Modified Exemplar-Based Image Inpainting via Primal-Dual Optimization

Veepin Kumar(✉), Jayanta Mukhopadhyay, and Shyamal Kumar Das Mandal

Indian Institute of Technology Kharagpur, Kharagpur 721302, West Bengal, India
{veepinkmr,sdasmandal}@cet.iitkgp.ernet.in, jay@cse.iitkgp.ernet.in
http://www.iitkgp.ac.in/

Abstract. In this paper we present a modified exemplar based image inpainting technique to remove objects from digital images. Traditional exemplar based image inpainting techniques do not take into account similarity among patches to be filled with neighbors inside the hole. This gives visually incoherent results. To correct this problem we formulate image inpainting as a global energy optimization problem. We use primal-dual schema of linear programming for optimization. We also modify the criteria for determining priority among candidate patches to be inpainted by introducing one *'edge length'* term which propagates linear structures better than the existing techniques. Results show the effectiveness of our method compared to other recent methods.

Keywords: Image restoration · Inpainting · Exemplar · Linear programming · Metric labeling

1 Introduction

Inpainting is the process of filling damaged portions of an image, or removing any portion of image and filling it such that it looks like an original image. Various applications of image inpaintng include photograph restoration, occlusion removal, image enhancement, etc. Inpainting methods developed so far can be broadly classified into structure based and texture based methods.

Structure based techniques are based on variational methods and solving a set of partial differential equations [1,2]. They are good for inpainting non textured and smaller regions. They interpolate the geometric structure of an image (e.g. level lines, edges, etc.) in the region to be inpainted. They are local in nature as they use information available only at the boundary between a known and an unknown region. However, they introduce some blur in the inpainted region.

Exemplar based methods give relatively good results for large target regions. But, these methods fill the hole by finding most similar patches from the rest of the image iteratively [3,4]. They give good results for texture or repetitive patterns. They are non-local as they search whole image to find the best exemplar. These methods may fail to synthesize a geometry, if there are no examples of it in the image.

M. Kryszkiewicz et al. (Eds.): PReMI 2015, LNCS 9124, pp. 116–125, 2015.
DOI: 10.1007/978-3-319-19941-2_12

Most of the exemplar based methods are greedy in the sense that each patch is filled only once, and after filling it is not checked again for better reconstruction. This may sometimes produce visually inconsistent results as they do not take into account consistency between neighboring patches in the inpainted region. To overcome this difficulty, some inpainting techniques [5–7] formulate the task as solving a discrete global energy optimization problem.

In this paper, we present exemplar based image inpainting as a global energy optimization problem. We use primal-dual optimization schema of the linear programming problem to achieve global optimization. Our cost function consists of one self cost and one neighbor cost term. Self cost term ensures consistency between the boundary pixels, and neighbor cost term ensures visual consistency among neighbors in the inpainting domain. To our knowledge primal-dual optimization schema has not been used previously for image inpainting. We also introduce a new parameter named *'edge length'* in determining priority for selecting candidate patches in exemplar based inpaintng [3]. This helps in propagating linear structures more effectively compared to existing techniques. Results demonstrate the effectiveness of our method.

In next section we describe our modification to the priority term of exemplar based inpainting technique [3]. In Sect. 3, we describe image inpaintng by primal-dual optimization. Typical results of the proposed inpainting algorithm are presented and discussed in Sect. 4. Conclusions are drawn in Sect. 5, highlighting future research directions that may come out of this work.

2 Modified Exemplar Based Inpainting

In exemplar based inpainting [3] we determine priority among candidate patches in the target region. The patch having maximum priority is inpainted first. Priority '$P(p)$' for each patch is given by:

$$P(p) = C(p)D(p). \tag{1}$$

where $C(p)$ is called the confidence term, and $D(p)$ is called the data term. They are given by:

$$C(p) = \frac{\sum_{q \in \psi_p \cap (I - \Omega)} C(q)}{\mid \psi_p \mid}, \quad D(p) = \frac{\mid \nabla I_p^{\perp} . n_p \mid}{\alpha}. \tag{2}$$

where $\mid \psi_p \mid$ is the area of the patch to be inpainted denoted by ψ_p as shown in Fig. 1a. α is the normalization factor, n_p is an orthogonal unit vector to the fill front and $\perp$ denotes the orthogonal operator.

