Optimized Scale-and-Stretch for Image Resizing

Yu-Shuen Wang[1] Chiew-Lan Tai[2] Olga Sorkine[3] Tong-Yee Lee[1]
[1]National Cheng Kung University [2]Hong Kong University of Science and Technology [3]New York University

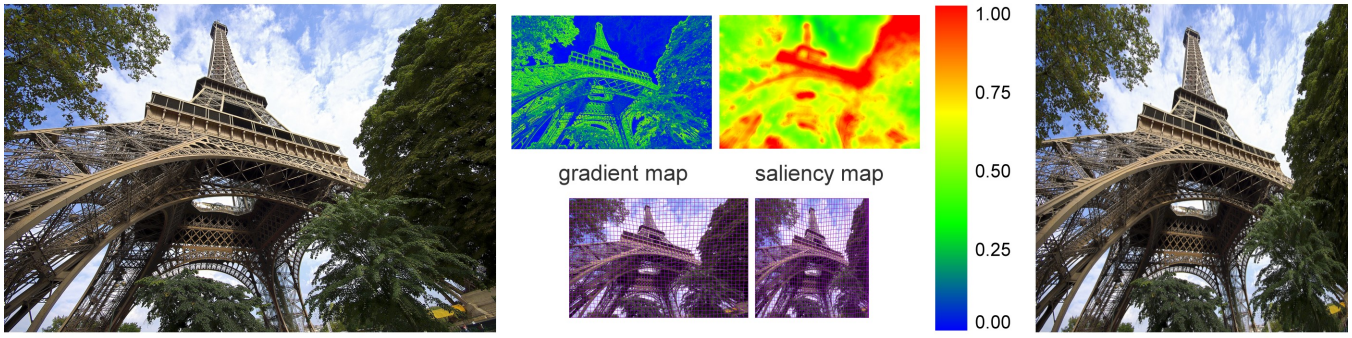


Figure 1: *We partition the original image (left) into a grid mesh and deform it to fit the new desired dimensions (right), such that the quad faces covering important image regions are optimized to scale uniformly while regions with homogeneous content are allowed to be distorted. The scaling and stretching of the image content is guided by a significance map which combines the gradient and the saliency maps.*

Abstract

We present a "scale-and-stretch" warping method that allows resizing images into arbitrary aspect ratios while preserving visually prominent features. The method operates by iteratively computing optimal local scaling factors for each local region and updating a warped image that matches these scaling factors as closely as possible. The amount of deformation of the image content is guided by a significance map that characterizes the visual attractiveness of each pixel; this significance map is computed automatically using a novel combination of gradient and salience-based measures. Our technique allows diverting the distortion due to resizing to image regions with homogeneous content, such that the impact on perceptually important features is minimized. Unlike previous approaches, our method distributes the distortion in all spatial directions, even when the resizing operation is only applied horizontally or vertically, thus fully utilizing the available homogeneous regions to absorb the distortion. We develop an efficient formulation for the nonlinear optimization involved in the warping function computation, allowing interactive image resizing.

Keywords: arbitrary image resizing, visual saliency, nonlinear optimization

1 Introduction

Research on automatic resizing of images is becoming ever more important with the proliferation of display units, such as television, notebooks, PDAs and cell phones, which all come in different aspect ratios and resolutions. Cropping images to fit the display medium inevitably discards information, and resizing images to arbitrary aspect ratios produces distortion. Any homogeneous algorithm that applies the same operator to every local region (e.g., bicubic interpolation) distributes distortion equally, the image to appear squashed or squeezed. Our goal is to design an image resizing scheme that minimizes noticeable distortion of prominent features and structural objects, such as people, vehicles or buildings.

Recently, seam carving [Avidan and Shamir 2007; Rubinstein et al. 2008] and image warping [Gal et al. 2006; Wolf et al. 2007] have been proposed to resize images non-homogeneously. Seam carving greedily removes or inserts 1D seams that pass through the less important regions in the image. Warping methods place a grid mesh onto the image and then compute a new geometry for this mesh, such that the boundaries fit the new desired image dimensions, and the quad faces covering important image regions remain intact at the expense of larger distortion to the other quads. Since humans are less sensitive to distortion of homogeneous information, such as clouds or sea, both classes of methods attempt to keep the prominent objects untouched and distort only the homogeneous regions. Unfortunately, keeping the prominent objects unchanged is certain to fail if their widths are larger than the target image width. In other words, the absence of homogeneous regions along the resizing direction would cause obvious distortion.

We present a warping method that, instead of enforcing the size of salient image regions to remain unchanged, determines an optimal scaling factor for each local region (see Figure 1). The scaling factors are iteratively optimized, and the amount of deformation to each region is guided by a significance map that characterizes the visual attractiveness of each pixel; this significance map is computed automatically using a combination of gradient- and salience-based measures. We call our strategy "optimized scale-and-stretch" since it allows regions with high importance to scale uniformly and regions with homogeneous content to be distorted. We warp the grid mesh that represents the image such that it fits the new image dimensions, and each quad's deformation matches the local scaling factor. The scaling transformations and the positions of the grid vertices are both variables in the global optimization process. We design a very efficient alternating optimization procedure that allows resizing fairly large images in real time and is easy to implement. The efficiency stems from the specially-tailored objective function formulation that reduces the nonlinear problem to a series of linear problems with a fixed system matrix. The matrix can be prefactorized, and each iterative step only requires a back-substitution.

