Optimized Image Resizing Using Seam Carving and Scaling

Weiming Dong[1]   Ning Zhou[2]   Jean-Claude Paul[3]   Xiaopeng Zhang[1]

[1]LIAMA-NLPR, CAS Institute of Automation, China   [2]Sony Corporation, Japan   [3]INRIA, France

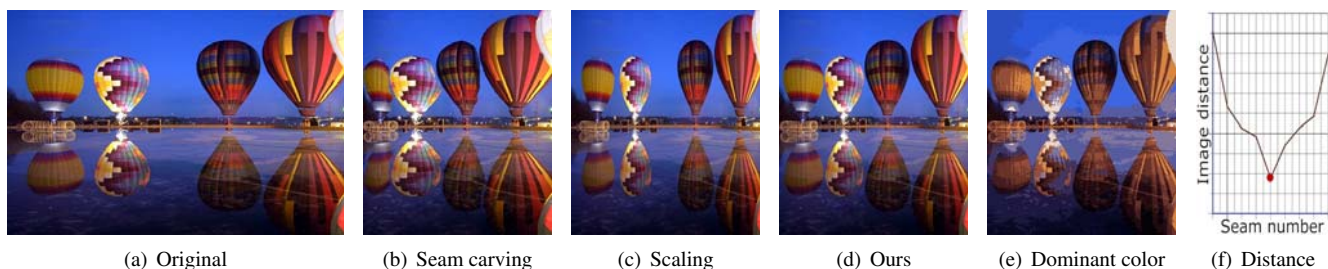| (a) Original | (b) Seam carving | (c) Scaling | (d) Ours | (e) Dominant color | (f) Distance |

Figure 1: *Our result is generated by first carving out 104 vertical seams from the original image (500×333) and then scaling to 280.*

Abstract

We present a novel method for content-aware image resizing based on optimization of a well-defined image distance function, which preserves both the important regions and the global visual effect (the background or other decorative objects) of an image. The method operates by joint use of seam carving and image scaling. The principle behind our method is the use of a bidirectional similarity function of image Euclidean distance (IMED), while cooperating with a dominant color descriptor (DCD) similarity and seam energy variation. The function is suitable for the quantitative evaluation of the resizing result and the determination of the best seam carving number. Different from the previous simplex-mode approaches, our method takes the advantages of both discrete and continuous methods. The technique is useful in image resizing for both reduction/retargeting and enlarging. We also show that this approach can be extended to indirect image resizing.

Keywords: Image resizing, Image distance function, IMED, DCD

1 Introduction

With the rapid growth of display device diversity and versatility today, new demands are made of the digital media. Adaptive resizing of images is one of the most useful techniques in relevant areas. For example, images can be changed to different sizes or aspect ratios for displaying on devices with various screen resolutions. Designers can provide different previews for photos on a website. A feasible resizing algorithm should be able to preserve the important content in an image as well as the global visual effect.

Seam carving, which can change the size of an image by gracefully carving out or inserting pixels at different locations, is an efficient technique for content-aware image resizing. A seam is constructed by searching for a connected path of pixels crossing the image from top to bottom, or left to right. Backward energy [Avidan and Shamir 2007] or forward energy [Rubinstein et al. 2008] is used to evaluate

the importance of a pixel. The main drawback is the frequently-occurred damage of local structure or global visual effect. For example in Figure 1, the boundaries of the balloons are damaged in the seam carving result. In Figure 2, the Eiffel Tower which is visually important for the image in Figure 2(a) is excessively removed in Figure 2(b). The reason is due to the energy-based strategy of the algorithm. This algorithm always removes the seams containing or inserting low energy until the desired image size is achieved, without considering the real visual effect.

Simple methods such as scaling and cropping also have clear drawbacks. Scaling the image in horizontal or vertical direction can be performed in real-time using interpolation and will preserve the global visual effects. However, scaling causes obvious distortion if the aspect ratio is different between the input and the output. The second approach is to crop the output to a window of the input image. This method will discard too much information of interest if the output resolution is significantly lower than the input resolution.

In this paper, we propose a new content-aware image resizing algorithm, which can resize an image by performing seam carving and scaling coherently. We define an image distance measure for quantifying the quality of a resizing result. The measure is useful for two purposes: (1) As an objective function within an optimization process to generate a well-resized image; (2) to quantitatively compare and evaluate resized images generated by different methods. The function is formulated as a combination of patch-based bidirectional image Euclidean distance (IMED) [Wang et al. 2005], image dominant color (DCD) [Manjunath et al. 2002] similarity and seam energy variation. Non-uniform saliency can also be incorporated into our optimization process if desired. We also compare our results with the multiple operators in [Rubinstein et al. 2009] which combines seam carving, scaling and cropping.

Our optimized content-aware image resizing algorithm starts from the seam carving operation on the original image. After each seam is removed, we directly scale the current image to the target size and compute the distance to the original image. The resized image with the minimum distance is the final result. Combining seam carving with scaling can protect the global visual effect and some local structures of the original image, especially when the output resolution is much lower than the input one. The quality of the resized image is assured by optimizing the image distance function. For image enlarging, we propose an *automatic* seam selection algorithm to replace the manual method in [Avidan and Shamir 2007]. Rather than manually specifying the number of seams to be dupli-

(a) Original image

(b) Seam carving

(c) Our result

(d) Scaling

(e) Interactive resizing
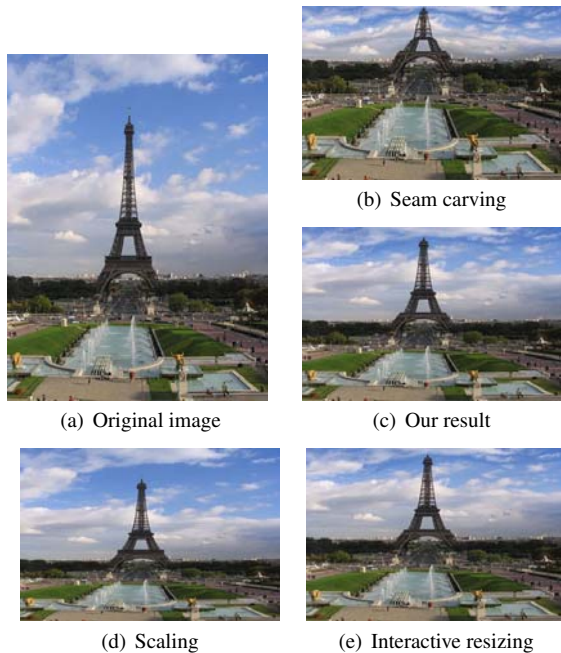
Figure 2: *Comparison of different image resizing methods. We resize the image in (a) from 375×500 to 375×250. Results show that our approach preserves the global visual effect of the original image and results a better picture than the other methods. For the image in (c), the carved out horizontal seam number is $N_{SC\_H} = 112$. In (e), we manually set $N_{SC\_H} = 125$.*

cated, our algorithm can *automatically* find the feasible seams to insert into the output image. We then show results of applying proposed distance measure for indirect image resizing.

