MARKOV RANDOM FIELD BASED IMAGE INPAINTING WITH CONTEXT-AWARE LABEL SELECTION

Tijana Ružić, Aleksandra Pižurica and Wilfried Philips

Ghent University
TELIN-IPI-IBBT
Sint-Pietersnieuwstraat 41, Ghent, Belgium

ABSTRACT

In this paper, we propose a novel global Markov Random Field based image inpainting method with context-aware label selection. Context is determined based on the texture and color features in fixed image regions and is used to distinguish areas of similar content to which the search for candidate patches is limited. Furthermore, we introduce a novel optimization approach, as an alternative to priority belief propagation framework, which further reduces the number of candidates and performs efficient inference to obtain final inpainting result. Experimental results show improvement over related state-of-the-art methods. Moreover, global optimization is significantly accelerated with the proposed inference approach.

Index Terms— inpainting, patch-based algorithms, Markov Random Fields, texture descriptors, inference methods

1. INTRODUCTION

Image inpainting, or image completion, is an image processing task of filling in the missing region in an image in a visually plausible way. In literature, two categories of image inpainting approaches can be distinguished: diffusion-based [1] and patch-based [2–5]. Patch-based methods produce better results, especially when inpainting large missing regions. The missing region is filled in patch-wise manner with patches from the known region that satisfy certain fitting criterion. Filling order is crucial for the success of the algorithm because it provides both propagation of textures and object lines and borders into the missing region.

Patch-based methods can be categorized into "greedy" [2, 6], non-local [5] and global [4]. The "greedy" ones choose only one best match at a time, which may be quite limiting and cause visually inconsistent results, while non-local methods choose multiple candidate patches and the final patch represents their weighted average. Finally, global methods define inpainting as a global optimization problem. This, in addition to the choice of multiple candidates (called *labels*), allows for one label to be chosen eventually for each position so that the whole set of patches (at all positions) minimizes a global optimization function. To achieve this in an efficient manner, priority belief propagation optimization method via priority message scheduling and label pruning was proposed in [4]. Although label pruning significantly reduces the number of labels, the method is still very complex and inefficient for large images. Some solutions on how to reduce the search in a meaningful way were proposed in [3], independently of [4].

In this paper, we propose a novel global Markov Random Field (MRF) based inpainting method where contextual features are used both to improve the inpainting result and to accelerate the search for candidate patches. The main novelty is context-aware label selection, which limits the search for labels to the areas of interest based on contextual information. We employ Gabor-based texture descriptors similar to those in [7,8] and extend them with color information. While the related context descriptors were used in other domains like scene recognition [7] and scene completion using millions of photographs [8], we do not know of any works where such descriptors were used for patch-based image inpainting. We demonstrate that the inpainting process can largely benefit from such a context aware label search and selection, both in terms of speed and quality.

Another important contribution of this paper is a novel optimization approach, which builds upon our recent inference method [9] to make it suitable for global inpainting problem with large number of labels. The main advantages of this approach over related global optimization methods like [4] are improved speed, simplicity and memory efficiency. Experimental results on different images are compared to recent global and non-local methods and demonstrate potentials of the proposed inpainting method.

The paper is organized as follows. Sec. 2 reviews briefly global patch-based inpainting. The proposed method is explained in Sec. 3 and experiments and results are presented in Sec. 4. The paper is concluded with Sec. 5.

2. GLOBAL IMAGE INPAINTING

Consider the input image I, with Ω the region to be filled, called the *target* region, and Φ the known part of the image, called the *source* region. We define Markov Random Field (MRF) $G = (\nu, \varepsilon)$ over the target region Ω as a lattice of overlapping $w \times w$ patches which intersect Ω. These patches then represent MRF nodes $p \in \nu$ whose labels are all possible patches $x_p \in \Lambda$ taken from Φ, while edges ε make a four neighbourhood system around each central node. Furthermore, data cost $V_p(x_p)$ of assigning a label x_p to node p is defined as the sum of squared differences (SSD) between the known pixels at the node and the corresponding label pixels (note that it is zero if the node is completely inside Ω). Finally, pairwise potential $V_{pq}(x_p, x_q)$, where p and q are neighbouring nodes, is similarly defined as SSD between labels x_p and x_q in their region of overlap. The global inpainting problem can now be formulated as minimizing the energy

$$E(\mathbf{x}) = \sum_{p \in \nu} V_p(x_p) + \sum_{(p,q) \in \varepsilon} V_{pq}(x_p, x_q). \qquad (1)$$