Figure 2: *The middle columns present direct and indirect narrowing results by the forward seam carving method [Rubinstein et al. 2008] (denoted as* forward SC*) and the warping method of [Wolf et al. 2007]. Indirect narrowing achieves the desired aspect ratio by first increasing the height of the image and then downsampling. Note that structural objects, such as the car and jumping people, suffer obvious discontinuity with seam carving. Direct resizing by seam carving preserves the girl's face but not the car, and indirect resizing preserves the car but not the girl's face. This is because the homogeneous information is distributed horizontally in the* Girl *image and vertically in the* Car *image. For the more complex* Walking people *image, both previous methods cannot preserve well the aspect ratios of the people with either direct or indirect resizing strategy due to their one-directional propagation of distortion. Our algorithm achieves better results for all three images.*

The key aspect of our method is that the distortion due to image resizing is *optimally* distributed over the image, irrespective of the direction of the resizing operation (horizontal, vertical or both). This gives our technique the full freedom to utilize homogeneous image regions to hide the distortion. Moreover, our method enjoys the advantage of respecting structures within the image (such as straight lines or arches) thanks to the continuity of the warping function.

To summarize, our main contributions are: (1) An image deformation framework that optimizes the local transformations to allow uniform scaling of prominent features, thus diverting the distortion of the image content to homogeneous regions, where it is less noticeable. (2) An efficient optimization method to resize an image of size $n \times m$ to arbitrary $n' \times m'$.

2 Related work

Many algorithms have appeared in the literature for retargeting images to displays of different resolutions and aspect ratios. Traditional methods perform homogeneous resizing without considering the image content, equally propagating the distortion over the entire image and noticeably squeezing prominent objects. To achieve resizing without distortion, many approaches attempt to remove the unimportant information from the image periphery [Chen et al. 2003; Liu et al. 2003; Santella et al. 2006; Suh et al. 2003]. Based on a face detection technique [Viola and Jones 2004] and a saliency measure [Itti et al. 1998; DeCarlo and Santella 2002], the image is cropped to fit the target aspect ratio and then uniformly resized by traditional interpolation. However, cropping methods may potentially remove prominent objects close to the image boundary.

Recently proposed retargeting methods try to retain prominent objects while reducing or removing other image content. Seam carving methods [Avidan and Shamir 2007; Rubinstein et al. 2008] reduce the image size in a certain direction by removing monotonic 1D seams of pixels that run roughly in the orthogonal direction (image expansion is achieved by duplicating such seams instead). To reduce artifacts, they search for minimal-cost seams that pass through homogeneous regions by computing their forward [Rubinstein et al. 2008] or backward energy [Avidan and Shamir 2007]. These methods produce very impressive results, however their discrete nature may cause noticeable jags in structural objects. Moreover, if the homogeneous information in the required spatial direction runs out, removing seams in that direction to change the aspect ratio would rapidly lead to noticeable distortion.

Image warping [Gal et al. 2006; Wolf et al. 2007] offers a continuous solution to image resizing. The warping functions are generally obtained by a global optimization that squeezes or stretches homogeneous regions to minimize the resulting distortion. Gal et al. [2006] warp an image into various shapes, enforcing the user-specified features to undergo similarity transformations. Wolf et al. [2007] automatically determine the importance of each pixel and merge the pixels of lesser importance in the reduction direction. Gal et al. [2006] employ a simple heuristic to determine the scaling of the marked features, so that when the image is squeezed in one direction, the features are uniformly scaled by the squeezing ratio, and when stretched, the features do not scale at all.

It follows that both the carving and the warping approaches proposed so far do not utilize well the homogenous information along the direction orthogonal to the resizing direction. As shown by the

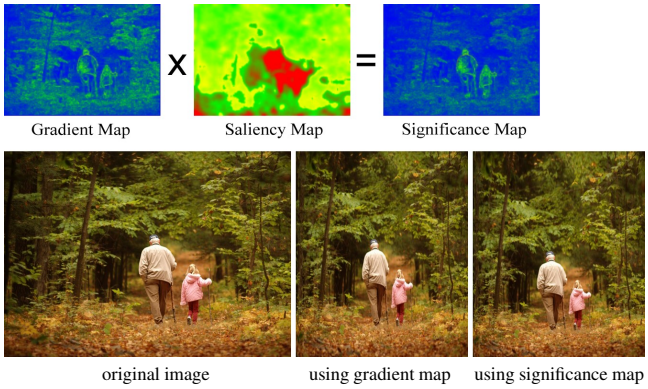


Figure 3: *We define the significance map as the product of the gradient magnitude and the saliency measure. Compared with the gradient map, the significance map is less sensitive to the disturbance of trees and leaves, focusing on the old man and the little girl. We compare the results of narrowing the original image using the gradient map and our significance map. The shapes of the old man and little girl are preserved better with our significance map.*