Our distance measure function only takes a few parameters which can be fixed for most examples without visible influence to the result quality. To summarize, our main contributions are as follows:

- An image distance measure between an image and its resized variations.
- A content-aware image resizing algorithm which optimizes this measure.
- Applications of this approach in indirect image resizing.

2 Related Work

Image resizing has appeared in many applications for displaying different resolutions and aspect ratios. Traditional methods work by uniformly resizing the image to a target size without considering the image content. These methods equally propagate the distortion throughout the entire image and noticeably squeeze prominent objects. To overcome this shortcoming, many approaches attempt to remove the unimportant information from the image periphery [Chen et al. 2003; Liu et al. 2003; Suh et al. 2003; Santella et al. 2006]. Another way is to use a face detector [Viola and Jones 2004] and a saliency measure [Itti et al. 1998; DeCarlo and Santella 2002; Walthera and Koch 2006]. The image is cropped to fit the target aspect ratio and then uniformly resized by traditional interpolation. More sophisticated cropping approaches usually require human intervention to create an optimal window for the most appropriate portion of the scene. These methods work well for some special applications such as surveillance [El-Alfy et al. 2007], but

tend to fail in general photo editing applications. Besides, prominent objects may be removed by cropping methods especially when the output resolution is significantly lower than the input resolution.

Recently, seam carving methods (denoted by SC in following sections) have been proposed to retain important contents while reducing or removing other image areas [Avidan and Shamir 2007; Rubinstein et al. 2008]. These techniques reduce or expand uniform regions scattered throughout the image, by removing or duplicating monotonic pixel-wide low-energy seams. SC produces impressive results, but may deform important content, especially structural objects, when the homogeneous information in the required spatial direction runs out. Moreover, the visual effect coming from global composition of the image may also be damaged in the output, because such techniques only preserve the "important" objects. As shown in Figure 2, our algorithm successfully balances the sky, the Eiffel Tower and the garden in the result.

When one operator does not perform well, it is natural to extend it to a multi-operator. Rubinstein et al. [2009] presented an image resizing algorithm to combine different operators in an optimal manner. Bi-cubic scaling, cropping and seam carving are used together in the process. They propose Bi-Directional Warping (BDW) as a similarity measure between images to compare and evaluate the resizing results. This measure is based on a non-symmetric variant of Dynamic Time Warping (DTW).

Continuous resizing methods have been demonstrated by using image warping. To minimize the resulting distortion, the local regions are squeezed or stretched by globally optimizing warping functions. The earlier work of Gal et al. [2006] warped an image according to the user specifications, preserving the shape of masked regions. Wolf et al. [2007] automatically transformed an existing video to fit the dimensions of an arbitrary display based on local importance detectors. Zhang et al. [2008] employed shrinkability maps and random walk model to accelerate the scaling process and decrease the storage requirements. Wang et al. [2008] present a "scale-and-stretch" warping method. The method iteratively updates a warped image that matches optimal local scaling factors. However, since the distortion is distributed in all spatial directions, some objects may be excessively distorted, damaging the global spatial structure of the original image.

Patch-based methods are also presented for image retargeting or image summarization. Cho et al. [2008] chose patch arrangements that fit well together to change the size of an image. The main drawback of this method is that it cannot preserve the completeness of the image. Pritch et al. [2009] represented operations such as image retargeting, object removal, or object rearrangement as an optimal graph labeling and used graph cut to solve this problem.

3 Image Distance Function

Our approach for content-aware image resizing is to remove partial pixels with SC and homogeneously scale the others. Therefore, the question is how to choose the appropriate ratio between SC and scaling, which is controlled by the seam carving number (N_{SC_V}, N_{SC_H}). Here we design an image distance measure to evaluate the resized images using different SC numbers. The image with the minimum distance to the original image is the final result.

We address the content-aware image resizing process as an optimization problem. Given an original image I, our goal is to calculate a new image T with user-specified size, minimizing the following distance function:

$$d(I,T) = \frac{d_{IE}(I,T)}{s_{DCD}(I,T) \cdot (1.0 + s_E(I,T) \cdot \alpha)} \qquad (1)$$

where d_{IE} represents a patch-based bidirectional distance between I and T (here we use IMED), ($s_{DCD} \in [0, 1]$) is the similarity of the two dominant color descriptors (DCD), ($s_E \in [0, 1]$) is a special seam-energy based factor which is used in our algorithm to revise the distance of the resized image, ($\alpha \in [0, 1]$) is a user-specified coefficient. We will describe the distance function in detail in the following sub-sections.

3.1 Bidirectional Image Euclidean Distance

We define the following distance function:

$$d_{IE} = \overbrace{\frac{1}{N_I} \sum_{P \subset I} \min_{Q \subset T} D(P, Q)}^{d_{IE\_complete}(I,T)} + \overbrace{\frac{1}{N_T} \sum_{Q \subset T} \min_{P \subset I} D(Q, P)}^{d_{IE\_cohere}(I,T)} \quad (2)$$

Specifically, for every patch $Q \subset T$ our algorithm searches for the most similar patch $P \subset I$, and measures their distance $D(,)$, and vice-versa. The patches are taken around every pixel and their sizes are set to be (20×20) in all our experiments. Similar bidirectional distance functions are presented in [Simakov et al. 2008] and [Wei et al. 2008], the two terms $d_{IE\_complete}$ and d_{IE_cohere} both have important complementary roles. In their methods, the SSD (Sum of Squared Distances) is used as the distance measure for $D(,)$. However, sometimes SSD are not appropriate for measuring image distances, as information about spatial relationships cannot be reflected by SSD [Wang et al. 2005], especially when scaling is used to resize the image. In our algorithm, we use IMED as the measurement of patch distance.