This type of optimization problems can be solved using loopy belief propagation (LBP) algorithms [10]. In LBP, the solution is found by communicating *messages* between the nodes. The message in the max-product version of LBP in $-\log$ domain,

978-1-4673-2533-2/12/$26.00 ©2012 IEEE

is defined as $m_{pq}(x_q) = \min_{x_p \in \Lambda}\{V_{pq}(x_p, x_q) + V_p(x_p) + \sum_{r:r\neq q,(r,p)\in\varepsilon} m_{rp}(x_p)\}$. The belief $b_p(x_p) = -V_p(x_p) - \sum_{r:(r,p)\in\varepsilon} m_{rp}(x_p)$ represents a probability of assigning a label x_p to node p and can be interpreted as the confidence of a node about its labels.

The inpainting method of [4] introduced an improved version of belief propagation called priority BP (p-BP), to deal more efficiently with problems where each node has a huge number of labels. In particular, a specific priority message scheduling and label pruning are applied. Priority is assigned to each node as inversely proportional to the number of labels whose relative belief $b_p(x_p)$ is higher than some threshold b_{conf}. This means that the nodes with more confidence about their labels labels will have higher priority and therefore, will be visited first (in practice the ones lying on an object border and having more known pixels). The p-BP contains a forward and backward pass conducted over multiple iterations. The forward pass visits previously unvisited nodes in order of highest priority, pruning their labels, sending messages to their unvisited neighbours and updating beliefs and priorities of those neighbours. During the backward pass, nodes are visited in the reverse order and the rest of messages are sent and beliefs and priorities are updated. Important part of the algorithm is label pruning whose purpose is to reduce the number of possible labels for each node to some value $L \in [L_{min}, L_{max}]$, $L_{max} \ll |\Lambda|$, Λ is the set of all possible labels, by discarding unlikely labels, i.e. the labels whose relative belief is smaller than some threshold b_{prune}. Note that in practice, label pruning takes place only in the first forward pass. For details of the algorithm, see [4].

A problem with [4] is that, prior to label pruning, all possible labels x_p for each node p are considered. This makes the algorithm very slow because all the message and belief computations are performed for huge number of variables, especially for bigger images. We will introduce contextual information to further limit the label set making the algorithm thereby much faster.

3. PROPOSED METHOD

Our proposed method consists of two parts: (1) context-aware label selection and (2) efficient energy optimization.

3.1. Context-aware label selection

We propose to guide the candidate patch selection (i.e. label selection) by contextual information. The context is characterized by texture and color descriptors within a fixed block around each node. Texture descriptors contain a set of low-level image features that describe the texture in an image or image area. We will use similar texture descriptors as [7, 8], which are obtained by filtering the image with a bank of multi-scale oriented filters and then averaging the outputs within square non-overlapping blocks [7]. Such a representation is called a *gist* and it gives coarse description of textures in the image and their spatial organization.

We divide the image into $M \times N$ square non-overlapping blocks (see Fig. 1) and for each block B_i we compute its texture descriptor g_i as:

$$g_i(n) = \frac{1}{\#\{B_i \cap \Phi\}} \sum_{y \in B_i \cap \Phi} |I(y) \otimes h_n(y)|^2, \forall n \in \{1, \ldots, N_f\}. \quad (2)$$

$\otimes$ is a convolution operator, $\#\{B_i \cap \Phi\}$ represents the number of *known* pixels y in a corresponding block B_i and N_f is the number of filters in a chosen filter bank. We use Gabor filters of six orientations and across three scales, total of 18 filters. Then g_i is a 18-dimensional vector whose components are ordered by orientation per each scale, from high to low scales, i.e. high to low spatial frequencies.