Car example in Figure 2, squeezing or stretching the sky and clouds clearly causes only little distortion, however reducing the image width by compressing the horizontal pixels not only warps the sky but also distorts the car. In contrast, reducing the image width indirectly by increasing its height, followed by a downsampling, can utilize the sky better to absorb distortion. Unfortunately, whether the direct or indirect resizing will achieve better results depends on the image content: the two strategies produce opposite effects for the *Car* and the *Girl* images. In contrast to the previous approaches, our algorithm finds optimal uniform scaling for prominent objects by means of a global optimization, thus implicitly allowing distortion to be distributed in all directions to fully utilize the available homogeneous areas.

3 Arbitrary image resizing

We represent an image as a mesh $\mathbf{M} = (\mathbf{V}, \mathbf{E}, \mathbf{F})$ with vertices $\mathbf{V}$, edges $\mathbf{E}$ and quad faces $\mathbf{F}$, where $\mathbf{V} = [\mathbf{v}_0^T, \mathbf{v}_1^T, ...\mathbf{v}_{end}^T]$ and $\mathbf{v}_i \in \mathbb{R}^2$ denote the initial vertex positions. The vertices and edges form horizontal and vertical grid lines partitioning the image into quads. To resize an image of $m \times n$ pixels into an arbitrary size of $m' \times n'$ pixels, we fix the position of the top-left vertex $\mathbf{v}_0$ and let the user specify a new position for the bottom-right vertex $\mathbf{v}_{end}$. The rest of the boundary vertices are constrained to slide along their respective boundaries in order to keep the image rectangular. We solve the problem of finding a deformed mesh geometry $\mathbf{V}' = [\mathbf{v}_0'^T, \mathbf{v}_1'^T, ...\mathbf{v}_{end}'^T]$, where ideally each quad undergoes a transformation that consists of uniform scaling only. Clearly, it is impossible to meet this goal for an arbitrary new image size, so some quads should inevitably deform; we spread the distortion according to the significance of each quad and obtain the new mesh geometry by a global optimization.

3.1 Quad significance

Previous image retargeting methods have used various measures to determine the significance value of a pixel automatically. Both Avidan and Shamir [2007] and Wolf et al. [2007] consider pixels with large gradient magnitudes as important. Rubinstein et al. [2008] determine the pixel significance by accumulating the discontinuity of its neighbors if the pixel is removed. We propose a new measure of quad significance that can better detect prominent image objects. We observe that only quads that are both attractive to the human eye and contain structured objects should be protected against distortion and thus should have high significance values. Hence, we combine two measures to determine the pixel significance: image gradient and saliency map [Itti et al. 1998]. To compute the saliency, Itti et al. [1998] apply various filters to extract color, intensity and orientation properties, and then look for regions that have different properties than their surroundings; this analysis is performed on multiple scales. The gradient indicates the presence of structures, while the saliency map determines the attractiveness of a region. In particular, the gradient magnitude can be misled by trivial and repeated structures, while the saliency measure also considers regions that are attractive but homogeneous as salient. By combining these two measurements, a region is considered significant only if it is both structural and attractive.

Let I denote the input image. We define the pixel significance map as $W = W_\alpha \times W_\beta$, where $W_\alpha = ((\frac{\partial}{\partial x} I)^2 + (\frac{\partial}{\partial y} I)^2)^{1/2}$ is the 2-norm of the gradient and W_β is the saliency map. The significance w_f of quad f is defined as the average pixel significance within the quad. We normalize the w_f values to obtain weighting factors within 0 and 1. Smaller values mean less importance. Figure 3 shows an example of a noisy gradient map and the significance map with the trees filtered out by the saliency measure.

3.2 Mesh-based image resizing

Given the new image size, represented by the new position for the lower right corner $\mathbf{v}_{end}$, we compute an optimally deformed mesh, such that quads of higher significance enjoy uniform scaling and quads of lower significance are allowed to be distorted more (i.e., non-uniformly squeezed or stretched). In addition, we would like to minimize the bending of the grid lines, because prominent objects usually span a set of connected quads. Specifically, the deformed mesh is solved for by optimizing the quad deformation and grid line bending energy terms subject to boundary constraints.

