local consistency of the inpainted region. This is done by selecting the optimal patch which maximises the local consistency with respect to abutting candidate patches. The similarity computation generates weights based upon an edge prior and the structural differences between inpainting exemplar candidates. This treatment permits the generation of an inpainting sequence based on a list of factors. The experiments show that the proposed method delivers a margin of improvement as compared to alternative methods.

1. Introduction

Image inpainting is a technique that aims at removing elements in an image or repairing damaged photographs using information available from the scene [1]. It has widespread applications in image and video processing, as well as texture synthesis tasks [4]. Inpainting techniques range from the automatic methods [5] to those labour intensive retouching methods available in digital imaging suites such as Photoshop and Corel Draw. Nevertheless, the main bulk of the image inpainting literature verses upon semisupervised methods, where the user provides a mask that serves as a label for the region to be restored. This is reminiscent of interpolation approaches [5] used in film restoration and the global optimisation method in [10].

Semisupervised methods often use image statistics to recover the missing information on the region to be inpainted. For instance, Bertalmio et al. [2] propagates information from the inpainting region boundary in a clear analogy to the basic concepts of expert inpainting domain knowledge. Oliveira et al. [6] make use of diffusion kernel over the inpainting region boundary and an iterative convolution scheme to tackle the image restoration task. In [6], edges are preserved so as to prevent the loss of detail in a setting akin to anisotropic diffusion methods [7]. These methods, despite being intuitively correct, cannot restore large image regions and, nonetheless the use of edge information, are prone to the loss of detail due to over-smoothing. To overcome these drawbacks, Criminisi et al. [3] make use of exemplars, i.e. image patches or blocks sampled from across the image, so as to remove large objects from the scene. The use of exemplars is reminiscent to that of image priors [8]. Exemplar and prior-based methods are sensitive to the filling sequence within the inpainted region. That is, as the inpainted region is filled, information has to be propagated from the boundary inwards in a manner such that detail is preserved. The choice on the inpainting sequence is a non-trivial problem since details should be preserved while abutting patches have to be consistent with one another.

In this paper, we extend the exemplar-based method in [3] to a local setting, where candidate image patches for a selected location on the boundaries of the missing regions are chosen following a list of factors aimed at preserving image sharpness and detail. We aim at preserving the local consistency of the inpainted region by exploring the relationship between a set of candidate inpainting patches and the neighbourhood of the missing region. The optimal patches are recovered by maximising the local consistency with respect to the neighborhood candidates and the inpainting region boundary pixels. This step uses the information conveyed by the image-patch structure so as to recover the missing region. The experimental results show that the proposed method delivers results whose inpainted regions are in better qualitative accordance with the image than those
yielded by other alternatives.

2. Local Optimisation

Starting from a missing region \( \Phi \) in an image, we define the patch surrounding an inpainted region boundary pixel \( p \in \Phi \) as \( \Psi_p \). In this section, we aim at selecting a patch \( x_p^* \) that maximises the local consistency. As mentioned before, this can be achieved by preserving the image sharpness and detail, while considering the relationship between the filling patches.

First, note that the structural differences between the optimal patch to the known inpainting region boundary pixels shall be minimised. This is reminiscent of other methods elsewhere in the literature, where patch selection is effected based upon a search region. Moreover, as pointed out by Criminisi et al [3], local structure and texture over the image have great impact on the inpainting results, which can be partly characterised by the edge intensity and fill-in priority in the target area. This priority determines the inpainting sequence which in turn affects the manner in which information "propagates" over the region. However, when a set of neighboring boundary pixels have the same priorities, determining the inpainting sequence is not a straightforward task. This can be further extended to the differences between neighbouring pixel priorities, which, ideally, should be small. Again, this is not a straightforward task since, for instance, inpainting from the pixel with highest priority may cause loss of detail or a suboptimal sequence over the entire inpainting region.

To solve this problem, we introduce a further constraint, i.e., the optimal patch \( x_p^* \) should be consistent with those patches used to fill-in neighboring pixels \( x_p^* \) in the inpainting region. This requires minimising the differences between \( x_p^* \) and \( x_p^* \). Thus, in contrast to other methods in which only one optimal patch is selected from the known region in the image for each \( \Psi_p \), we maintain a list of supporting candidate patches \( X_p \) for a given pixel \( p \) and denote the collection of candidate patches for the neighborhood pixels of \( p \) as \( X_p \).

Being more formal, we can express the above process as that of minimising a joint distribution of the inpainted region given by

\[
P(\Psi_p, x_p, x_{\bar{p}}) = P(\Psi_p, x_{\bar{p}} | x_p)P(x_p)
\]

where \( x_p \in X_p \) and \( x_{\bar{p}} \in X_{\bar{p}} \).

