Multi-operator Media Retargeting

Michael Rubinstein[1] Ariel Shamir[1] Shai Avidan[2]

[1]The Interdisciplinary Center, Herzliya [2]Adobe Systems Inc.

Original Cropping Scaling Seam carving Warping Multi-operator

Figure 1: The multi operator algorithm uses dynamic programming to find the optimal combination of retargeting operators. Here we show a comparison of several methods. The original image (left) is retargeted using: simple cropping, uniform scaling, seam carving [Rubinstein *et al.* 2008], non-uniform warping [Wang *et al.* 2008] and our multi-operator algorithm. In this example, the multi-operator algorithm combines cropping, scaling and seam carving to optimize our new image-to-image similarity measure, termed Bidirectional Warping (BDW). The algorithm can use other retargeting operators and similarity measures.

Abstract

Content aware resizing gained popularity lately and users can now choose from a battery of methods to retarget their media. However, no single retargeting operator performs well on all images and all target sizes. In a user study we conducted, we found that users prefer to combine seam carving with cropping and scaling to produce results they are satisfied with. This inspires us to propose an algorithm that combines different operators in an optimal manner. We define a *resizing space* as a conceptual multi-dimensional space combining several resizing operators, and show how a path in this space defines a sequence of operations to retarget media. We define a new image similarity measure, which we term Bi-Directional Warping (BDW), and use it with a dynamic programming algorithm to find an optimal path in the resizing space. In addition, we show a simple and intuitive user interface allowing users to explore the resizing space of various image sizes interactively. Using key-frames and interpolation we also extend our technique to retarget video, providing the flexibility to use the best combination of operators at different times in the sequence.

Keywords: media retargeting, multi-operator, resizing space, bi-directional warping

1 Introduction

Media retargeting has become an important problem due to the diversity of display devices and versatility of media sources for both images and video. Recently, content aware methods such as seam carving and non-uniform warping were proposed to supplement content oblivious methods such as scaling or cropping. A content aware retargeting operator relies on an importance map to preserve the important parts of the media at the expense of the less-important ones. Importance measures include image gradients, saliency and entropy, as well as high level cues such as face detectors, motion detectors and more.

However, content aware methods do not succeed in all cases and for all sizes. For example, in case the important object occupies large portions of the image or video frame, content aware resizing might distort it. Often, the best resizing method depends on the image itself: one method might work best on one image, while another on a different image. In such cases using a combination of several methods (operators) might achieve better results than any specific one alone (Figure 1). In this paper we propose to combine several operators together, instead of searching for the best operator that will work on all images. Our approach is supported by a user study we conducted that clearly shows that users prefer to use more than one operator to achieve better results.

We first define the *resizing space* as a conceptual multi-dimensional space combining several retargeting operators. Each axis in this space corresponds to a particular type of operator, and a point in this space corresponds to a particular target image size. A path in this space defines a sequence of operations that retargets an image to a particular size (Figure 4). Many paths arrive at the same point, meaning that there are many ways to retarget an image to a particular size. But not all paths are created equal because resizing operators are *not* commutative (e.g. scaling followed by cropping is different from cropping followed by scaling).

To combine several operators there is a need to compare and evaluate different retargeting results. Hence, we need some global similarity measure between the source and target images. And given the similarity measure, we need an algorithm that maximizes this measure by finding the best path (i.e. sequence of operators) to the respective point in resizing space.

In this paper we propose a novel similarity measure between images that we term Bi-Directional Warping (BDW). This measure

is based on a non-symmetric variant of Dynamic Time Warping (DTW) [Sakoe 1978]. DTW takes two $1D$ signals (e.g. rows or columns of pixels) and finds the best non-uniform alignment between them, subject to order constraints. To measure the similarity between two images, BDW measures the similarity between every row (or column) and then takes the *maximum* alignment error as the distance. We also extend the measure to work on a row (or column) of patches instead of pixels, as patches can better capture spatial information.

There are infinitely many paths that can be used to retarget an image. Unless mentioned otherwise, we focus on *monotonic* paths, i.e. paths where all operators either increase the size of the image, or decrease it, but not both. Of all the monotonic paths, we consider two types of paths that we term *regular* and *mixed*. A *regular* path is composed of consecutive single operator sequences, one per operator (e.g. first apply seam carving, then cropping and finally scaling). In this case, the only question left is how many times to apply each operator in the retargeting process? The search space is polynomial in the image size and can be enumerated to find the optimal *regular* path. However, in a *mixed* path, the order of the operations, as well as the number of times each operator is used is not fixed. Hence, the search space is exponential in the image size. However, using a simple assumption we show a polynomial algorithm that automatically determines the optimal *mixed* multi-operator path. In both cases the search space is exponential in the number of retargeting operators. Nevertheless, the number of operators is typically very small (say four operators), making the solutions tractable.

It is worth noting that the multi-operator algorithm can work with various image similarity measures as well as different retargeting operators. *Regular* paths can also be controlled by the user and we show a simple user interface for image retargeting. Finally, we extend the *regular* path approach to video retargeting by interpolating paths between key-frames. This approach provides the flexibility to use the best combination of operators at different times in the video. We demonstrate our approach for high quality reduction and expansion of images and videos.

