REGION-ADAPTIVE TEXTURE-AWARE IMAGE RESIZING

Jiangyang Zhang and C.-C. Jay Kuo

Signal and Image Processing Institute, Ming Hsieh Department of Electrical Engineering
University of Southern California, Los Angeles, CA 90089-2564, USA
E-mail: jiangyaz@usc.edu and cckuo@sipi.usc.edu

ABSTRACT

In this paper, we analyze the effect of texture regularity on the performance of image resizing (or called image retargeting), and then propose an efficient texture-aware resizing algorithm. Being perceived as unimportant due to spatial homogeneity, textured patterns are largely deformed (or warped) in existing image resizing algorithms. However, arbitrary warping without considering the specific texture property tends to lead to noticeable visual artifacts. To address this issue, we exploit region features, including the scale and the shape information, to preserve both local and global structures. Guided by region and contour information, we propose a mesh-based resizing technique, which is formulated as a nonlinear least squares optimization problem and solved by an iterative Gauss-Newton method. Texture redundancy is effectively reduced through texture regularity analysis and real-time texture synthesis. The superior performance of the proposed image resizing technique is demonstrated by experimental results.

***Index Terms*—** Image resizing, region-adaptive, texture redundancy

1. INTRODUCTION

The demand to adapt images to display devices of various aspect ratios and resolutions calls for new solutions to image resizing (or called image retargeting). Traditional image resizing techniques are incapable of meeting this requirement since they may either discard important information (*e.g.,* cropping) or produce distortions by over-squeezing the content (*e.g.,* non-uniform scaling).

Recently, several techniques have been proposed for content-aware image resizing. Avidan and Shamir [1] proposed a seam-carving algorithm, which resizes an image by incrementally removing or inserting seams. Another class of image resizing methods is based on image warping. In [2], a grid mesh is placed onto the image, and resizing is formulated as computing the new mesh geometry based on a specified size. Studies show that no single retargeting operator could perform well on all images, and an algorithm combining multiple retargeting operators has been recently proposed [3]. Although this method demonstrates superior performance, its computational cost is relatively expensive for practical usage.

The above methods and their variants are mostly guided by pixel-wise significance, which is computed using either the gradient or saliency information. The saliency map implies visual attractiveness of an area while the gradient map indicates the presence of edge components. However, using an individual pixel as a basic unit may result in inaccurate characterization since both maps are weakly correlated with the underlying object structure. Specifically, with these measures, part of an object may be well preserved as it contains high contrast fine structures while the rest undergoes heavy distortion due to its homogenous surface.

Texture redundancy is another issue that remains unaddressed in previous image resizing work. Repetitive textures could be modeled as a primitive texture element that is replicated according to certain placement rules [4]. An ideal resizing solution for these textures would be reducing the total number of replications while keeping the primitive element intact. However, previous resizing schemes usually leaves the replication number unchanged and distorts the shape of primitive texture element.

More recently, Wu [5] proposed a novel image resizing technique that further explores the underlying image semantics. Symmetrical regions are resized by summarization while non-symmetrical regions are handled with traditional warping method. This method effectively addresses texture redundancy, and opens a new direction for media retargeting.

In this paper, we propose a computationally efficient image resizing algorithm that is aware of texture properties and capable of preserving underlying object structures. The image is divided into three different types of regions, each of which is treated differently. We resize salient and irregular regions by shape-preserving warping method and regular regions using fast texture-synthesis technique.

The rest of this paper is organized as follows. The impact of texture regularity on image resizing is studied in Section 2. The proposed algorithm is presented in Section 3. Experimental results are shown in Section 4. Finally, concluding remarks are given in Section 5.

2. IMPACT OF TEXTURE REGULARITY ON RESIZING

Based on the degree of randomness, most real-world textures can be roughly classified as *regular* or *stochastic*. Regular textures usually appear as periodic patterns with repeating intensity, color and shape elements. Stochastic textures, with features opposite to repetitiveness, exhibit less noticeable structures and display rather random patterns. As revealed by the experiment in [4], regularity plays a significant role as a high-level feature for human texture perception.

Previous image resizing algorithms are mostly guided by the saliency map [6], which gives higher importance to regions that present distinct properties (color, intensity or orientation) with respect to their surroundings. Textured regions, regular and stochastic, are usually perceived as unimportant due to spatial homogeneity, and largely deformed during resizing in order to keep the prominent object intact.