Quad deformation. We formulate the shape distortion energy for each quad by measuring how far the deformed quad is from a uniformly scaled version of the original quad. Ideally, for each $f \in \mathbf{F}$ there would be a scale factor s_f such that for each vertex $\mathbf{v}$ of the quad, $\mathbf{v}' = s_f \mathbf{v} + \mathbf{t}$ (where $\mathbf{t}$ is a constant translation vector). Let us denote the set of (directed) edges of f by $\mathbf{E}(f)$; the distortion energy of f is defined as

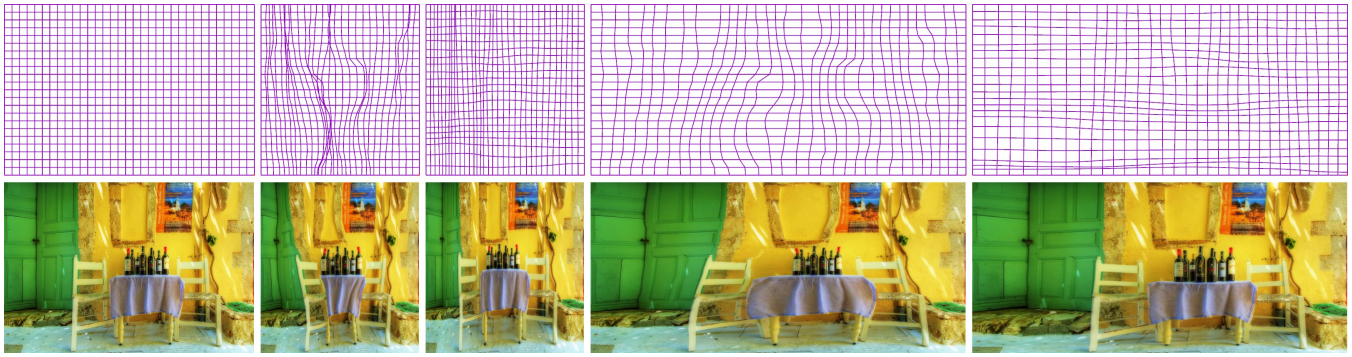
$$D_u(f) = \sum_{\{i,j\} \in \mathbf{E}(f)} \|(\mathbf{v}_i' - \mathbf{v}_j') - s_f(\mathbf{v}_i - \mathbf{v}_j)\|^2. \quad (1)$$

The translation vector $\mathbf{t}$ is eliminated by considering the edge vectors. For equal aspect ratio distortion, larger quads are penalized more, since the distortion is more visible on an enlarged area. Note that the optimal scaling factor s_f is completely defined by $\mathbf{v}_i$ and $\mathbf{v}_i'$: indeed, if the vertices of the original and deformed quads are fixed, we obtain s_f that minimizes $D_u(f)$ by differentiating Eq. (1) and equating to zero:

$$\frac{\partial D_u(f)}{\partial s_f} = \sum_{\{i,j\} \in \mathbf{E}(f)} 2\left(\|\mathbf{v}_i - \mathbf{v}_j\|^2 s_f - (\mathbf{v}_i - \mathbf{v}_j)^T(\mathbf{v}_i' - \mathbf{v}_j')\right)$$

$$\frac{\partial D_u(f)}{\partial s_f} = 0 \;\Rightarrow\; s_f = \frac{\sum_{\{i,j\} \in \mathbf{E}(f)} (\mathbf{v}_i - \mathbf{v}_j)^T(\mathbf{v}_i' - \mathbf{v}_j')}{\sum_{\{i,j\} \in \mathbf{E}(f)} \|\mathbf{v}_i - \mathbf{v}_j\|^2}. \quad (2)$$

Therefore, s_f is defined in terms of the deformed mesh vertices, and $D_u(f)$ is only dependent on those. We define the total mesh uniformity energy by summing up the individual quad energy terms and



original image [Gal et al. 2006] our result [Gal et al. 2006] our result

Figure 4: *We compare our method with [Gal et al. 2006]. The lower row shows the reduction and expansion results (guided by our significance map in all cases), and the upper row displays the respective deformed meshes. Our grid line bending energy ensures that the grid lines are smoothly bent to reduce distortion, while [Gal et al. 2006] produces C^0-continuous grid lines, distorting structural objects.*

adding the significance weights, such that more distortion would be allowed in areas of lesser significance:

$$D_u = \sum_{f \in \mathbf{F}} w_f D_u(f)\,. \tag{3}$$

Grid line bending. Since prominent objects often occupy multiple connected quads, to prevent their distortion we wish to minimize the bending of the grid lines. Specifically, the optimization system scales the edge lengths but retains the edge orientations during deformation. We define the length ratio of the edges before and after deformation as $l_{ij} = \|\mathbf{v}'_i - \mathbf{v}'_j\|/\|\mathbf{v}_i - \mathbf{v}_j\|$, and introduce the following energy term:

$$D_l = \sum_{\{i,j\} \in \mathbf{E}} \|(\mathbf{v}'_i - \mathbf{v}'_j) - l_{ij}(\mathbf{v}_i - \mathbf{v}_j)\|^2. \tag{4}$$

We illustrate the effect of this energy term by comparing with the result of [Gal et al. 2006] which does not regard grid line bending (see Figure 4). We use the same significance map in both cases. It can be observed that by preventing the grid lines from serious bending, our algorithm avoids distorting the door and chairs. In addition, this energy term also alleviates the edge flipping problem which could occur in image warping methods [Wolf et al. 2007; Gal et al. 2006]. This is because the edge ratio l_{ij} is always positive in our definition, encouraging each edge towards its original direction. Figure 5 demonstrates the effectiveness of this measure, comparing with the result of [Wolf et al. 2007]. Note that both results were obtained without using constrained systems (see details in the Appendix).