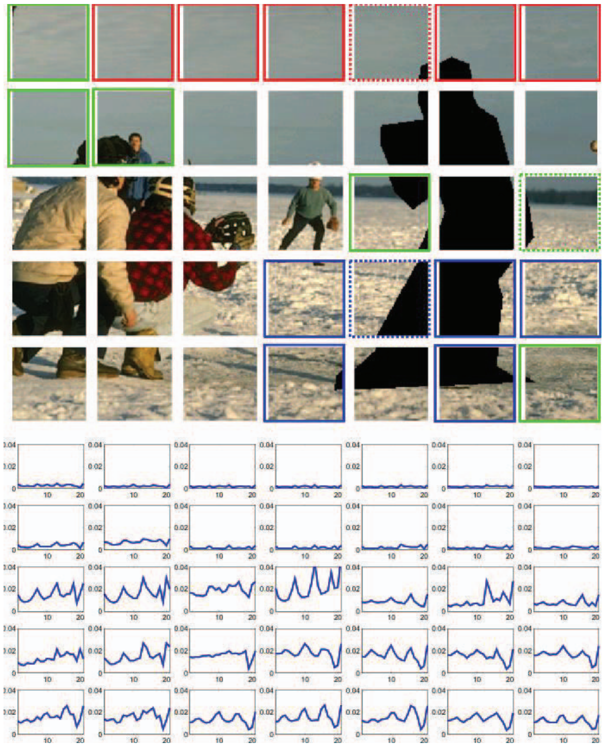

Fig. 1. Top: Division of the image into 5×7 non-overlapping blocks. Block matches of blocks in dashed squares in the top image are shown in squares of matching color. Bottom: Corresponding contextual descriptors plotted over 21 components and with values ranging between 0 and 0.04 (see text for details).

In addition to texture, it is also beneficial to include color as a feature for contextual description. Therefore, we add three more components into the feature vector g_i which represent the average color within the block per each HSV color channel:

$$g_i(N_f + n) = \frac{1}{\#\{B_i \cap \Phi\}} \sum_{y \in B_i \cap \Phi} I_n(y), \forall n \in \{1, \ldots, C\}, \quad (3)$$

where $C = 3$ is the number of color channels. The averaged color values per channel are typically higher than averaged filter responses. Hence, we normalize the color components by the factor f, $g_i(N_f + n) = g_i(N_f + n)/f$, which is the ratio between maximum value of three color components and maximum value of the averaged filter responses on first N_f components:

$$f = \frac{\max_{k \in \{N_f+1, N_f+2, N_f+C\}} g_i(k)}{\max_{l \in \{1, \ldots, N_f\}} g_i(l)}. \quad (4)$$

The resulting $(N_f + C)$-dimensional feature vector g_i $(N_f + C = 21)$ shows dominant orientations and scales within the block B_i and the average color of that block. Fig. 1 illustrates these feature vectors corresponding to different blocks of an image. We can see that

Algorithm 1: Algorithm for context-aware label selection

```
1  for $p \leftarrow 1$ to $P$ do // $P$ is the number of nodes
2      find the block $B_i$ to which $p$ belongs
3      compute block reliability $\rho_i$
4      if $\rho_i = 1$ then
5          $e(j) = \sum_{n=1}^{N_f}(g_i(n) - g_j(n))^2, \forall j = \{1, \ldots, MN\}$
6          choose $K = MN/r$ blocks $\hat{B}_i^j$ whose $g_j$ yield $K$
           smallest $e(j)$
7          define new source region $\Phi_p = \cup\{\hat{B}_i^1, \ldots \hat{B}_i^K\}$
8      else
9          $\Phi_p = \emptyset$
10         foreach neighbouring block $B_n$ do
11             repeat steps 4-6 and $\Phi_p = \cup\{\Phi_p, \hat{B}_n^1, \ldots \hat{B}_n^K\}$
12         end
13     end
14 end
```

Algorithm 2: Algorithm for efficient energy minimization

```
1  initialization:
2  for $p \leftarrow 1$ to $P$ do
3      compute $S_p(x_p) = V_p(x_p), \forall x_p \in \Lambda_p$
4      compute priority $Pr_p = 1/\#\{x_p | S_p(x_p) < T_{sim}\}$
5  end
6  label pruning:
7  for $t \leftarrow 1$ to $P$ do
8      $p$ = unvisited node of highest priority
9      apply label pruning: $x_p \in \{x_p^1, \ldots x_p^L\}, L \ll |\Lambda_p|$
10     for any unvisited neighbour $q$ of node $p$ do
11         $S_q(x_q) = S_q(x_q) + V_{pq}(x_p, x_q), \forall x_q \in \Lambda_q$
12         update priority of node $q$, $Pr_q$
13     end
14 end
15 inference method: $\hat{\mathbf{x}} = \arg\min E(\mathbf{x})$
```

texture features (the first N_f components) are small for nearly flat blocks (most of the blocks in the first two rows). For the blocks with dominant edges the peaks appear at positions corresponding to a particular orientation and tend to increase when the scale increases. Textured blocks containing snow for example, have smaller descriptor values and smaller peaks at multiple orientations.

Now we can use the feature vectors defined above to find blocks with similar content and we will limit the label set only to those blocks. The idea is to constrain the source region for node $p \in B_i$ to $\Phi_p \subset \Phi$, as shown in pseudo code in Algorithm 1. The block B_i itself is always included in this limited source region ($e(i) = 0$). For examples of block matches see marked blocks at the top of Fig. 1. Note that we also introduce binary variable

$$\rho_i = \begin{cases} 1 & \text{if } \#\{B_i \cap \Phi\} > \frac{\#B_i}{2} \\ 0 & \text{otherwise} \end{cases}$$

that represents the reliability of the block because some of the blocks that intersect the target region can have too little or even no known pixels based on which context information can be obtained, in which case we use the neighbouring information.