We find that Eq. (1) gives equal priority to the two points 'A' and 'B' as shown in Fig. 2. Clearly, point 'B' should be given more priority in order to propagate linear structure inside the hole in a better way. To take care of this situation, we introduce *'edge length'* term defined as:

$$E(p) = \frac{\sum_{q \in \psi_p \cap (I - \Omega)} I(q)}{\mid \psi_p \mid}. \tag{3}$$

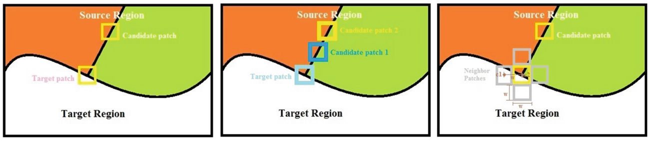

Fig. 1. (From left to right) (a) Diagram showing an image, with target region ω, its contour $\delta\omega$, source region ϕ, target patch and candidate patch. (b) Diagram showing two candidate patches for a target patch. (c) Diagram showing neighboring patches for a target patch.

where:

$$I(q) = \mid I_x(q) \mid + \mid I_y(q) \mid . \tag{4}$$

In the above equations, I_x and I_y are respectively intensity gradients in x and y directions. This *'edge length'* term gives a measure of number of pixels, which are part of an edge, and belong to known part of the candidate patch to be filled. Thus, it gives more priority to that edge whose length is more. The modified priority term is given by:

$$\tilde{P}(p) = C(p)D(p)E(p). \tag{5}$$

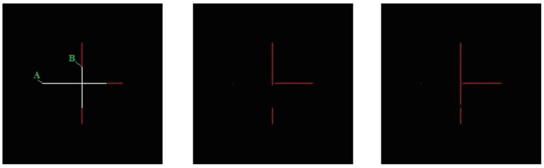

Fig. 2. (From left to right) (a) White line is to be inpainted. Both points A and B are given equal priority according to [3] but point B is given more weightage due to the $E(p)$ term in our technique. (b) Result of inpainting due to [3]. (c) Result of inpainting by our modified technique.

Figures 2a–c demonstrate the effectiveness of $E(p)$ term to generate linear structures.

3 Inpainting by Primal-Dual Optimization

We apply the modified exemplar based inpainting algorithm to the image which is to be inpainted. While filling each patch Ψ_p during modified exemplar based inpainting we find first two exemplars $\Psi_{\hat{p_1}}, \Psi_{\hat{p_2}}$ which are most similar to Ψ_p according to minimum distance criterion, as shown in Fig. 1b.

We pose the inpainting problem as a metric labeling problem [8]. Here patches in target region correspond to objects, and the best two exemplars $\Psi_{\hat{p_1}}$ and $\Psi_{\hat{p_2}}$

correspond to its two candidate labels. We further consider the integer programming formulation of the metric labeling problem introduced in [8]. For our inpainting problem it becomes:

$$min(\sum_{\psi_p \in V, \psi_1 \in L} c_{\psi_p,\psi_1} \mu_{\psi_p,\psi_1} + \sum_{(\psi_p,\psi_q)\in E} \sum_{\psi_1,\psi_2 \in L} d_{\psi_1\psi_2} \mu_{\psi_p\psi_q,\psi_1\psi_2}). \tag{6}$$

$$s.t. \sum_{\psi_1} \mu_{\psi_p,\psi_1} = 1 \qquad \forall \psi_p \in V. \tag{7}$$

$$\sum_{\psi_1} \mu_{\psi_p,\psi_1} = 1 \qquad \forall \psi_2 \in L, (\psi_p,\psi_q) \in E. \tag{8}$$

$$\sum_{\psi_2} \mu_{\psi_p\psi_q,\psi_1\psi_2} = \mu_{\psi_p,\psi_1} \qquad \forall \psi_1 \in L, (\psi_p,\psi_q) \in E. \tag{9}$$

$$\mu_{\psi_p,\psi_1}, \mu_{\psi_p\psi_q,\psi_1\psi_2} \in \{0,1\} \quad \forall \psi_p \in V, (\psi_p,\psi_q) \in E, \{\psi_1,\psi_2\} \in L. \tag{10}$$