This treatment may not provide satisfactory resizing results. Distortion in stochastic texture may be less noticeable because of its structural randomness. However, for regular textures, structural defects caused by arbitrary warping may result in serious visual artifacts. This is due to human's innate ability to perceive symmetry and the human vision system has specialized receptive fields responding to the disruption in regularity [7]. An example is given in Fig. 1, where we compare the resizing results of three different schemes [1, 2, 8] on regular and stochastic textures.

Fig. 1. Image resizing results for *brick wall* (regular texture) and *sky* (stochastic texture) using three different schemes.

As shown in Fig. 1, we see that all three algorithms produce reasonable resizing results on *sky*, an example of stochastic texture, but they perform differently on *brick wall*, an example of regular texture. For the *brick wall* image, the patch-based texture synthesis approach [8] performs the best among all three algorithms. Seam carving [1] and scale-and-stretch [2] do not fully exploit pattern regularity in resizing so that it fails to reduce texture redundancy. The geometric defect on each brick piece induced by these two approaches leads to disruption in texture regularity, which is highly visible to human eyes. On the other hand, for the sky image, the distortion caused by [1, 2] is relatively subtle since its structure is less regular and the resultant deformation is less noticeable.

3. REGION-ADAPTIVE IMAGE RESIZING

The proposed region-adaptive resizing method consists of three main steps. First, the original image is partitioned into multiple regions and classified as *salient*, *regular* or *irregular* according to visual saliency and texture regularity. This is called the region map. Second, we resize the region map through mesh warping which incorporates both the region and the contour information. Finally, based on region features, the final result would be generated using either warping or texture synthesis. Each individual step is detailed below.

3.1. Region Map Generation

The image is segmented using mean-shift segmentation method [9], which takes three parameters as input: spatial bandwidth h_r, color bandwidth h_s and minimum pixel number per region N. To speed up, we segment the down-sampled original image and then up-sample its result back to the original size. Median filter is applied to further smoothen region boundaries.

Then, we compute individual pixel significance using the saliency measure proposed in [6]. Based on our observation, high-saliency regions computed by this measure sometimes covers only part of real prominent objects, while some areas surrounding prominent objects may be mistakenly considered as salient. Therefore, we use region saliency, computed by averaging pixel saliency within the region, to ensure uniform resizing of the underlying object.

To measure pattern regularity, we apply the scoring system based on the Gabor-filtering-based texture descriptor [10]. Each region is filtered with a set of 24 Gabor filters (including 6 orientations and 4 scales). The filtered results are projected along horizontal/vertical directions, and the normalized autocorrelation function (NAC) is computed. The periodicity of NAC could be captured by multiple projections for highly structured textures (*e.g.* brick wall, fence, etc.) while this periodicity is either very weak or does not exist at all for stochastic textures (*e.g.* sky, grass, etc.).

To generate the region map, regions with prominent saliency and texture regularity are classified as *salient* and *regular*, respectively, while the rest are labeled as *irregular*.

3.2. Mesh Warping

We cover the region map with a grid mesh denoted by $G = (\mathbf{V_q}, \mathbf{E}, \mathbf{F})$, where $\mathbf{V_q}$, $\mathbf{E}$ and $\mathbf{F}$ represent sets of quad vertices, edges and quad faces, respectively. In addition to quad vertices, we add a set of contour vertices, $\mathbf{V_c}$, to better preserve region geometry. The contour vertices are derived by tracing along the region contour and sampling contour pixels at every interval of T_s (see Fig. 2).

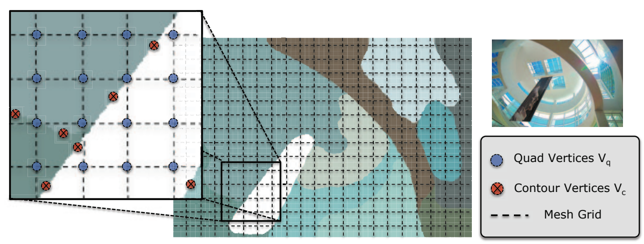


Fig. 2. Two types of vertices for mesh warping: *quad* and *contour*.

Our mesh warping algorithm takes as input the initial vertex positions and solves for the new mesh geometry. To formulate this global optimization problem, we consider the following factors.

1) Saliency-weighted shape preservation

Given quad face f and its quad vertices $V_q(f)$, we measure the shape deformation of f as its loss of squareness with the following energy:

$$D(f) = \sum_{v_i, v_j \in V_q(f)} \left\| (\mathbf{v}'_i - \mathbf{v}'_j) - s_f(\mathbf{v}_i - \mathbf{v}_j) \right\|^2,$$

where s_f is the optimum scaling factor of quad f. The shape-warping energy of all quad faces is given by

$$\sum_{f \in \mathbf{F}} w_{f(R)} D(f), \tag{1}$$

where $w_{f(R)}$ is the saliency of the region affiliated with quad f. Apparently, quad faces from the same region would be resized homogeneously while the majority of distortion tends to be diffused to non-salient regions.