A couple of implementation details are described next. Note that a node can span over multiple blocks, in which case comparison is performed for all the covered blocks, and for each of them K best matches are found. Also, our contextual descriptors are computed within non-overlapping blocks, which means that the labels spanning over neighbouring blocks are not considered. Therefore, we take labels from a block extended by $w/2$.

3.2. Efficient energy minimization

Although we use contextual descriptors to limit the labels to areas of interest and, therefore, substantially reduce their number ($\Lambda_p \in \Phi_p$, $|\Lambda_p| \approx |\Lambda|/r$), we are still dealing with thousands of labels per MRF node, which is too complex for subsequent optimization. Here we introduce an efficient inference method, by extending our recent approach from [9].

We propose to first prune the labels of each node by visiting them in the order of priority, where both pruning and priority are determined based on only one term for each label of a node $S_p(x_p)$ that we call *similarity*. This allows us to define a computationally tractable MRF and perform simple and fast inference method to obtain the final inpainting result. With such an approach, we avoid computing both messages and beliefs like in p-BP, which is faster, more memory efficient and allows the application on bigger images.

A pseudo-code of this algorithm is given under Algorithm 2. We can see that similarity $S_p(x_p)$ is initially computed based on the agreement of node's labels with the known part of the patch $V_p(x_p)$ and subsequently updated with the neighbouring influence expressed with pairwise potential $V_{pq}(x_p, x_q)$, where both $V_p(x_p)$ and $V_{pq}(x_p, x_q)$ are defined in Sec. 2. Based on this similarity measure, both priorities are computed (as defined in step 4 of Algorithm 2) and label pruning is performed: L labels with highest similarity are kept as node's labels, while others are discarded.

At this point, we have a completely defined 4-connected MRF, where each node p has a set of L possible labels, where $L \ll |\Lambda_p|$, and where the potential functions are defined like in Sec. 2. The proposed way of label pruning, although resembling the first forward pass of p-BP, has the advantage of being simpler and more computationally efficient. Now we employ our recent inference method called neighbourhood-consensus message passing (NCMP) [9] to determine one label per node, where the set of labels $\hat{\mathbf{x}}$ over all nodes minimizes the energy in 1. This method is simpler and faster than belief propagation and was proved to give good results in other patch-based MRF models. Finally, chosen patches need to be stitched together in the region of overlap. As suggested in [4], we use minimum error boundary cut [11] to find the line where two neighbouring patches match best.

4. EXPERIMENTS AND RESULTS

We tested our method on a number of different natural images. The parameters of the algorithm are the following: number of labels per node $L = 10$, number of iterations of NCMP $T = 10$, number of chosen blocks $K = MN/r$, where $r = 6$, and $T_{sim} = SSD_0/2$, where SSD_0 is a predefined median value of SSDs between $w \times w$ patches. Patch size w and number of blocks $M \times N$ were varied and the optimal ones were chosen. Results on some of the images are shown on Fig. 2. For images from top to bottom, we used the same patch sizes for all global methods, $w = 15$, $w = 13$ and $w = 11$, respectively, while block divisions for the proposed method are 5×7, 3×4 and 4×5, respectively. We can see that simplified global inpainting introduced in Sec. 3.2 produces similar results as original method from [4], while the proposed method with context-aware label selection gives the best result compared by visual inspection.