where, L is a set of labels containing ψ_1 and ψ_2. ψ_1 corresponds to first best matching exemplar patch and ψ_2 corresponds to second best matching exemplar patch. V is a set of vertices, and E is a set of edges of a graph (V,E). The patches to be filled inside the target region (ψ_p, ψ_q, etc.) correspond to the set of vertices or nodes. Set of edges consists of pairs of neighboring vertices. We have considered four connected neighborhood. Distance between the centers of two neighboring patches is w, where $w \times w$ is the patch size as shown in Fig. 1c. Let "$\psi_p \sim \psi_q''$ denotes either $(\psi_p,\psi_q) \in E$ or $(\psi_q,\psi_p) \in E$. μ_{ψ_p,ψ_1} is 1 when vertex ψ_p is labeled ψ_1, otherwise it is set to 0. Similarly, $\mu_{\psi_p\psi_q,\psi_1\psi_2}$ is 1 when ψ_p is labeled ψ_1 and ψ_q is labeled ψ_2, otherwise it is set to 0. c_{ψ_p,ψ_1} denotes the cost of assigning label ψ_1 to node ψ_p. It is given by the sum of squared differences of the already filled pixels of the two patches ψ_p and ψ_1. Let $d_{\psi_1\psi_2}$ denotes the neighborhood cost of assigning label ψ_1 to node ψ_p and label ψ_2 to node ψ_q. It is given by the sum of squared differences of the already filled pixels of the two patches ψ_1 and ψ_2.

Constraint expressed in Eq. (10) is relaxed to $\mu_{\psi_p,\psi_1} \geq 0$ and $\mu_{\psi_p\psi_q,\psi_1\psi_2} \geq 0$, so that the above integer program becomes a linear program. Dual problem [9] of this linear program is given below:

$$max \sum_{\psi_p} \xi_{\psi_p}$$

$$s.t. \quad \xi_{\psi_p} \leq c_{\psi_p,\psi_1} + \sum_{\psi_q:\psi_q \sim \psi_p} \xi_{\psi_p\psi_q,\psi_1} \quad \forall \psi_p \in V, \psi_1 \in L. \tag{11}$$

$$and \quad \xi_{\psi_p\psi_q,\psi_1} + \xi_{\psi_q\psi_p,\psi_2} \leq d_{\psi_1\psi_2} \quad \forall (\psi_1,\psi_2) \in L, (\psi_p,\psi_q) \in E. \tag{12}$$

Here, ξ_{ψ_p} is the dual variable for each vertex ψ_p. $\xi_{\psi_p\psi_q,\psi_1}$ and $\xi_{\psi_q\psi_p,\psi_1}$ are two dual variables for each pair of neighboring vertices (ψ_p,ψ_q) and any label ψ_1.

We define an auxiliary variable $ht^{\xi}_{\psi_p,\psi_1}$ called "height variable" for any label ψ_1 as:

$$ht^{\xi}_{\psi_p,\psi_1} \equiv C_{\psi_p,\psi_1} + \sum_{\psi_q:\psi_q \sim \psi_p} \xi_{\psi_p\psi_q,\psi_1}. \tag{13}$$

This variable gives a measure of the cost of assigning a label ψ_1 to a node ψ_p.

We use the following primal dual schema [9] to design our algorithm:

Primal-Dual Schema: *Generate a sequence of pairs of integral-primal, dual solutions* $\{\mu^k, \xi^k\}_{k=1}^t$ *until the elements* $\mu = \mu^t$ *and* $\xi = \xi^t$ *of the last pair of the sequence are both feasible, and satisfy the relaxed primal complementary slackness conditions.*

We have used "PD1 algorithm" [10] of the primal-dual schema. The relaxed primal complementary slackness conditions for our pair of primal-dual linear program are given below.

$$\xi_{\psi_p\psi_q,\psi_1} + \xi_{\psi_q\psi_p,\psi_1} = 0. \tag{14}$$

$$\xi_{\psi_p} = min_{\psi_1} ht^{\xi}_{\psi_p,\psi_1}. \tag{15}$$

$$ht^{\xi}_{\psi_p,\mu_{\psi_p}} = min_{\psi_1} ht^{\xi}_{\psi_p,\psi_1}. \tag{16}$$

The feasibility condition for our algorithm, which can be derived from dual constraint given in Eq. (12), is given below.

$$\xi_{\psi_p\psi_q,\psi_1} \leq d_{\psi_p\psi_q,\psi_1}/2 \quad \forall \psi_1 \in L, \psi_p \sim \psi_q. \tag{17}$$

where, $d_{\psi_p\psi_q,\psi_1}$ denotes the neighborhood cost of assigning first best matching exemplar patches to nodes ψ_p and ψ_q. Now, our solution has to satisfy Eqs. (14) – (17).