Total energy and boundary conditions. In summary, we wish to minimize the sum of the quad deformation and the line bending energies:

$$D = D_u + D_l\,, \tag{5}$$

subject to some boundary constraints. The boundary constraints are the locations of the top left and bottom right vertices

$$\mathbf{v}'_0 = (0,0)^T,\ \ \mathbf{v}'_{end} = (n', m')^T, \tag{6}$$

and in addition, the y coordinates (or x, respectively) of horizontal (vertical) boundary vertices are constrained to remain constant to make sure we get a rectangular image shape:

$$\mathbf{v}'_{i,y} = \begin{cases} 0 & \mathbf{v}_i \text{ is on the top boundary} \\ m' & \mathbf{v}_i \text{ is on the bottom boundary,} \end{cases}$$

$$\mathbf{v}'_{i,x} = \begin{cases} 0 & \mathbf{v}_i \text{ is on the left boundary} \\ n' & \mathbf{v}_i \text{ is on the right boundary.} \end{cases} \tag{7}$$

original image [Wolf et al. 2007] our result

Figure 5: *The original image is narrowed by [Wolf et al. 2007] and our algorithm. Since [Wolf et al. 2007] only constrains the vertically connected pixels to have similar displacements, self-intersection may occur. In this example, the right part of the image gets flipped, concealing some petals.*

These constraints are simply substituted into the linear system during the optimization.

We solve for the deformed mesh using an iterative solver; note that both the scaling transformations s_f and the deformed edge lengths l_{ij} are unknown, and the latter depend nonlinearly on the vertex positions. The iterative solver starts from an initial guess for $\mathbf{V}'$ (see details in the next section) and determines the quad transformations s_f and the edge ratio l_{ij} for each edge. The new vertex set $\mathbf{V}'$ is then solved for by minimizing the total energy (5) subject to constraints (6) and (7). Note that with s_f and l_{ij} fixed, the energy is a quadratic function of $\mathbf{V}'$, thus the minimization problem is linear. Furthermore, when keeping s_f fixed, the x and y coordinates of the grid vertices are not coupled and can be solved separately with the same system matrix (and different right-hand sides), reusing the matrix factorization. The alternating steps continue until all the vertex movements are smaller than 0.5.

In our experiments we have observed that the scaling transformations s_f of neighboring prominent quads should be similar because the prominent objects usually span several quads. We therefore propose to reduce the difference of adjacent scaling factors after they are determined by (2), i.e., smooth the s_f's in each iteration. Denote by $\mathbf{N}(f)$ the adjacent quads of f and by w_g the average of all the quad significance values; we obtain the smooth scaling factors s'_f by minimizing the following energy each time:

$$\sum_{f \in \mathbf{F}} \sum_{q \in \mathbf{N}(f)} \frac{1}{2}(w_f + w_q)(s'_f - s'_q)^2 + \sum_{f \in \mathbf{F}} w_g (s'_f - s_f)^2. \tag{8}$$

Figure 6 shows a resizing example demonstrating the effect of this smoothing process.

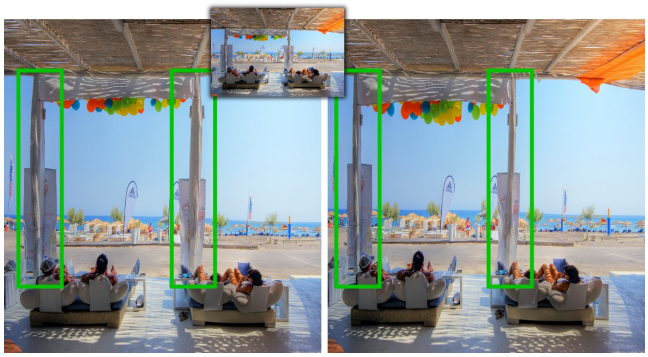


Figure 6: *The original image (inset) is expanded in height. The sky region is stretched to reduce distortion, causing the quads covering various regions of the pillars to be scaled to different sizes (left). We solve an optimization problem to make the sizes of connected prominent quads similar; this keeps the uniform thickness of the pillars (right).*

To make sure the minimization of the non-negative objective function D eventually converges, we determine the value of D after each new vertex set $\mathbf{V}'$ is obtained and verify that $D^{t+1} \leq D^t$, where t is the iteration number. Specifically, we repeatedly update the vertex set $\mathbf{V}^{t+1} = 0.7\,\mathbf{V}^{t+1} + 0.3\,\mathbf{V}^t$ until the new obtained D^{t+1} is smaller than D^t or all the vertex movements are smaller than 0.5. Overall, the alternating optimization approach is quite efficient, since the system matrix (the gradient matrix of D), as well as the smoothing matrix in (8), is sparse and remains fixed. Therefore, we can precompute a sparse factorization, and then each iteration requires only a back-substitution.