Our main contributions are as follows, 1. We show that using several operators can potentially give better results for retargeting than using a single operator, 2. We present a new global measure, Bi-Directional Warping, to assess the retargeting results, 3. We give an algorithm for finding an optimal multi-operator retargeting sequence under some assumptions, 4. We describe an intuitive user interface that helps users combine multiple operators interactively, and 5. We show how our method is extended to support multi-operator video retargeting.

2 Background

Content-aware retargeting has drawn a lot of attention in recent years. Most methods proposed use a two-step approach where first some saliency or importance map is created from the media and then a resizing operator is applied based on this map. As our work concentrates on combining multiple operators and not on saliency, we focus on the different types of operators for resizing media.

Cropping was used by Suh *et al.* [2003] for automatic thumbnail creation, based on either a saliency map or the output of a face detector. Similarly, Chen *et al.* [2003] considered the problem of adapting images to mobile devices, by automatically detecting the most important connected region in the image and transmitting it to the mobile device. Liu *et al.* [2003] also addressed image retargeting to mobile devices, suggesting to trade time for space. Given a collection of Regions Of Interest (ROI), they construct an optimal path through these regions and display them in a consecutive manner. Santella *et al.* [2006] use eye tracking, in addition to composition rules to crop images intelligently. Setlur *et al.* [2005] use segmentation and re-composition for non-photorealistic retargeting.

Several different methods could be characterized as non-homogeneous scaling. Liu and Gleicher [2006] find the ROI and construct a novel Fisheye-View warp that essentially applies a piecewise linear scaling function in each dimension to the image. This way the ROI is maintained while the rest of the image is warped. In their video retargeting work they use a combination of cropping, virtual pan and shot cuts to retarget the video frames. Gal *et al.* [2006] solve the general problem of warping an image into an arbitrary shape while preserving user-specified features. The feature-aware warping is achieved by a particular formulation of the Laplacian editing technique, suited to accommodate similarity constraints on parts of the domain. Wolf *et al.* [2007] extend this approach to video using non-homogenous mapping of the source video frames to the target resized frames. They use a combination of motion detectors and face detectors to define the saliency map. A different approach presented by [Wang et al. 2008] partitions the image into a grid mesh and deforms it to fit the new desired dimensions. Important image regions are optimized to scale uniformly while regions with homogeneous content are allowed to distort.

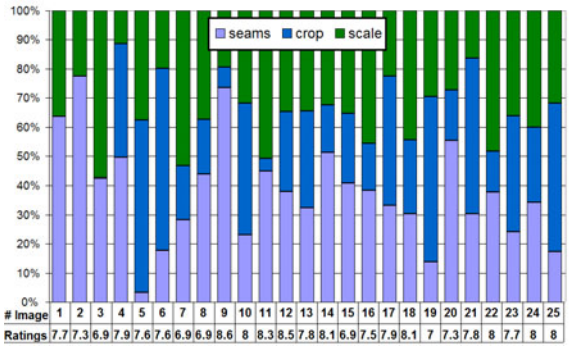
Recently, several works used the seam carving operator originally proposed by Avidan and Shamir [2007] to resize images in a content aware fashion. They use dynamic programming to find the optimal seam in an image according to some image energy map (usually based on the gradient field of the image). The seams can be removed for shrinking images, or duplicated for expanding them. Later, this work was extended by Rubinstein *et al.* [2008] for video retargeting. The dynamic programming was replaced by a graph cut approach and a new image energy was proposed that creates less artifacts in the resulting media. Graph cuts were also used by Chen and Sen [2008] for temporally resizing video.

In cases where one of these operators does not perform well, it might be better to use another or revert to simpler resizing methods such as cropping and scaling. On the one hand the latter methods are not content aware, but on the other, they can be considered less harmful as they do not distort the media. The key question is how to decide when one operator fails, and which operator to use instead?

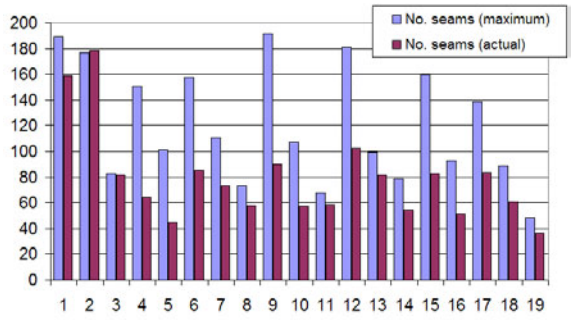
Some measures were suggested, for example, by Avidan and Shamir [2007] to indicate the order of seam carving by their cost. However, we have not found this cost to be indicative for measuring retargeting quality. Moreover, similar measures are not easy to find for other operators. We follow more global measures such as the bi-directional similarity [Simakov et al. 2008] and inverse texture synthesis [Wei et al. 2008] that define image similarity. Two images S and T are considered visually similar if all patches of S (at multiple scales) are contained in T, and vice versa. Although this approach is effective on several applications for summarization and synthesis, it does not preserve the order of elements inside the image. Trying to match two images while preserving full order is a difficult problem [Keysers and Unger 2003]. Still, using some constraints we present a variant of Dynamic Time Warping [Sakoe 1978; Uchida and Sakoe 1998] that can be utilized to measure retargeting quality.