In the PD1 algorithm, we generate a series of primal-dual pairs of solutions, one primal-dual pair per iteration. At each iteration we make sure that Conditions expressed by Eqs. (14), (15) and (17) are automatically satisfied by the current primal-dual pair. Condition expressed by Eq. (16) requires that the height of label ψ_1 assigned to any node ψ_p must be lower than that of all other labels at that node. Let ψ_p be a node for which this condition fails i.e. suppose height of some label ψ_2 is less than that of the currently applied label ψ_1. To satisfy Eq. (16), we need to raise height of label ψ_2 up to that of label ψ_1 by increasing one of the balance variables $\{\xi_{\psi_p\psi_q,\psi_2}\}_{\psi_q:\psi_q\sim\psi_p}$ according to Eq. (13). But as we increase $\xi_{\psi_p\psi_q,\psi_2}$, neighbor variable $\xi_{\psi_q\psi_p,\psi_2}$ decreases according to Eq. (14). Thus, the height of label ψ_2 at the neighboring vertex ψ_q decreases. This may result in making height of label ψ_2 at ψ_q lower than the height of currently applied label at this node, thus violating Eq. (16). We observe that any update of the balance variables can be simulated by pushing flow through an appropriately constructed capacitated graph. The optimal update can be achieved by pushing maximum flow through that graph. The computational steps are briefly discussed below:

1. We start the iterative algorithm by assigning the first best exemplar patch $\Psi_{\hat{p_1}}$ to each patch Ψ_p in the target region.
2. We compare the height of both the labels (exemplar patches) $\Psi_{\hat{p_1}}$ and label $\Psi_{\hat{p_2}}$ for each patch Ψ_p. If height of label $\Psi_{\hat{p_2}}$ is less than that of label $\Psi_{\hat{p_1}}$ for any patch, then we need to rearrange label heights such that the label we

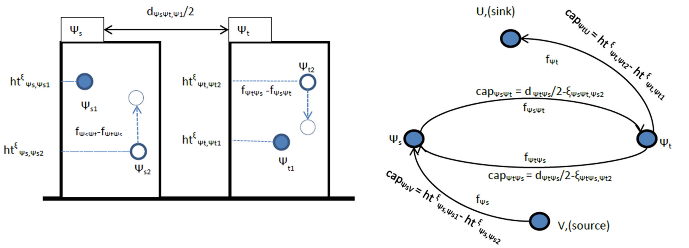

Fig. 3. (From left to right) (a) An arrangement of labels (represented by circles) for a graph G with 2 vertices Ψ_s, Ψ_t and an edge $\Psi_s\Psi_t$. The labels are $\Psi_{\hat{p1}}$ and $\Psi_{\hat{p2}}$. The dashed arrows show how the labels will move after adding flows calculated by max-flow algorithm and dashed circle indicate final position of those labels. (b) Shows the corresponding graph that will be used for the update of the dual variables.

assign to a node should have minimum height at that node. This is to obtain better visual consistency with neighboring patches.

3. We construct a directed graph to update label's heights. The graph G is augmented by two external nodes - the source 'V' and the sink 'U' as shown in Fig. 3. Let this graph be called $G^{\mu,\xi}$. All other nodes of graph $G^{\mu,\xi}$ which are also the nodes of graph G are known as internal nodes.

Interior Edges: Corresponding to each edge $(\psi_p, \psi_q) \in G$, there are two interior edges $\psi_p\psi_q$ and $\psi_q\psi_p$ in graph $G^{\mu,\xi}$. $f_{\psi_p\psi_q}$ is the amount of flow leaving ψ_p through $\psi_p\psi_q$, and it gives increase in balance variable $\xi_{\psi_p\psi_q,\psi_2}$. Also, $f_{\psi_q\psi_p}$ is the amount of flow entering ψ_p through $\psi_q\psi_p$, and it gives the decrease in $\xi_{\psi_p\psi_q,\psi_2}$. Thus, total change in $\xi_{\psi_p\psi_q,\psi_2}$ is given by:

$$\xi'_{\psi_p\psi_q,\psi_2} = \xi_{\psi_p\psi_q,\psi_2} + f_{\psi_p\psi_q} - f_{\psi_q\psi_p}. \tag{18}$$