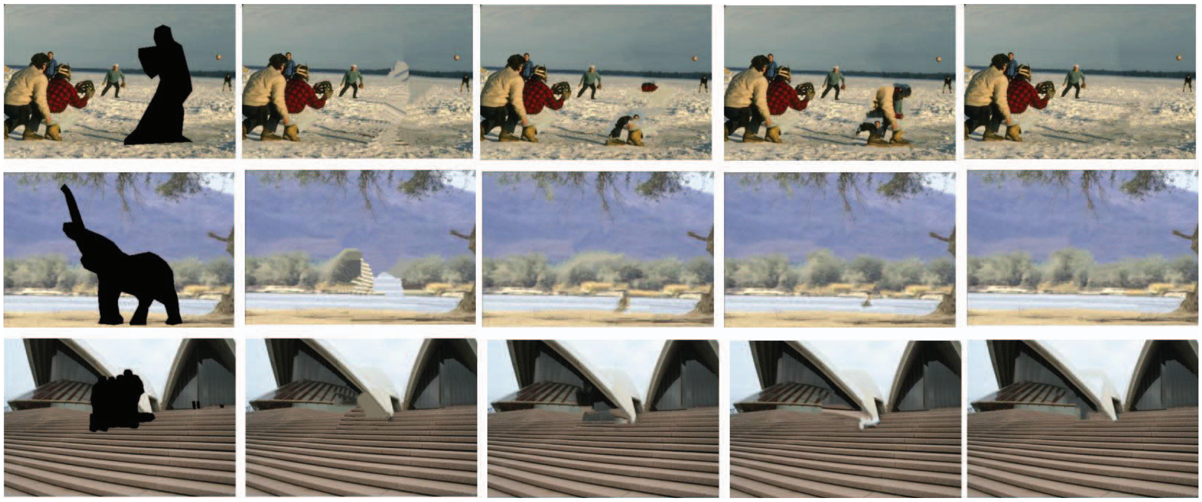

Fig. 2. Inpainting results. From left to right: image with missing region in black, results of [5], results of [4], results of the simplified optimization approach from Sec. 3.2 without context-aware label selection, results of the complete proposed method.

We also compare the results with state-of-the-art method from [5] obtained with patch size $w = 7$, three levels of hierarchy and search window of 31×31. On the tested images, the proposed method produces more accurate and more visually pleasing results. Additional results, including the dependence on block division and patch size, are available on http://telin.ugent.be/∼truzic/ICIP/.

Table 1. Comparison of computation times for different images.

Image (patch size)	Method from [4]	Proposed method
"elephant" ($w = 13$)	1566.8s	331.9s
"baseball" ($w = 15$)	1152.8s	284.3s
"sydney" ($w = 11$)	1187s	385.1s

Table 1 shows computation times of the proposed method and the global method from [4] on images from Fig. 2 using MatLab implementation on Intel i5-2520M 2.5 GHz CPU with 6GB RAM. The proposed method is obviously much faster (3 to 4 times) in all the tested cases, with different image and patch sizes and different sizes of the missing region.

5. CONCLUSION

In this paper, we introduced a novel MRF based inpainting method that uses contextual descriptors to reduce the number of possible labels per MRF node. Additionally, labels are better chosen to fit the surrounding context. We also proposed a simple and efficient way to perform optimization by first pruning the labels to some small number and then separately employing the inference method to obtain final inpainting result. Results demonstrated the benefits of such an approach in comparison with the state-of-the-art methods, both in terms of quality and speed.

6. ACKNOWLEDGEMENT

We thank to Olivier Le Meur for providing the results of [5] for comparison.

7. REFERENCES

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *SIGGRAPH (2000)*, New Orleans, USA, 2000.

[2] A. Criminisi, P. Perez, and K.Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. on Imag. Proc.*, vol. 13, no. 9, pp. 1200–1212, Sept. 2004.

[3] J. Jia and C.-K. Tang, "Image repairing: robust image synthesis by adaptive nd tensor voting," in *CVPR*, 2003, pp. 643–650.

[4] N. Komodakis and G. Tziritas, "Image completion using efficient belief propagation via priority scheduling and dynamic pruning," *IEEE Trans. on Imag. Proc.*, vol. 16, no. 11, pp. 2649–2661, Nov. 2007.

[5] O. Le Meur, J. Gautier, and C. Guillemot, "Examplar-based inpainting based on local geometry," in *ICIP*, 2011, pp. 3462–3465.

[6] T. Ružić, B. Cornelis, L. Platiša, A. Pižurica, A. Dooms, W. Philips, M. Martens, M. De Mey, and I. Daubechies, "Virtual restoration of the ghent altarpiece using crack detection and inpainting," in *ACIVS 2011*, 2011, pp. 417–428.

[7] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *Int. Jour. of Comp. Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[8] J. Hays and A. A. Efros, "Scene completion using millions of photographs," *Comm. ACM*, vol. 51, no. 10, pp. 87–94, 2008.

[9] T. Ružić, A. Pižurica, and W. Philips, "Neighbourhood-consensus message passing as a framework for generalized iterated conditional expectations," *Pat. Rec. Letters*, vol. 33, pp. 309–318, Feb 2012.

[10] J. S. Yedidia and W. T. Freeman, "On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs," *IEEE Trans. on Inf. Theory*, vol. 47, no. 2, pp. 736–744, Feb. 2001.

[11] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *SIGGRAPH 2001*, 2001, pp. 341–346.