3.3 Initial guess

The convergence speed of nonlinear optimization depends on its starting position. One possibility is to use the original mesh as the initial guess, however this leads to slow convergence if the final deformed mesh is very different from the original mesh. It is desirable to choose a good starting point that is close to the optimal solution. We use two types of initial guesses in our experiments:

Previous frame. If a user resizes an image by continuously manipulating the handle vertex $\mathbf{v}_{end}$, we take the previous frame as an initial guess. Since the deformation is continuous and the sizes of two consecutive frames are similar, their optimal solutions are expected to be close to each other. We used this type of initial guess for all the examples in this paper except Figure 7, and found that the number of iterations is typically less than 10.

Homogenous resizing. If an image is resized directly by specifying the new dimensions, a better initial guess may be obtained by simple homogeneous resizing $(x, y) \rightarrow (ax, by)$. Figure 7 illustrates such an initial guess and the iterative refinement process.

3.4 Significance-aware initial mesh

To achieve interactive resizing, we represent the image by a coarser quad mesh, and create the resized image by linearly interpolating the interior content after the mesh is nonlinearly deformed. The quad mesh can be considered as a subspace of the given image. To reduce the linearization artifacts, it makes sense to place the mesh vertices more densely in the salient regions, so as to approximate the nonlinear deformation better there. Using an adaptive mesh might is one option; however this complicates the mesh structure

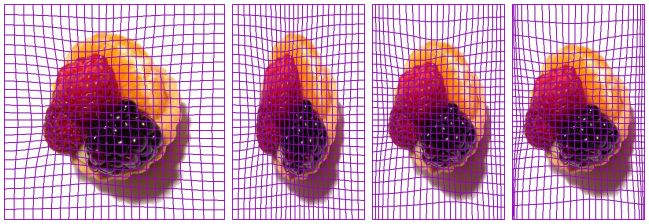


Figure 7: *The original image is resized starting from an initial guess obtained by homogeneous resizing, and then iteratively refined by our optimization algorithm such that the distortion concentrates in the homogeneous areas.*

and makes the implementation more difficult as well. To enjoy the regular structure, we propose to adapt the initial uniform grid mesh $\mathbf{M}$ to the image significance map by slightly deforming the shapes and sizes of the quads while keeping the mesh connectivity intact (see Figure 8).

The basic idea is to attract more vertices to the important regions. Specifically, we shorten an edge $\{i, j\}$ if its nearby quads cover some prominent objects. To deform the mesh, we minimize the following objective functional:

$$\Omega = \sum_{\{i,j\} \in \mathbf{E}} (1 + w_{ij}) \|\mathbf{v}'_i - \mathbf{v}'_j\|^2, \tag{9}$$

where w_{ij} is a weighting factor determined by averaging the significance of the quads sharing the edge $\{i, j\}$. The constant value 1 is added to reduce the proportion of this weighting factor. The optimization is constrained by the boundary vertices in the same fashion as in Section 3.2. Since the entire mesh is constrained to remain rectangular, the edge contraction process actually redistributes edge lengths according to their weighting factors. The edges of higher significance will get shorter, while those of smaller significance will get longer. We solve for the mesh geometry iteratively, updating the vertex positions by minimizing (9) in each step and recomputing the weights, as the quad significance changes during the deformation. We stop when all the vertex movements are smaller than 0.5 pixel.

4 Results and discussion

We have implemented our image resizing system on a PC with Duo CPU 2.33GHz. The technique is very efficient even though the solver iteratively updates the vertex positions until the process converges. This is because the matrix of our least-squares system is fixed and its factorization can be pre-computed. Therefore, each step only needs a back substitution to determine the vertex positions. Obviously, the computational cost depends on the size of the mesh. A finer mesh leads to better results but slower interactive speed. In all our experiments, we found that an initial quad of 20×20 pixels produces sufficiently good results. Using such a coarse mesh is reasonable since the deformations applied to neighboring regions should be similar. Factoring a matrix of 1976 vertices for a 1024×754 image typically takes 0.034 seconds, and a back substitution to update the vertex positions takes 0.002 seconds.

We demonstrate the effectiveness of our resizing technique with several examples (see Figures 2, 4, 5, 9, 11, 12). By adding user interface to place and manipulate additional positional constraints, our system also allows the user to perform freeform deformation on general images without significantly distorting the prominent objects. We show the interactive system in the accompanying video.

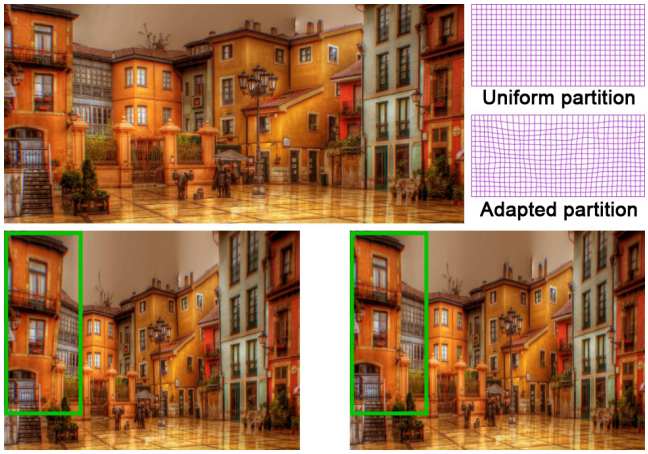


Figure 8: *Results of resizing an original image (top) using a uniformly partitioned mesh (bottom left) and a significance-aware mesh (bottom right). Notice that the building structures are preserved better with the use of the significance-aware initial mesh.*