2) Laplacian coordinates preservation

During the mesh warping, we preserve the region contour Laplacian coordinates by minimizing the following energy:

$$\sum_{\mathbf{v_i} \in \mathbf{V_c}} \left\| L(\mathbf{v_i}) - \mathbf{T_i}\delta_\mathbf{i} \right\|^2 , \qquad (2)$$

where $\mathbf{T_i}$ is a 2×2 transform matrix of $\mathbf{v_i}$ and will be updated iteratively during optimization.

3) Mean-value preservation

To prevent contour vertices from shifting to another quad during the mesh warping, we preserve its relative position with surrounding quad vertices by maintaining its mean value coordinates [11]. For every contour vertex $\mathbf{v_i} \in \mathbf{V_c}$, we try to minimize:

$$\sum_{\mathbf{v_i} \in \mathbf{V_c}} \left\| \mathbf{v_i} - \sum_{\mathbf{v_j} \in \mathbf{F}(\mathbf{v_i})} \lambda_{ij} \cdot \mathbf{v_i} \right\|^2 , \qquad (3)$$

where $\lambda_{ij} = m_{ij} \setminus \sum_j m_{ij}$, m_{ij} is the mean value coordinate of $\mathbf{v_j}$ with respect to $\mathbf{v_i}$, and $\mathbf{F}(\mathbf{v_i})$ is the quad face where $\mathbf{v_i}$ was originally located.

Let us use $\|\mathbf{PV} - \mathbf{Q}\|^2$ to denote the position constraint imposed by the new image size, $h_{new} \times w_{new}$. By incorporating the energy terms in Eq. (1)- (3), the total energy we want to minimize could be written in the following matrix format:

$$\underbrace{\|\mathbf{DV} - s(\mathbf{V})\|^2}_{\text{Eq.(1)}} + \underbrace{\|\mathbf{LV} - \delta(\mathbf{V})\|^2}_{\text{Eq.(2)}} + \underbrace{\|\mathbf{CV}\|^2}_{\text{Eq.(3)}} + \|\mathbf{PV} - \mathbf{Q}\|^2 .$$

Different weights may be used to balance different objectives. Then, we can express the above objective function as

$$\min_V \|\mathbf{AV} - \mathbf{b}(\mathbf{V})\|^2 , \qquad (4)$$

where

$$\mathbf{A} = \begin{pmatrix} \mathbf{D} \\ \mathbf{L} \\ \mathbf{C} \\ \mathbf{P} \end{pmatrix}, \quad \mathbf{b}(\mathbf{V}) = \begin{pmatrix} s(\mathbf{V}) \\ \delta(\mathbf{V}) \\ 0 \\ \mathbf{Q} \end{pmatrix}.$$

The nonlinear least squares optimization problem given in Eq. (4) can be solved using the iterative Gauss-Newton method. The vertex positions are initialized under the homogenous resizing condition, and they are updated iteratively via

$$\mathbf{V}^{(k)} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}(\mathbf{V}^{(k-1)}) = \mathbf{H} \cdot \mathbf{b}(\mathbf{V}^{(k-1)}), \qquad (5)$$

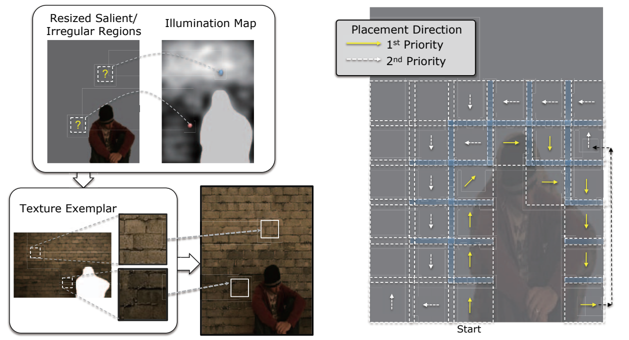
where $\mathbf{V}^{(k)}$ is the vector of vertex positions after the kth iteration. As $\mathbf{H}$ is only dependent on $\mathbf{A}$, it can be precomputed and stay fixed during the iteration.

3.3. Texture Re-synthesis

Based on the generated region map, *salient* and *irregular* regions are resized with mesh warping as discussed in Sec. 3.2. In this subsection, we focus on image synthesis in *regular* regions.