IMED considers the spatial relationship between the pixels of different images and is therefore robust enough to withstand small perturbations. The previous IMED methods such as [Wang et al. 2005] and [Li and Lu 2009] define distance functions to compare images with fixed sizes. Here we modify this method to better compare images of arbitrary sizes and to be integrated in our image distance function. An image patch of size $n \times m$ can be written as a vector $P = \{p_1, p_2, \ldots p_{nm}\}$ according to the color value of each pixel. The IMED $D(P, Q)$ in Eq. (2) between two image patches P and Q with size $n \times m$ is then defined as:

$$D(P, Q) = \sqrt{\sum_{i=1}^{nm} \sum_{j=1}^{nm} g_{ij} (p_i - q_i)(p_j - q_j)} \quad (3)$$

where g_{ij} is the metric coefficient indicating the spatial relationship between pixels $p_i \in P$ and $q_j \in Q$. The definition of g_{ij} is given by

$$g_{ij} = f(d_{ij}^s) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(d_{ij}^s)^2}{2\sigma^2}\right)$$

where d_{ij}^s is the spatial distance between p_i and q_j on the image lattice, and σ is the width parameter ($\sigma = 1.0$ in our experiments). Using a different approach to previous methods [Wang et al. 2005; Li and Lu 2009], we divide the horizontal/vertical coordinate of a pixel by the width/height of the image so as to compare images with different sizes. Thus, if p_i is at location (k, l) and q_j is at location (k', l'), then d_{ij}^s is

$$d_{ij}^s = \sqrt{(k - k')^2 + (l - l')^2}$$

Since IMED considers the spatial relationship between pixels, it is relatively insensitive to small spatial deformations. In our current implementation, the patch distance is measured in YIQ color space and normalized by the patch size. To accelerate the computation speed, for $d_{IE\_complete}$ we do not calculate distances of

patches around every pixel. Instead we just uniformly sprinkle some overlapping patches on the original image (as illustrated in Figure 3(b), P_1, P_2 are two patches) and use the average distance of these patches as the $d_{IE\_complete}$ value.

(a) Original image   (b) Sprinkling patches

(c) SC   (d) Scaling   (e) Ours
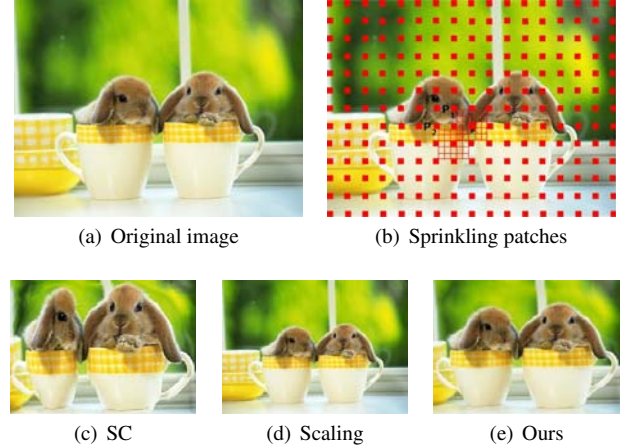
Figure 3: *IMED uses discrete pathes. The optimized seam number is* $(N_{SC\_V} = 158, N_{SC\_H} = 95)$.

3.2 Dominant Color Similarity

The similarity of the dominant color descriptors (DCD) between the original image and the resized image is also an important term in our image distance function. A set of dominant colors in a region of interest or in the whole image provides a compact description that is easy to index [Manjunath et al. 2001]. It has been widely used in content-based image retrieval. A DCD specifies a small number of dominant color values and their statistical properties: distribution and variance [Manjunath et al. 2002]. The structure of a DCD, F, is defined as

$$F = \{p_i, c_i, \mathbf{v}_i, s\}, \quad i = 1, 2, \ldots, N_{DCD} \quad (4)$$

where N_{DCD} is the number of dominant colors ($N_{DCD} = 16$ in our experiments), s is the spatial coherency value that represents the overall spatial homogeneity of the dominant colors, p_i is the percentage of pixels in the image corresponding to the ith dominant color, $\mathbf{v}_i$ is a vector representing the ith dominant color, and the c_i is the variation of the dominant color values of the pixels around $\mathbf{v}_i$. In our algorithm, s is omitted.

We adapt the Euclidean distance in [Min and Cheng 2009] to calculate the similarity of two DCDs:

$$s_{DCD} = \frac{1}{N} \sqrt{\sum_{i=1}^{N} [(p_{1i} - p_{2i})^2 + (c_{1i} - c_{2i})^2 + \|\mathbf{v}_{1i} - \mathbf{v}_{2i}\|^2]}$$

where $N = \max(N_1, N_2)$ if N_1 is not equal to N_2. Note that if $S_{DCD} = 0$, we set the image distance d to be a pre-defined maximum value. The value of c and $\mathbf{v}$ is both normalized to $[0, 1]$ so that the similarity value will also be in $[0, 1]$ domain.

Dominant color examples are shown in Figure 1(e) and Figure 4(f). We can see that our method can exactly extract the dominant colors of the images. In Figure 4, we see the importance of the DCD term. In this example, we omit the measured distance revision term ($s_E = 0.0$). We can see that the algorithm cannot obtain a good result by comparing only the IMED. DCD not only contains the color

(a) Original image   (b) SC   (c) Scaling   (d) Without DCD   (e) Ours with DCD   (f) DCD of (e)
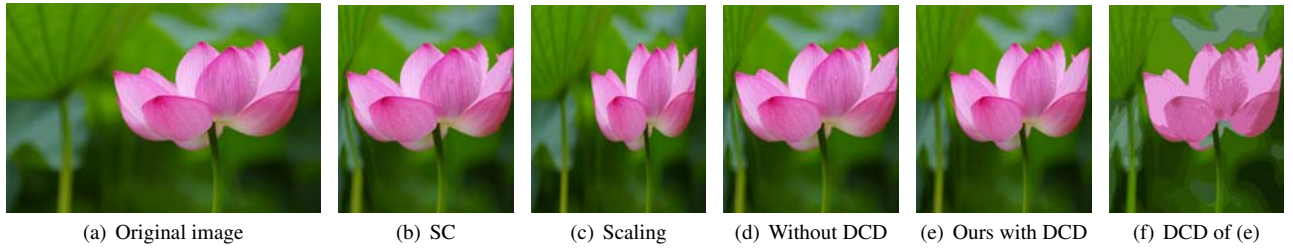
Figure 4: *The effect of DCD. The image in (d) ($N_{SC\_V} = 157, s_{DCD} = 0.77$) is generated only using IMED. We can see that the image in (e) ($N_{SC\_V} = 122, s_{DCD} = 0.82$) which integrates DCD preserves the global visual effect better. Note that we do not use s_E here.*