Capacity $cap_{\psi_p\psi_q}$ of an interior edge $\psi_p\psi_q$ represents the maximum allowed increase of the $\xi_{\psi_p\psi_q,\psi_2}$ variable. These capacities are given by:

$$cap_{\psi_p\psi_q} = cap_{\psi_q\psi_p} = 0 \;\; if \; \Psi_p = \psi_2 \; or \; \Psi_q = \psi_2. \tag{19}$$

otherwise if $\Psi_p \neq \psi_2$ and $\Psi_q \neq \psi_2$,

$$\xi_{\psi_p\psi_q,\psi_2} + cap_{\psi_p\psi_q} = d_{\psi_p\psi_q,\psi_1}/2 = \xi_{\psi_q\psi_p,\psi_2} + cap_{\psi_q\psi_p}. \tag{20}$$

Exterior Edges: Each external node is connected to either the sink (If $ht^{\xi}_{\psi_p,\Psi_{\hat{p_2}}} > ht^{\xi}_{\psi_p,\Psi_{\hat{p_1}}}$) or source node (If $ht^{\xi}_{\psi_p,\Psi_{\hat{p_2}}} \leq ht^{\xi}_{\psi_p,\Psi_{\hat{p_1}}}$) through an external edge. Capacities of external edges depend on following three cases.

Case 1: $ht^{\xi}_{\psi_p,\Psi_{\hat{p_2}}} < ht^{\xi}_{\psi_p,\Psi_{\hat{p_1}}}$: The flow f_{ψ_p} passing through this edge represents the total increase in height of label $\Psi_{\hat{p_2}}$:

$$ht^{\xi\prime}_{\psi_p,\Psi_{\hat{p_2}}} = ht^{\xi}_{\psi_p,\Psi_{\hat{p_2}}} + f_{\psi_p}. \tag{21}$$

where:

$$f_{\psi_p} = f_{\psi_p \psi_q} - f_{\psi_q \psi_p}. \tag{22}$$

Capacity of $\psi_s \psi_p$ is determined by the fact that we want to increase the height of $\Psi_{\hat{p_2}}$ only upto the height of $\Psi_{\hat{p_1}}$, so:

$$cap_{\psi_s \psi_p} = ht^{\xi}_{\psi_p, \Psi_{\hat{p_1}}} - ht^{\xi}_{\psi_p, \Psi_{\hat{p_2}}}. \tag{23}$$

Case 2: $ht^{\xi}_{\psi_p, \Psi_{\hat{p_2}}} > ht^{\xi}_{\psi_p, \Psi_{\hat{p_1}}}$. The flow f_{ψ_p} passing through this edge represents the total decrease in height of label $\Psi_{\hat{p_2}}$:

$$ht^{\xi\prime}_{\psi_p, \Psi_{\hat{p_2}}} = ht^{\xi}_{\psi_p, \Psi_{\hat{p_2}}} - f_{\psi_p}. \tag{24}$$

Capacity of $\psi_p \psi_t$ is determined by the fact that the maximum decrease in height of $\Psi_{p\hat{\imath}_2}$ can be upto the height of $\Psi_{p\hat{\imath}_1}$, so:

$$cap_{\psi_s \psi_p} = ht^{\xi}_{\psi_p, \Psi_{\hat{p_2}}} - ht^{\xi}_{\psi_p, \Psi_{\hat{p_1}}}. \tag{25}$$

Case 3: $ht^{\xi}_{\psi_p, \Psi_{\hat{p_2}}} = ht^{\xi}_{\psi_p, \Psi_{\hat{p_1}}}$. Here, we want to keep the height of $\Psi_{\hat{p_2}}$ fixed. So, $f_{\psi_p} = 0$. By convention we set capacity of the edge $\psi_s \psi_p$ $cap_{\psi_s \psi_p}$ equal to 1.