Edge flipping, which leads to local self-intersections, may occur in warping algorithms. This causes discontinuity and may conceal prominent objects. Solving for the deformed mesh using a constrained system (see the Appendix for details) can completely eliminate this problem, with the tradeoff of increased computational cost. Fortunately, our grid line bending energy is effective in alleviating edge flipping; it rarely occurs even when the constrained system is turned off. This energy component also reduces the frequency of re-factorization when the constrained system is turned on.

Comparison. In Figure 12, we compare some results of [Wolf et al. 2007; Rubinstein et al. 2008] and our method. It can be observed that Wolf et al.'s and our methods, which are both warp-based, produce smooth results, while seam carving produces noticeable discontinuity, especially in images containing structural objects. For instance, the coral, people, San Francisco Heart and house roof are jagged since the pixels are directly removed. Comparing with the results of [Wolf et al. 2007], our method can preserve the aspect ratios of prominent objects better and avoid self-intersections (see supplemental material for more comparison results). In Figure 11, we demonstrate that seam carving guided by forward energy [Rubinstein et al. 2008] produces less discontinuities than seam carving using our significance map. This is because the forward energy considers the discontinuity of merged pixels after seam removal, while our significance map does not. Nevertheless, our significance map coupled with our optimized warping produces smoother results and preserves structural objects better than seam carving with forward energy. We believe the combination of our significance map and the forward energy can also benefit the seam carving method.

Seam carving is excellent in dealing with highly textured areas. For example, removing pixels from a sandy region would be better than compressing it. In addition, direct editing of pixels in seam carving methods leads to higher flexibility in terms of image content manipulation, enabling applications like object removal. However, the discrete nature of seam carving may damage structures because the information on the removed seam is lost. In contrast, warping methods have better ability to preserve structural objects since the applied deformations are continuous and the image components are linearly interpolated. Moreover, warping methods solve a global optimization on the entire image at once, rather than proceeding sequentially on 1D components, thus they do not suffer from greediness effects. Figure 9 demonstrates how the greedy strategy could misdirect the resizing process. Finally, the computational cost of warping methods is independent of the resizing dimensions while that of seam carving is proportional to the number of seams.

Figure 9: *Greedy resizing strategies may produce artifacts. When the image on the left is shortened by [Rubinstein et al. 2008], seams passing through the flower vase have low accumulated distortion; however, their removal destroys the vase structure. Our algorithm solves a global optimization and preserves the vase.*

Figure 10: *(left) Original image. (right) Prominent lines not parallel to the main axes may be distorted due to different degrees of squeezing of adjacent quads.*

Limitations. Our algorithm may contract a quad into a line or even a point, removing the quad content. Such direct removal may introduce discontinuities (although the removed information is the least important). It could be argued that when the significance map allows it, our algorithm performs cropping, which is actually the most reasonable in some cases.

Our algorithm may significantly stretch homogeneous quads to preserve the aspect ratios of prominent objects, causing the linear interpolation of the quad interior to show artifacts. Filling the interior using texture synthesis methods can solve this problem [Fang and Hart 2007]. For images without any homogeneous regions, our method has no region to which to concentrate the distortion and thus produces results similar to conventional (homogeneous) resizing.

Like all previous methods, our method may fail to preserve the shapes of prominent image lines of arbitrary orientations (as opposed to parallel to the main axes) due to the quads not being aligned with the feature. The adapted mesh we create is an attempt to compensate for that somewhat, but obviously, if the feature is diagonally oriented, it is very hard to adapt the mesh by our mechanism. Although our grid line bending energy term can reduce the bending distortion, when adjacent quads covering the prominent line are squeezed to different degrees, the slope of each prominent segment may change inconsistently (see Figure 10). User-defined marking of the quads covering prominent lines and increasing their significance can successfully keep prominent lines straight. However, this strategy may lead to an over-constrained system causing other regions to distort more.

original image | SC using our map | SC using forward energy | our result

Figure 11: *Comparison of our result with seam carving methods (denoted by SC) using our significance map and the forward energy. SC with forward energy does a better job of preserving the structures of the castle and bridge than SC using our map. In contrast, our warp-based method has no discontinuity problems, and preserves the aspect ratios of the castle and bridge better than SC with forward energy.*

5 Future work

In addition to the arbitrarily resizing of images, our method has potential use in video resizing. It would be necessary to consider continuity between adjacent frames, especially when there are substantial differences in their contents. Since our system preserves prominent objects according to their significance and that of the surrounding components, the aspect ratio and size of a deformed object may vary from one frame to another due to the changes of the homogeneous regions. Preservation of the aspect ratios and sizes of prominent objects requires devising a temporally-coherent significance map that considers the entire image sequence.