(a) Original image   (b) SC   (c) [Wang et al. 2008]   (d) Ours without revision   (e) Ours
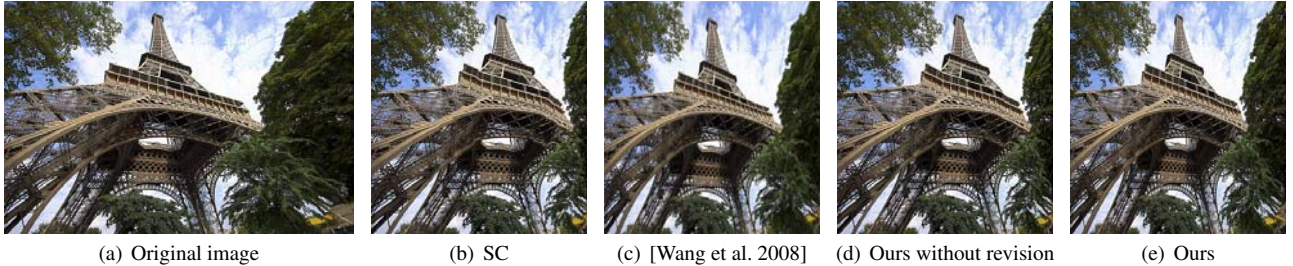
Figure 5: *Distance measure revision. Our method generates similar result ($N_{SC\_V} = 96$) as [Wang et al. 2008] for preserving the edge continuity of the Eiffel Tower, while causing less distortion to other objects. Compared with the image in (d) ($N_{SC\_V} = 116$), adding revision term better preserves the object boundaries.*

value information, but also the percentage of pixels in the image corresponding to each dominant color. Both of them will play important roles when DCD distance is calculated, which is why it can detect the fine changes that occur while gradually resizing an image. We can see that DCD helps to achieve a nice balance between the flower and the leaf in Figure 4.

3.3 Distance Measure Revision

Since we use a scaling operation to protect the global visual effect and some local structures, this will increase the patch-based bidirectional distance more quickly than using pure SC [Simakov et al. 2008]. Due to similar reason, IMED can also be affected in particular for images with strong structured parts. To solve this problem, as shown in Eq. (1) we use an additional term s_E to revise the distance. Without losing generality, suppose we narrow an image from $n \times m$ to $n' \times m$, the current carved out seam number during the optimization process is $N_{SC\_V} = t$, then the revision term is defined as

$$s_E = \frac{\sum_{i=1}^{n} E(i) - \sum_{i=1}^{t} E(i)}{\sum_{i=1}^{n} E(i)} \quad (5)$$

where $E(i)$ is the energy of the ith seam. Intuitively, this term can approximately estimate how much damage the SC process will cause to the global visual effect or the local structures if we continue to use SC to narrow the current image to the target size. s_E also describes the energy variation during the SC process. We use a small $\alpha = 0.2$ in Eq. (1) so that it acts a soft constrain to IMED and DCD similarity terms. In fact, for most examples we can just ignore this revision term. Note that it can be used when only one direction of the image is going to be resized.

4 Optimized Image Resizing

In this section, we describe our SC-Scaling-Combined optimized image resizing algorithm in detail. The image distance function in Eq. (1) is integrated into the optimization process of the algorithm.

4.1 Image Reduction/Retargeting

Image reduction/retargeting generalizes aspect ratio change in one or two dimensions so that an image I of size $n \times m$ will be changed to size $n' \times m'$, and for the time being, we assume that $n' \leqslant n$ and $m' \leqslant m$, where $n - n' = c, m - m' = r$. The goal of our algorithm is to find the feasible SC numbers in both vertical and horizontal directions, where $0 \leqslant N_{SC\_V} \leqslant c$ and $0 \leqslant N_{SC\_H} \leqslant r$). Namely, to obtain a resized image of best visual effect, we need to find out how many width/height ratios should be first reduced by SC and how many others then by scaling. The result is achieved with an optimization process by evaluating the image distance function of Eq. (1). Our image reduction/retargeting algorithm is summarized as follows:

1. Start the resizing process with SC process. After each seam removal operation, scale the image directly to the target size and calculate the image distance to the original image, using the distance function Eq. (1). Record the seam number (N_{SC_V}, N_{SC_H}) of the current step.

2. The resized image which has the minimum distance to the original image is the final result. At the same time, we also know the optimized seam number for obtaining best visual effect.

Note that we follow the same SC method presented in [Avidan and Shamir 2007] and [Rubinstein et al. 2008]. Forward energy [Rubinstein et al. 2008] is employed in all our experiments.

| (a) Original image | (b) [Avidan and Shamir 2007] ($k = 40$) | (c) Scaling | (d) Ours ($N_{SC\_V} = 37$) |
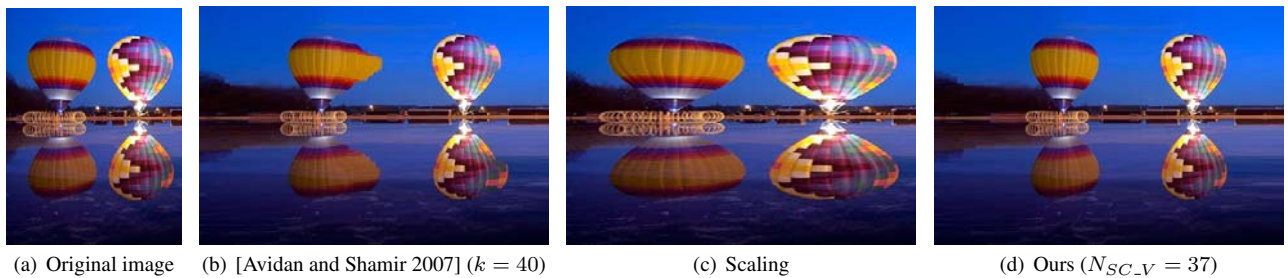
Figure 6: *Image enlarging. The original image is enlarged from 220×300 to 440×300. We first indicate $k = 40$ as the probable seam number. Then our algorithm automatically finds that the first 37 seams are the best choice to be duplicated in the original image.*

4.2 Image Enlarging

With the SC method, to enlarge an image, we need to perform an "inversion" of image reduction, and insert new "artificial" seams into the image [Avidan and Shamir 2007]. To be effective, it is important to balance between the original image content and the artificially inserted parts. Therefore, to enlarge the image by k, the SC method finds the first k seams for removal, and duplicates them in the output. Unfortunately, this approach would probably produce potential artifacts because the seams may contain non-homogeneous pixels and destroy the object shape. For excessive image enlarging (greater than 50%), the method in [Avidan and Shamir 2007] breaks the process into several steps but still cannot essentially solve the problem. In our method, we use a new approach to find the appropriate seams which can be inserted into the output image.

Without losing generality, assume we want to enlarge an image from $n \times m$ to $(n + l) \times m$. As suggested also by [Avidan and Shamir 2007], we indicate a probable seam number k at first. Here k is the forecast to the probable number of "homogeneous" seams in the image. Then we use the optimized image retargeting algorithm to resize the image to $(n - k) \times m$ and find the optimal N_{SC_V} value in the domain of $[0, k]$ using our distance measure. Finally, we only duplicate the first N_{SC_V} seams image to get the enlarged result. This method is more effective and accurate than setting the seam numbers manually (Figure 6).

5 Results and Discussions

We have implemented and tested our image resizing system on a PC with Duo CPU 2.33GHz, 2GB RAM.