Regular regions are usually perceived as non-salient and largely squeezed during mesh warping. To preserve texture content, we re-synthesize textures on the resized *regular* region using the original texture as exemplar. For speed up, we adopt a real-time texture synthesis method proposed in [8]. Since salient and irregular regions are already known, synthesizing on regular regions is a constraint synthesis problem [8].

Directly applying texture synthesis may destroy the illumination of the original patterns. To enhance the authenticity of the re-synthesized result, we first use a nonlinear decoupling filter [12] to extract the illumination map, which is then used to guide the



(a) (b)

Fig. 3. (a) Patch selection guided by resized illumination map, (b) Priority-queue-based texture placement following spiral order.

placement of each texture patch (see Fig. 3(a)). During texture re-synthesis, a patch is selected as a qualified candidate only if it minimizes the blending error and also matches the illumination level at the desired location.

To further reduce the discontinuity artifacts at region boundaries, we use a priority queue to store the location for patch placement. Highest priority is given to areas close to salient regions, followed by those within other region border zones. Patches are placed consecutively in spiral order (Fig. 3(b)).

4. EXPERIMENTAL RESULTS

The proposed image resizing algorithm was implemented on a PC with Duo CPU 2.4 GHz. For region-map generation, values of $h_r = 5.0$, $h_s = 4.0$, $N = 800$ and $T_s = 10$ were used for all test images. We used quad size of 20×20 pixels for mesh warping, and patch size $p \times p = 90 \times 90$, overlap width $w_p = 15$ in texture re-synthesis.

We demonstrate the effectiveness of our proposed region-adaptive texture-aware resizing algorithm by comparing the results obtained by [1, 2, 3] with ours in Fig. 4. Due to the greedy nature of seam carving [1], it fails to prevent seams from passing through prominent objects when the non-salient region is relatively more structural than the prominent object (*e.g., boy*) and produces noticeable distortion on structural objects (*e.g.*, *getty* and *blueman*).

As compared with [2], the proposed region-adaptive method is better in maintaining the shape of prominent objects since mesh warping is guided by the region information. This ensures that the underlying object undergoes homogenous scaling, as in *getty* image, where the roof structure is non-uniformly warped by [2]. In particular, the proposed algorithm performs well on images containing large areas of regular textures, *e.g.*, *boy* image. As compared with [1, 2], both of which produce noticeable distortion to the background wall texture, the proposed method preserves the shape of each individual wall brick and efficiently reduces texture redundancy.

For most images, our method is able to achieve comparable results as the multi-operator approach [3]. However, [3] may sometimes crop prominent contents, as seen in the *boy* image. In addition, texture redundancy is not perfectly reduced under this scheme since cropping is the only operator among the three that effectively addresses spatial redundancy. Our approach differs from [5] in that we resize all regions with different ratios, depending on their relative region saliency, while in [5] this inter-region relationship was not considered. In addition, we reduce spatial redundancy through a fast and efficient texture synthesis technique.