Acknowledgements

We thank the anonymous reviewers for their valuable comments. We are grateful to Daniel Cohen-Or for the insightful discussions, and to Ariel Shamir and Michael Rubinstein for providing the seam carving comparisons. We also thank Oscar Kin-Chung Au for the discussion on algorithms and Michael Brown for providing the video voice over. The following flickr members have kindly allowed us to use their images: stuckincustoms (car, deck, child, tomb and San Francisco heart), darwinbell (fruit), meantux (girl in front of wall), almarmon1 (buildings), millzero (surfer), wolfgangstaudt (forecourt, bower, church, Japanese house, river course and street), edbrambley (woman in room), charlesfred (walking people), crushedredpepper (old man and girl), cuellar (castle), minebeyaz (canal house), extranoise (upward sky and brick house), ikytrkr (lotus), wili (boys), tookie (jumping girls), marko_k (fish), yanivba (courtyard), shawnhenning (cat), booleansplit (Eiffel Tower), betsyjean79 (doll) and Jstar (biker). This work is supported in part by the Landmark Program of the National Cheng-Kung University Top University Project (Contract B0008), the National Science Council (Contracts NSC-95-2221-E-006-193-MY2, NSC-96-2628-E-006-200-MY3 and NSC-97-2628-E-006-125-MY3), Taiwan, Republic of China and the Research Grant Council of the Hong Kong Special Administrative Region, China (Project No: 620107).

References

AVIDAN, S., AND SHAMIR, A. 2007. Seam carving for content-aware image resizing. *ACM Trans. Graph. 26*, 3, 10.

CHEN, L. Q., XIE, X., FAN, X., MA, W. Y., ZHANG, H. J., AND ZHOU, H. Q. 2003. A visual attention model for adapting images on small displays. *ACM Multimedia Systems Journal 9*, 4, 353–364.

DECARLO, D., AND SANTELLA, A. 2002. Stylization and abstraction of photographs. *ACM Trans. Graph. 21*, 3, 769–776.

FANG, H., AND HART, J. C. 2007. Detail preserving shape deformation in image editing. *ACM Trans. Graph. 26*, 3, 12.

GAL, R., SORKINE, O., AND COHEN-OR, D. 2006. Feature-aware texturing. In *Proceedings of Eurographics Symposium on Rendering*, 297–303.

ITTI, L., KOCH, C., AND NIEBUR, E. 1998. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. Pattern Anal. Mach. Intell. 20*, 11, 1254–1259.

LIU, H., XIE, X., MA, W.-Y., AND ZHANG, H.-J. 2003. Automatic browsing of large pictures on mobile devices. In *Proceedings of ACM International Conference on Multimedia*, 148–155.

RUBINSTEIN, M., SHAMIR, A., AND AVIDAN, S. 2008. Improved seam carving for video retargeting. *ACM Trans. Graph. 27*, 3.

SANTELLA, A., AGRAWALA, M., DECARLO, D., SALESIN, D., AND COHEN, M. 2006. Gaze-based interaction for semi-automatic photo cropping. In *Proceedings of CHI*, 771–780.

SUH, B., LING, H., BEDERSON, B. B., AND JACOBS, D. W. 2003. Automatic thumbnail cropping and its effectiveness. In *Proceedings of UIST*, ACM, 95–104.

VIOLA, P., AND JONES, M. J. 2004. Robust real-time face detection. *Int. J. Comput. Vision 57*, 2, 137–154.

WOLF, L., GUTTMANN, M., AND COHEN-OR, D. 2007. Non-homogeneous content-driven video-retargeting. In *Proceedings of IEEE ICCV*, 1–6.

Appendix

To completely eliminate the mesh self-intersection problem, we can solve for the deformed mesh while checking the inequality constraints $\mathbf{e}_{ij}^T \mathbf{e}'_{ij} \geq 0$, where $\{i,j\} \in \mathbf{E}$, $\mathbf{e}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ and similarly for $\mathbf{e}'_{ij}$. We modify the objective function to be $L = D + \sum_{ij} \lambda_{ij} \|\mathbf{e}'_{ij}\|^2$, where λ_{ij} is a weighting factor determined according to the inspection of the mesh after each iteration. Initially we set $\lambda_{ij} = 0$ since all the inequality constraints are satisfied. During the search for the optimal solution, we detect conflicts with the inequality constraints after each iteration. If this happens, we increase the corresponding factor λ_{ij} to a large number (we used 10000) to enforce the flipping edge to have zero length. This changes the objective functional L, so we recompute the system matrix $\partial L / \partial \mathbf{V}'$ and its factorization.

original image | [Rubinstein et al. 2008] | [Wolf et al. 2007] | our results

Figure 12: *Comparison of our results with those of improved seam carving [Rubinstein et al. 2008] and the warping method of [Wolf et al. 2007]. The results of [Wolf et al. 2007] and our method tend to be smoother than those of seam carving. Notice the discontinuities in the corals, people, San Francisco Heart and the house roof, which are due to the pixels being directly removed. Compared with [Wolf et al. 2007], our method can preserve the aspect ratios of prominent features better.*