Image Resizing / Retargeting. In Figure 1-4, Figure 6-9 and Figure 11, we compare our image resizing results with those of SC and scaling. It is shown that our algorithm generates better results by taking the advantages of both seam carving and scaling. Our bidirectional image distance measure is a patch-based method, for each patch in the original/resized image we calculate its distance to the patches in the resized/original image and save the minimum value. We use a new IMED rather than SSD to calculate the patch-patch distance. If SSD is used, apparently for the pure SC result, many or some of the patch-patch distances will be zero, so in most cases it will introduce a biased preference for seam carving. A key advantage of IMED is its relative insensitivity to small perturbation [Wang et al. 2005]. It is feasible here for our algorithm to evaluate the scaling-enhanced result. IMED adds global information to the patch-patch comparison and is more robust than SSD (only calculates the local pixel-pixel distance) in many cases.

Figure 10 shows results of our gradual resizing algorithm and the

| | (b) SC | (c) Scaling |
| (a) Original image | (d) Cropping | (e) Ours |
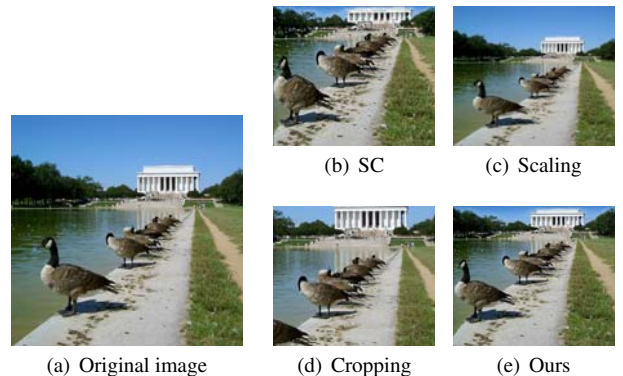
Figure 7: *Optimal order retargeting. The original image is resized from (500×500) to (300×260). For our algorithm, $(N_{SC_V} = 118,\ N_{SC_H} = 124)$.*

corresponding distance $d(I, T)$ of Eq. (1). In the first few images, $T_1, \ldots, T_5$, the loss of visual information is gradual, accompanied by a slow increase in the distance measure $d(I, T)$. Starting from T_6, there is a sharp increase in $d(I, T)$, and the resizing operation almost changes to pure scaling. This may suggest an automatic way to identify a good stopping point in the resize reduction [Simakov et al. 2008].

As shown in Figure 5, using the revision term can obtain a better result. This term partly reflects how much damage SC has caused to the image. The user could simply adjust the weight value to increase or decrease the effect of the revision term to the distance measure. However, s_E itself is not practical for optimized resizing when scaling should be used. In our algorithm, s_E is a monotonic decreasing function. If only s_E is used, the only way to stop SC is by manually setting a threshold. Apparently this approach cannot obtain an optimized result. As shown in Figure 6, using s_E with a manually set threshold cannot obtain a good result. The user has to adjust the threshold value many times to determine which result is better, while our algorithm can automatically find the optimized seam number. In fact in our further experiments, we find that using the max operator in Eq. (2) has a similar effect on the results.

In Figure 3 and Figure 7, we show that our algorithm is feasible for resizing both the width and height of the image, especially when the resizing proportions are different (Figure 7). Our method finds the best result by starting from the optimal seams-order operation [Avidan and Shamir 2007] and directly performing 2D scaling after each optimal seam (horizontal or vertical, according to the seam energy) is removed. In Figure 8, we show the efficiency of our algorithm. For the "Rope skipping" example, seam carving can result in the

(a) Original image (b) Seam carving (c) Scaling (d) Our result

Figure 8: *Comparison of our results with seam carving and scaling. For each example, from left to right, the image distances, DCD similarity and optimized seam carving numbers (denoted as* (d, s_{DCD}, N_{SC_V})*) are listed beneath. Rope skipping:* $(42.27, 0.75, -)$*,* $(57.13, 0.74, -)$*,* $(42.27, 0.75, 192)$*. Secret #5:* $(105.44, 0.80, -)$*,* $(126.13, 0.95, -)$*,* $(75.47, 0.85, 168)$*. Face:* $(158.97, 0.69, -)$*,* $(126.24, 0.75, -)$*,* $(119.51, 0.75, 53)$*.*


(a) Original image (b) Seam carving (c) Scaling (d) Our result

Figure 9: *Comparison of our results with seam carving and scaling. The objects in the rectangles are well protected in our results. For each example, from left to right, the image distances, DCD similarity and optimized seam carving numbers (denoted as* (d, s_{DCD}, N_{SC_V})*) are listed behind. Boy:* $(74.62, 0.69, -)$*,* $(78.99, 0.72, -)$*,* $(46.83, 0.79, 128)$*. Road Sign:* $(86.70, 0.66, -)$*,* $(106.06, 0.68, -)$*,* $(85.07, 0.68, 86)$*.*
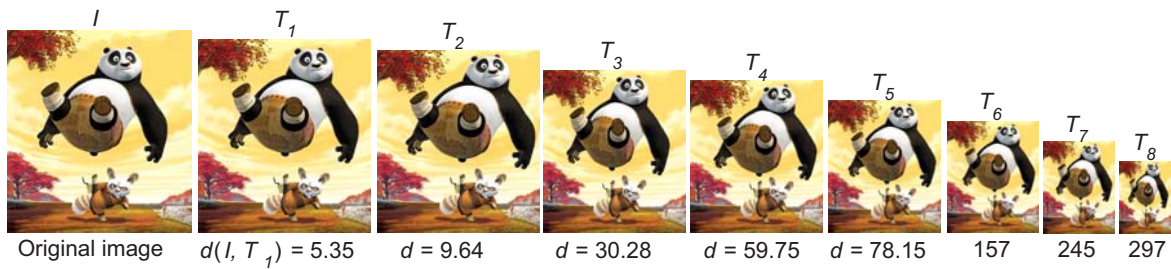
Figure 10: *Gradual image resizing. Results of our algorithm for gradually decreasing targeting size, with distance values $d(I, T)$ given by Eq. (1). Note that when the target size is very small, our algorithm can automatically choose to use more scaling than seam carving.*