After construction of the graph, a maximum flow algorithm [11] is applied to it to get flows through the edges. These flows are used to update the height of label $\Psi_{\hat{p_2}}$ as:

$$ht^{\xi\prime}_{\psi_p, \Psi_{\hat{p_2}}} = ht^{\xi}_{\psi_p, \Psi_{\hat{p_2}}} + f_{\psi_p \psi_q} - f_{\psi_q \psi_p}. \tag{26}$$

4. Based on the resulting heights we update the primal variables by assigning new labels to the vertices of G. If the new height of label $\Psi_{\hat{p_2}}$ is greater than that of label $\Psi_{\hat{p_1}}$, we assign label $\Psi_{\hat{p_2}}$ as the new label of node ψ_p (because the active label at ψ_p should be the lowest at ψ_p, (refer to Eq. (16)). This means that we have filled the current patch of the target region with the second best exemplar patch. In order to maintain visual consistency among filled patches, patches with filling priority (refer to Eq. (5)) greater than the patch ψ_p are assigned label ψ_2 i.e. are filled with second best exemplar patch.
5. Now the priorities of patches corresponding to boundary pixels of the target region which includes the patch ψ_p changes as it has been assigned label $\Psi_{\hat{p_2}}$. So, we again run the base inpainting algorithm to the remaining target region and calculate 'first two best matching exemplars for each patch to be filled'.
6. We repeat steps 1 to 5 for the remaining target region.

We keep on repeating steps 1 to 6 till the algorithm converges.

4 Results and Discussion

We applied our algorithm to remove objects from images. Number of iterations required depends upon image size, size of object to be removed, and the patch size. We have experimented with several images. A few typical results of our experimentation are presented here. We also compare the results with those obtained by the techniques reported in [3, 7]. The technique reported in [3] was implemented by us and the code for technique reported in [7] was provided by the author[1]. We took patches of size 9×9. While presenting these results we have shown outputs of modified inpainting algorithm using edge length as a factor in the computation of priority. Subsequently, the final result through primal-dual optimization algorithm is shown. It is observed that there is an improvement in the reconstruction quality in successive stages of above processing.

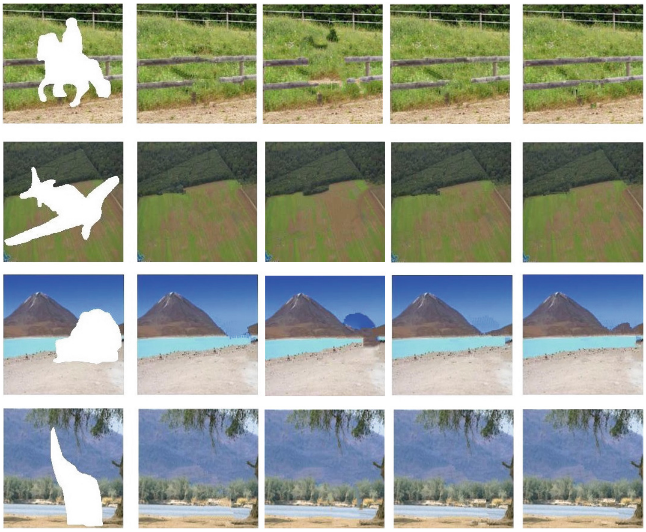

Fig. 4. (From left to right) (a) Original image with mask. (b) Result of [3]. (c) Result of [7]. (d) Result of our modified exemplar based inpainting using edge-length measure in the priority computation. (e) Result of PD1 primal dual optimization.

Fig. 5. (From left to right) Zoomed versions of inpainted regions of (b), (c), (d), (e) of first row for clarity.

[1] We are thankful to Mr. Yunqiang Liu for providing code for his paper [7].

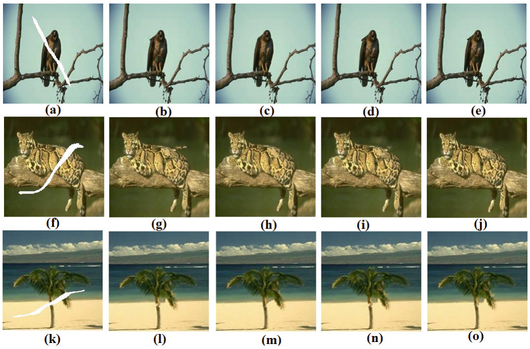

Fig. 6. Results for scratch inpainting. (a) Image with mask. (b) Criminisi's [3] (41.53 dB, 19.06 s). (c) Liu's [7] (43.25 dB, 2.09 s). (d) Modified exemplar based inpainting (41.66 dB, 21.09 s). (e) PD1 primal-dual optimization (41.75 dB, 156.87 s). (f) Image with mask. (g) Criminisi's [3] (35.43 dB, 35.43 s). (h) Liu's [7] (37.89 dB, 3.46 s). (i) Modified exemplar based inpainting (36.16 dB, 36.16 s). (j) PD1 primal-dual optimization (36.68 dB, 145.47 s). (k) Image with mask. (l) Criminisi's [3] (40.42 dB, 18.09 s). (m) Liu's [7] (43.89 dB, 2.03 s). (n) Modified exemplar based inpainting (40.42 dB, 18.61 s). (o) PD1 primal-dual optimization (41.70 dB, 74.21 s).