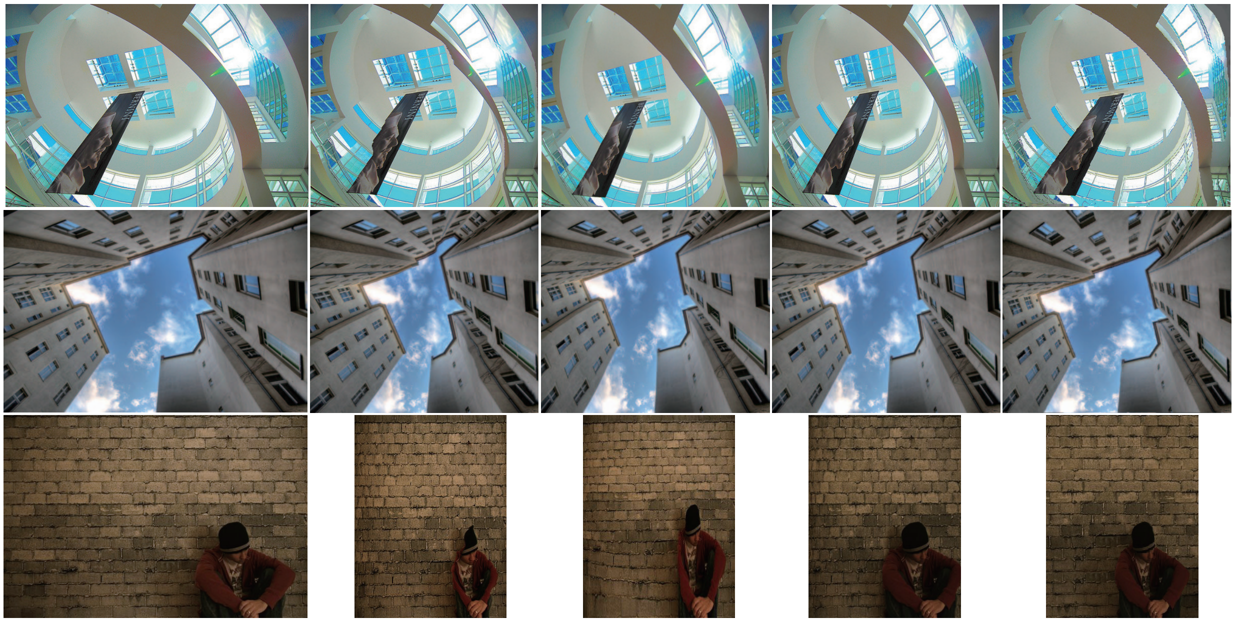


Fig. 4. Visual comparison of re-sized images using the proposed algorithm and [1, 2, 3]. From top to bottom: *getty*, *blueman* and *boy*.

Our method is computationally efficient since both region segmentation and matrix factorization can be pre-computed. In our experiments, the iterative algorithm converges within 3-10 iterations. Table 1 shows the timing statistics for the three test images. The computational time of our scheme is comparable to that of existing real-time image resizing algorithms.

Table 1. Computational time for each procedure.

	getty	*blueman*	*boy*
Vertex Number ($\|V_q\| + \|V_c\|$)	425+232	425+213	600+148
Region Map Generation	0.28 s	0.27 s	0.31 s
Matrix Factorization	0.20 s	0.21 s	0.23 s
Back Substitution	0.01 s	0.01 s	0.01 s
Texture Synthesis	-	-	0.08 s

The proposed texture-aware resizing algorithm is not without limitations. The effectiveness of this approach depends on the result of region regularity detection. When we fail to identify a regular-textured region, it will be resized by normal warping, which may lead to noticeable artifacts. Another limitation is the performance of texture synthesis. Though we try to minimize artifacts through priority-based synthesis order and illumination adjustments, the synthesis result may still be unsatisfactory for patterns with highly complex structures.

5. CONCLUSION

In this work, we proposed an efficient texture-aware image resizing algorithm using the segmented region as basic unit. Guided by region and the contour information, mesh warping was formulated as a non-linear least square optimization problem, which strives to preserve the local as well as the global object structures. Texture redundancy was effectively reduced through pattern regularity detection and real-time image synthesis. Experimental results demonstrated improved image quality over state-of-the-art image resizing algorithms.

6. REFERENCES

[1] Shai Avidan and Ariel Shamir, "Seam carving for content-aware image resizing," in *ACM SIGGRAPH 2007*, New York, 2007, p. 10.

[2] Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee, "Optimized scale-and-stretch for image resizing," in *SIGGRAPH Asia '08*, New York, NY, USA, 2008, pp. 1–8, ACM.

[3] Michael Rubinstein, Ariel Shamir, and Shai Avidan, "Multi-operator media retargeting," *ACM Trans. Graph.*, vol. 28, July 2009.

[4] A. Ravishankar Rao and Gerald L. Lohse, "Identifying high level features of texture perception," *CVGIP: Graph. Models Image Process.*, vol. 55, no. 3, pp. 218–233, 1993.

[5] H. Wu, Y.-S. Wang, Wong, Lee T.-T., T.-Y., and P.-A. Heng, "Resizing by symmetry-summarization," *ACM Trans. Graph.*, vol. 29, 2010.

[6] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254 –1259, nov. 1998.

[7] O. G. Sezer and A. Ercil, "Using perceptual relation of regularity and anisotropy in the texture with independent component model for defect detection," *Pattern Recogn.*, vol. 40, no. 1, pp. 121–133, 2007.

[8] L. Liang, C. Liu, Y.-Q. Xu, B. Guo, and H.-Y. Shum, "Real-time texture synthesis by patch-based sampling," *ACM Trans. Graph.*, 2001.

[9] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 5, pp. 603 – 619, may. 2002.

[10] P. Wu, B.S. Manjunanth, S.D. Newsam, and H.D. Shin, "A texture descriptor for image retrieval and browsing," 1999, p. 3.

[11] Michael S. Floater, "Mean value coordinates," *Computer Aided Geometric Design*, vol. 20, pp. 2003, 2003.

[12] B. M. Oh, M. Chen, J. Dorsey, and F. Durand, "Image-based modeling and photo editing," New York, NY, USA, SIGGRAPH '01, ACM.