(a) Original image   (b) Seam carving   (c) Warping   (d) SSD   (e) Ours
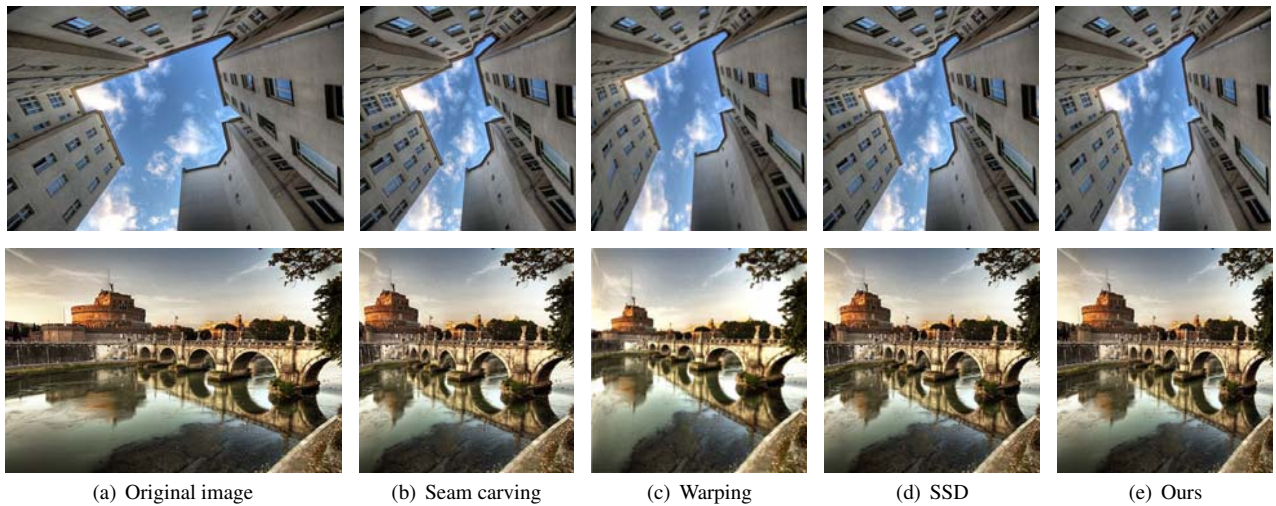
Figure 11: *Comparison of our results with those of seam carving, the optimized warping method [Wang et al. 2008] and our framework using SSD [Simakov et al. 2008]. The results of [Wang et al. 2008] and our method tend to be smoother than those of seam carving.*


(a) Original image   (b) Seam carving   (c) Scaling   (d) Cropping   (e) Warping   (f) Ours
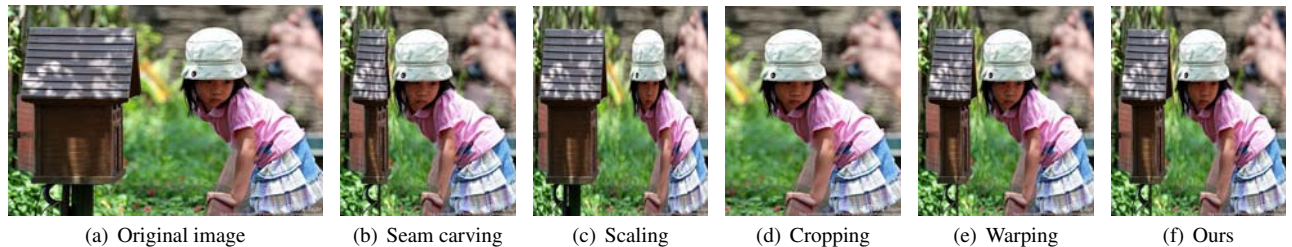
Figure 12: *Retargeting with saliency map. Our result keeps more information of the whole image. Warping is non-homogeneous scaling from [Zhang et al. 2008].*

(a) Original image   (b) Indirect seam carving   (c) [Wang et al. 2008]   (d) Ours (indirect)   (e) Ours (direct)
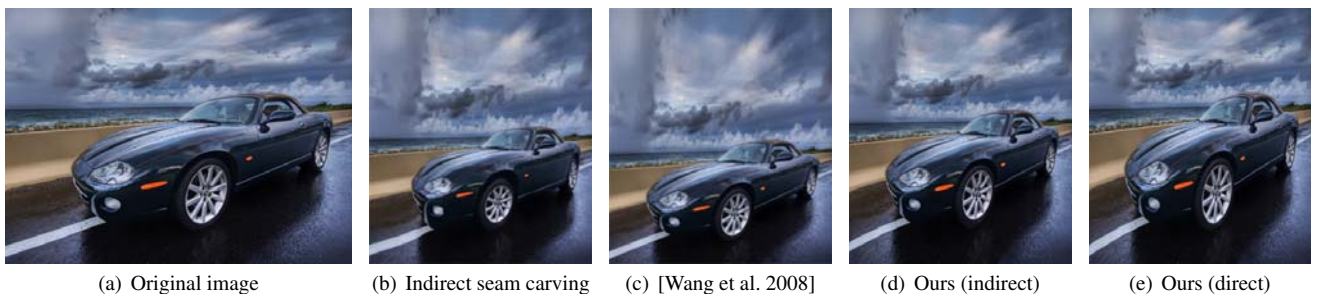
Figure 13: *Indirect image resizing with our algorithm. In (d), the original image is resized from 500×368 to 320×368 by two normal resizing steps and one direct scaling step.*

(a) Original image   (b) Seam carving   (c) Scaling   (d) Cropping   (e) Multi-operator   (f) Ours
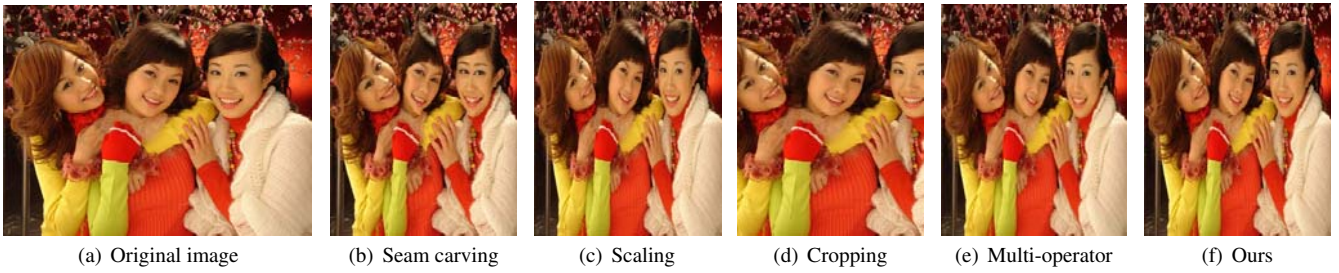
Figure 14: *Comparison of our results with those of seam carving, scaling, cropping and multi-operator [Rubinstein et al. 2009] (combining seam carving and scaling).*