In the first row of Fig. 4, the techniques reported in [3,7] do not reconstruct the wooden stick, while our technique reconstructed it partially. In Fig. 5, zoomed versions of inpainted regions of this set of images are shown for better visualization. In the second row of images of Fig. 4, a spike of shadow is produced by the technique reported in [3], and a rectangular shadow block is also observed in the output obtained by the technique in [7]. Our modified exemplar based inpainting using edge-length in the priority computation reduces this effect marginally. But, the overall optimization process reduces it considerably. In the third row, there is a faulty reconstruction of a portion of river and mountain by the techniques reported in [3,7], while our technique produces a better quality of reconstruction. In the fourth row, there appears an abrupt change in boundaries between tree bushes and dirt terrain in the inpainted region by the techniques in [3,7], while the proposed technique provides a smoother transition. We applied our technique to remove scratches from images as shown in Fig. 6. Here, scratch (white region in Fig. 6) becomes the target region and rest of image becomes the source region. The same algorithm is applied as explained in Sects. 2 and 3. We computed peak signal to noise ratio (PSNR) of the reconstructed images as:

$$PSNR = 10 \times log_{10}(\frac{255 \times 255}{MSE}). \tag{27}$$

where, MSE is mean squared error. For reference image (original image I) and reconstructed image (I_{re}) of size $m \times n$ it is given by:

$$MSE = \frac{1}{3 \times m \times n} \sum_{i=0}^{2} \sum_{j=0}^{m-1} \sum_{k=0}^{n-1} [I(i,j,k) - I_{re}(i,j,k)]^2. \tag{28}$$

Our technique gave better results than technique in [3], while results of technique in [7] are better than our technique. But, our technique outperforms the technique in [7] for object removal as discussed in the beginning of this section.

5 Conclusion and Future Work

In this paper we first proposed a modification of the exemplar based image inpainting technique [3]. We have introduced a new "edge length" term in the priority equation of [3]. This propagates the linear structures in a better way. Next, we have applied the "PD1 primal dual linear programming approximation" with two labels to the image inpainting problem. In future, we would like to extend our technique for handling more labels.

References

1. Masnou, S.: Disocclusion: a variational approach using level lines. IEEE Trans. Image Process. **11**, 68–76 (1981)
2. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, pp. 417–424. ACM Press/Addison-Wesley Publishing Co. (2000)
3. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. IEEE Trans. Image Process. **13**, 1200–1212 (2004)
4. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, pp. 1033–1038. IEEE (1999)
5. Komodakis, N., Tziritas, G.: Image completion using efficient belief propagation via priority scheduling and dynamic pruning. IEEE Trans. Image Process. **16**, 2649–2661 (2007)
6. Wexler, Y., Shechtman, E., Irani, M.: Space-time completion of video. IEEE Trans. Pattern Anal. Mach. Intell. **29**, 463–476 (2007)
7. Liu, Y., Caselles, V.: Exemplar-based image inpainting using multiscale graph cuts. IEEE Trans. Image Process. **22**, 1699–1711 (2013). IEEE
8. Chekuri, C., Khanna, S., Naor, J.S., Zosin, L.: Approximation algorithms for the metric labeling problem via a new linear programming formulation. In: Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 109–118. Society for Industrial and Applied Mathematics (2001)
9. Chandra, S., Jayadeva, Mehra, A.: Numerical Optimization with Applications. Alpha Science International, United Kingdom (2009)
10. Komodakis, N., Tziritas, G.: A new framework for approximate labeling via graph cuts. In: IEEE International Conference on Computer Vision, vol. 2, pp. 109–118. IEEE Computer Society (2005)
11. Gibbons, A.: Algorithmic Graph Theory. Cambridge University Press, New York (1985)