Since the selection process of \( x_{\bar{p}} \) is independent from \( \Psi_p \), the above equation can be rewritten as follows

\[
P(\Psi_p, x_p, x_{\bar{p}}) = P(\Psi_p | x_p)p(x_{\bar{p}} | x_p)p(x_p)
\]

and further

\[
P(\Psi_p, x_p, x_{\bar{p}}) = P(\Psi_p | x_p)P(x_p | x_{\bar{p}})P(x_{\bar{p}})\]

It is worth noting that by having a closer look at equation 3, we can relate the three terms on the right-hand side of the equation to the three requirements over the inpainted pixels described above. First, note that \( P(\Psi_p | x_p) \) reflects the similarity between the candidate patches to the target region. Thus, we can define this term as a function of the square distances between \( x_p \) and \( \Psi_p \), i.e.

\[
P(\Psi_p | x_p) = \exp\left(-d(x_p, \Psi_p)^2\right)
\]

Also, note that \( P(x_p | x_{\bar{p}}) \) corresponds to the differences between neighboring filling patches. The global optimal of such differences can be obtained through methods based on dynamic programming, such as the Viterbi algorithm [9]. Here, to reduce the computational cost and achieve a fast inpainting method we instead use a relaxed strategy where we define the consistency as the minimum differences of a candidate patch for the target region with respect to all the candidate patches in the neighborhood. Thus, we have

\[
P(x_p | x_{\bar{p}}) = \min_{x_p \in X_p} \exp\left(-d(x_p, x_{\bar{p}})^2\right)
\]

Finally, \( P(x_{\bar{p}}) \) defines the edge priority of the neighboring patches, which is computed using the method in [3].

With each component of the Equation 3 at hand, the optimal filling patch for the pixel \( p \) can be obtained by minimising the joint distribution above. Note that there can be several neighboring boundary pixels to \( p \), the optimisation is then achieved by considering the local consistency of the filling patch with respect to those of the neighbors

\[
P(\Psi_p, x_p^*, x_{\bar{p}}) = \min_{x_p \in X_p} \sum_{i=1}^{K} P(\Psi_p, x_p, x_{\bar{p}})
\]

where \( K \) is the number of neighbor boundary pixels.

Please note that, in contrast with the method in [10], which hinges in a global optimisation scheme and, hence, is computationally expensive, our method can be treated as a divide-and-conquer one. It divides the whole inpainting region into small neighborhoods around the boundary pixels and operates on the local neighbourhood to each of them. Thus, the steps of the proposed method are summarised as follows:

1. Select the pixel on the boundary of the missing area with the highest filling priority using method in [3].

2. Extract a list of patches from locations in the rest of the image other than the missing area for the selected pixel.
3. Extract patches for neighboring boundary pixels

4. Compute the local consistency value so as to select the optimal supporting patch for the current boundary pixel as introduced in the above section.

5. Go back to 1 until the full region has been inpainted.

3. Experimental Results

In this section, we report the experimental results using the proposed local optimisation approach. We compared our method with the exemplar-based method reported in [3] and the global optimisation method reported in [10]. We implemented the proposed method and that in [3] in-house, whereas software implementing the algorithm in [10] can be downloaded from [11]. Note that, for our method, after obtaining the supporting patches at a boundary pixel, we construct the list of supporting patches by keeping the 10-highest $P(\Psi_p | x_p)$ for the neighbourhood patches. We have done this for the sake of computational efficiency. Also, we only extract image patches from areas that are not occluded by the objects to be removed, i.e. we only use full patches, and set the patch size to be $9 \times 9$ pixels.

For our first experiment, we show a scene that contains rich context of textures, which has been widely used in the inpainting papers to show the effectiveness of a number of methods. Figure 1(a) is a scene that is occluded by a person who is jumping on a bungee, with manual input mask displayed in Figure 1(b). Figures 1(c), 1(d), 1(e) display the results yielded by the methods in [3], [10] and our local optimisation approach, respectively. From the panels, we can see that the results from the proposed method and [10] resemble closely one another. Moreover, nonetheless the person has been completely removed, while the natural background that contains the house, water body and bushland has been restored naturally, our method has preserved the curved bush region on the water shore, which the alternative in [10] has abruptly “cut”. Further, the result yielded by the algorithm in [3] contains some defects. For example, the house has an indent, the region above the house contains a white block and bushes below the house stretch into the water body in an unnatural manner. This is due to the fact that in the inpainting process proposed in [3] only considers the current boundary pixel at process time. While both, our method and the one in [10] employ neighborhood consistency measures. The improvement of our method over that in [10] can be attributed to the further constraint over the local neighbour optimality given a list of support patches as candidates. This assures that the bush on the water shore would be naturally continued as induced by the local neighbourhood.

In the next example, we show another image in which we attempt to remove a flamingo from the foreground. The upper row of Figure 2 shows the original image and mask, while the other panels show the result obtained when using the method in [3], the algorithm in [10] and our inpainting approach. Note that the method in [3] does not correctly inpaint the central region of the image. Despite the method of Wexler et al. [10] yields better results, a closer look at the details of the inpainting results in Figure 3 reveals that it has over-smoothed some of the regions in the image. This is due to the fact that in the inpainting process, as presented in [10], the priority values for the image patches is set

![Image](image.png)
by a global iterative optimisation. As a result, inpainting patches that are smooth over the neighbourhood are privileged over textured ones as the optimisation process progresses. This ultimately yields results in which edges and details diminish after inpainting if the method does not converge in few iterations. Our method, on the other hand, assigns weights to the image patches in the neighbourhood according to their importance over a set of candidates, and thus, does not suffer from oversmoothing. This preserves detail and provides sharper results while maintaining local image consistency.

4. Conclusions

In this paper, we propose an inpainting method that preserves image detail, textures and sharp edged in the image by extending the exemplar-based method to a local consistency setting. We combine the local edge prior with the similarity of the candidate inpainting exemplars to the inpainting region boundary over a local neighborhood. The similarity computation generates weights for each candidate patch, which in turn contributes to the inpainting sequence decision through a list of iterates. The experimental results show the merits of the method as compared to alternatives elsewhere in the literature.

References