(a) Original image   (b) Seam carving   (c) Scaling   (d) Cropping   (e) Multi-operator   (f) Ours
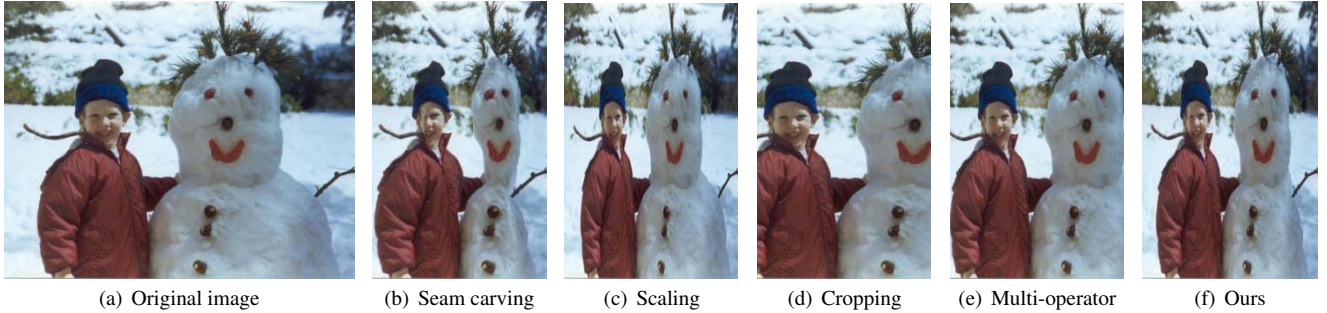
Figure 15: *Comparison of our results with those of seam carving, scaling, cropping and multi-operator [Rubinstein et al. 2009] (combining seam carving, scaling and cropping).*

best resizing. Here our algorithm also detected this character (with minimum distance to the original image) and the result image was exactly the same. For the "Secret #5" example, our algorithm preserves both the words and the notebook joint which are both partly damaged in the seam carving result. For the "Face" example, results show that we can use scaling alone to obtain a relatively good result, but our algorithm still optimally removes some seams in order to use less scaling operations to avoid distorting the prominent "Face" object. In Figure 9, we can see that our results outperform the SC and scaling methods by preserving both the local shape of the objects and the dominance of the prominent objects in the resized images.

Our algorithm takes the advantages of both the discrete SC and the continuous global scaling methods. Compared with the optimized warping methods such as [Zhang et al. 2008] and [Wang et al. 2008], our method is better at preserving the global visual effect and relative spatial aspect among the local objects. At the same time, the warping methods can avoid the SC-caused greediness effects [Wang et al. 2008] and protect well the aspect ratio of separate objects which are labeled by the importance map. In Figure 11, we compare our results with those of SC [Rubinstein et al. 2008] and optimized warping [Wang et al. 2008]. For SC results, notice the discontinuities in the buildings and bridge arches, which are due to the pixels being excessively removed. Compared with optimized warping, our method can preserve the spatial proportions of the objects better. As shown in the "Buildings" example, our image successfully avoided the slope distortion problem found in warping's result. In the "St Angelo Castle" example, our method maintains the relative aspect ratio between the castle and the bridge. The castle which is important in the original image is too small in warping's result. However, the aspect ratio of the bridge itself is protected in warping's image whilst our algorithm cannot preserve the ratio. Compared with the results using Simokov et al. [2008]'s SSD-based measure, our algorithm better preserves the object boundaries.

Another limitation of our algorithm is that the computation time is relatively long. However, as [Simakov et al. 2008], the computationally heavy nearest-neighbor search may be significantly made faster by using the location obtained from the previous evaluation step to constrain the search. We can also first do the optimization process at a lower resolution and predict the probable seam number range in the original resolution, or just manually limit the searching range of the seam number. Our algorithm could also be made more rapid by using an optimization scheme such as gradient descent [Snyman 2005]. Gradient descent might get stuck in local minima but may be good enough for some examples. The current implementation in C++ takes around 40 seconds to resize a 500×500 image to half size.

Our algorithm also supports other energy functions including saliency maps. In Figure 12, we used the algorithm of [Walthera and Koch 2006] to generate a saliency map. We can also see that our result strikes the best balance among all the objects in the image. In our result, the small wood house is bigger than the others but deform the girl's shape.

The order of SC and scaling is fixed in our algorithm. We do so because the structural objects in an image are more prominent than homogeneous ones. Starting with SC can preserve the shape of the dominant objects. Of course, switching the order of resizing operators may generate better results than current ones, it will be an important research direction in our future work.

Indirect Resizing. We can protect the aspect ratio of a prominent object in an original image by using interactively an indirect resizing method [Wang et al. 2008]. As shown in Figure 13, we first narrowed the original image from 500×368 to the target size 320×368 ($N_{SC_V} = 51$) using our resizing algorithm. The width-height proportion of the car was changed. Then we enlarged the image to 320×440 ($N_{SC_H} = 15$) and the height 440 was manu-

| (a) Original image | (b) Multi-operator | (c) Ours |

Figure 16: *Result of 2D retargeting. The multi-operator result (combining seam carving and scaling) is generated by applying Simakov et al. [2008]'s SSD-based image distance measure.*

ally chosen. In this step, the more homogeneous sky was enlarged more than the car. Finally, we directly scaled the image back to 320×368 to get the final result, and this will partly fix the width-height proportion change of the car appears in the direct resizing result (Figure 13(e)). Compared with the results of indirect SC and [Wang et al. 2008], our method better preserves the dominant status of the car.

Multi-operator retargeting (MOR).   In Rubinstein et al. [2009], a multi-operator media retargeting method is presented. The algorithm also integrates SC and scaling together to resize an image. BDW is employed to compare the similarity between the result and the original image. In Figure 14, we can see that our result is visually similar to the result of MOR, which computes the optimal mixed sequences using SC and scaling. In Figure 15, the MOR result outperforms ours due to the employment of one extra operator (cropping). We can see that sometimes the cropping operator is important for preserving the prominent objects. We compare the 2D retargeting results of different methods in Figure 16. In this example, our IMED-based measure is directly applied to find the best result, while the BDW in [Rubinstein et al. 2009] cannot. The running time will be dramatically increased if the two-dimension BDW is used. We can also see that our measure is better in preserving the aspect ratio of some objects, for example the keyboard in Figure 16. Furthermore, the DCD term in our algorithm is useful in balancing the prominent objects and the global visual effect in an image (as shown in Figure 4). This is more flexible than the pure patch-based BDW for some examples.

6   Conclusion and Future Work

We presented a novel technique for content-aware image resizing. Seam carving and homogeneous scaling were integrated together to form a hybrid system (combining discrete and continuous methods). A well defined image distance measure was used to optimize the resizing process. The measure was formulated as a distance which integrated bidirectional IMED, DCD similarity and an additional seam energy-based coefficient together. We described a principle approach to reduction/retargeting and enlarging of images. Furthermore, we demonstrated applications of our technique in indirect image resizing.

In addition to the arbitrary 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. In our technique we use dominant colors to evaluate the change of the global visual appearance. One potential solution is to keep colors that do not appear much in the image, so that small objects do not disappear. Currently the small but important objects could be protected by seam carving if they have high energy. Preservation of the aspect ratios and sizes of prominent objects in our framework is also a challenging problem. We also plan to extend our image distance measure to other

applications, such as automatic cropping, texture synthesis and image retrieval. Nevertheless, integration of more operators such as cropping and non-homogeneous warping into the resizing process is another interesting research direction.

References

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

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

CHO, T. S., BUTMAN, M., AVIDAN, S., AND FREEMAN, W. T. 2008. The patch transform and its applications to image editing. In *IEEE Conference on Computer Vision and Pattern Recognition 2008 (CVPR 2008).*, 1–8.

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

EL-ALFY, H., JACOBS, D., AND DAVIS, L. 2007. Multi-scale video cropping. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, ACM, New York, NY, USA, 97–106.

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 Transactions on Pattern Analysis and Machine Intelligence 20*, 11 (Nov), 1254–1259.

LI, J., AND LU, B.-L. 2009. An adaptive image euclidean distance. *Pattern Recogn. 42*, 3, 349–357.

LIU, H., XIE, X., MA, W.-Y., AND ZHANG, H.-J. 2003. Automatic browsing of large pictures on mobile devices. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, ACM, New York, NY, USA, 148–155.

MANJUNATH, B. S., OHM, J. R., VASUDEVAN, V. V., AND YAMADA, A. 2001. Color and texture descriptors. *Circuits and Systems for Video Technology, IEEE Transactions on 11*, 6, 703–715.

MANJUNATH, B., SALEMBIER, P., AND SIKORA, T. 2002. *Multimedia Content Description Interface*. Wiley, Chichester.

MIN, R., AND CHENG, H. D. 2009. Effective image retrieval using dominant color descriptor and fuzzy support vector machine. *Pattern Recogn. 42*, 1, 147–157.

PRITCH, Y., KAV-VENAKI, E., AND PELEG, S. 2009. Shift-map image editing. In *ICCV 2009: Proceedings of the Twelfth IEEE International Conference on Computer Vision*, to be appeared.

RUBINSTEIN, M., SHAMIR, A., AND AVIDAN, S. 2008. Improved seam carving for video retargeting. *ACM Trans. Graph. 27*, 3, 1–9.

RUBINSTEIN, M., SHAMIR, A., AND AVIDAN, S. 2009. Multi-operator media retargeting. *ACM Trans. Graph. 28*, 3, 1–11.

SANTELLA, A., AGRAWALA, M., DECARLO, D., SALESIN, D., AND COHEN, M. 2006. Gaze-based interaction for semi-automatic photo cropping. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, ACM, New York, NY, USA, 771–780.

SIMAKOV, D., CASPI, Y., SHECHTMAN, E., AND IRANI, M. 2008. Summarizing visual data using bidirectional similarity. In *IEEE Conference on Computer Vision and Pattern Recognition 2008 (CVPR 2008)*, 1–8.

SNYMAN, J. A. 2005. *Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms*. Springer Publishing.

SUH, B., LING, H., BEDERSON, B. B., AND JACOBS, D. W. 2003. Automatic thumbnail cropping and its effectiveness. In *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology*, ACM, New York, NY, USA, 95–104.

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

WALTHERA, D., AND KOCH, C. 2006. Modeling attention to salient proto-objects. *Neural Networks 19*, 9, 1395–1407.

WANG, L., ZHANG, Y., AND FENG, J. 2005. On the euclidean distance of images. *IEEE Trans. Pattern Anal. Mach. Intell. 27*, 8, 1334–1339.

WANG, Y.-S., TAI, C.-L., SORKINE, O., AND LEE, T.-Y. 2008. Optimized scale-and-stretch for image resizing. *ACM Trans. Graph. 27*, 5, 1–8.

WEI, L.-Y., HAN, J., ZHOU, K., BAO, H., GUO, B., AND SHUM, H.-Y. 2008. Inverse texture synthesis. *ACM Trans. Graph. 27*, 3, 1–9.

WOLF, L., GUTTMANN, M., AND COHEN-OR, D. 2007. Non-homogeneous content-driven video-retargeting. In *Proceedings of the Eleventh IEEE International Conference on Computer Vision (ICCV-07) ICCV 2007*, 1–6.

ZHANG, Y. F., HU, S. M., AND MARTIN, R. R. 2008. Shrinkability maps for content-aware video resizing. *Computer Graphics Forum 27*, 7, 1797–1804.

A   Dominant Color Extraction

We propose a novel method to efficiently extract the DCD of an image. A clustering algorithm is used to extract the dominant colors from the color histogram of the image. The steps are summarized as follows:

1. Transfer an image in RGB to an YIQ color space.

2. Build a histogram, $\{Hist(Y, I, Q), 0 \leqslant Y \leqslant L_Y, 0 \leqslant I \leqslant L_I$, and $0 \leqslant Q \leqslant L_Q\}$ in a 3-D (Y, I, Q) space, with L_Y bins along the Y direction, L_I bins along the I direction, and L_Q bins along the Q direction. We set $L_Y = L_I = L_Q = 16$ in our experiments.

3. Choose the N_{DCD} bins with the largest number of pixels. Set the average color of each bins $\mathbf{index}_i$ as the initial dominant colors $\mathbf{v}_i$.

4. Use generalized Lloyed algorithm for color clustering. This problem is formulated as minimizing the distortion D_i in each cluster i

$$D_i = \sum_{j=1}^{n} w(j)\|\mathbf{index}_j - \mathbf{v}_i\| \quad \mathbf{index}_j \in C_i$$

where n is the number of histogram bins in the cluster, $\mathbf{v}_i$ is the centroid of cluster C_i, $\mathbf{index}_j$ is color vector at bin, $w(j)$ is perceptual weight for bin j. We simply set $w(j) = 1$ in our experiments except when an importance map is used ($w(j)$ is the importance value).

5. For each selected cluster $i(i = 1, 2, \ldots, N_{DCD})$, calculate the centroid $\mathbf{v}_i$ from the formula

$$\mathbf{v}_i = \frac{\sum_j \mathbf{index}_j \cdot N_j}{\sum_j N_j}$$

where $\mathbf{index}_j$ is the average color vector of bin j in the YIQ color space, and N_j is the number of pixels at bin j. The percentage p_i is calculated as

$$p_i = \frac{\sum N_j}{N_{IM}}$$

where N_{IM} is the total number of pixels. Moreover, we set c_i as the standard variance of each cluster.

6. Based on each cluster in descending order of the number of cluster, fill in the DCD structure of Equation (4) with p_i, c_i, and $\mathbf{v}_i$.