3 The User Study

Our basic hypothesis in this work is that using multiple operators for resizing images is often better than using a single one. To assess this hypothesis we conducted an experiment where users are given the option to use a combination of three operators: seam carving, cropping and scaling, including also the option of using just a single one. We present users with an original image in one window,



(a) The mean of the ratio of operators used for retargeting each image using multiple operators. Although the ratio depends on the image, in all cases, when given the option, users combine several operators to achieve better results. Users were asked to rate the results between $1-10$. As can be seen most users were satisfied with the retargeting results (average 7.7). For image enlargement (the first three results) only seam carving and scaling were allowed. Some of the actual images are shown below.



(b) Comparing the maximum number of adequate seam removals (or insertions) to the actual number when using multiple operator retargeting. The mean values of the overlapping participants and images in both experiments are shown. In most cases, users prefer switching to other operators for retargeting even though the seam carving results were rated as adequate. Note that when images are enlarged, i.e. seams are inserted (first three results), the numbers are much closer, and in one case the number is even slightly larger.

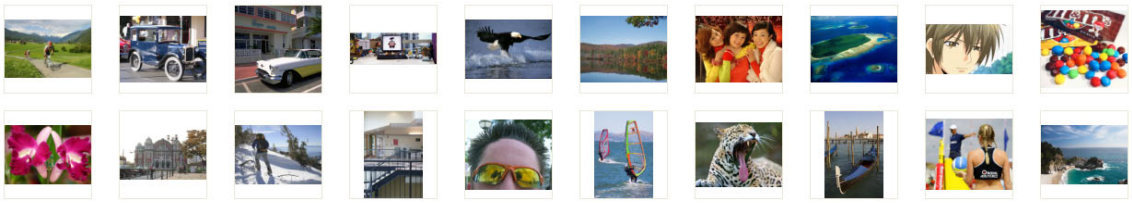


Figure 2: A user study of 50 participants clearly indicates that combining multiple operators can be beneficial for retargeting.

and in another window, a resized image that is either reduced or enlarged in one dimension to a fixed size (the change in size was between 30% to 60% of the original size). We specifically chose images that contain either structure or content that presents difficulties for existing methods. The resized image is retargeted using a combination of the above three operators using *regular* sequences. For example, to reduce the width of an image by n pixels, first n_1 seams are removed, then n_2 columns are cropped from the left and right sides of the image and lastly the image is scaled by n_3 pixels, where $n_1 + n_2 + n_3 = n$. Users were asked to change the ratio between n_1, n_2, and n_3 interactively using a scroll bar while examining the resulting image, until they reach the best results for the given fixed size. Note that for image enlargement only seam carving and scaling were used while the ratio between them could be changed. The user interface itself is described later in Section 7.

Figure 2(a) summarizes the results of this experiment for 50 participants. Most participants were computer-science students or graphic designers. They were asked to rate their graphical background level from novice to expert: 28 rated themselves as novices, 15 as intermediates, and 7 as graphic experts. 22 images of different nature were used in the experiment (Figure 2 bottom). As can be seen from the combined mean results (no significant differences were found between the groups) we have $n_i > 0$ for all $i = 1, 2, 3$. Moreover, for almost all images and all participants we had $n_i > 0$ for all i. In general, this suggests that better results are achieved using a combination of more than one operator.

In a different experiment, users were given the option to change the size of an image using seam-carving alone and were asked to minimize (or maximize) the width (or height) of the image as long as the resulting image appears visually adequate. Figure 2(b) compares the mean of the number of seams removed (or inserted) in this experiment to the number of seams actually removed (or inserted) while using multiple operator resizing on the same images in the first experiment. Results clearly show that although users were satisfied with the quality of removing (or inserting) more seams from an image, they still preferred using other operators while retargeting the image to achieve better visual results.

4 The Resizing Space

4.1 Multi-Operator Sequences

We define a *retargeting operator* O as a procedure that reduces or enlarges an image either in its width or its height, while preserving its rectangular shape. We concentrate on retargeting operations that are *discrete* and *separable* (in dimension). This means that the atomic operation in our setting is adding or removing one pixel to the width or the height of the image. Two dimensional resizing can be treated as a sequence of width and height resizing, which means that different operators can be used for different dimensions (e.g. use scaling for height change and seam carving for width change). In this paper we use bi-cubic scaling (SL), cropping (CR) and seam carving (SC). This particular combination seems promising, as these operators take somewhat complementary approaches to resize media. Using other operator combinations is left for future work.

Not all retargeting operators can actually support enlarging. For instance, cropping is usually used only for reducing image size. However, for the sake of completeness we define crop-enlarging as adding a black frame to an image (letter-boxing). Similarly, to make cropping separable, we remove rows or columns from the image borders independently. We also choose the sides separately, for instance, either the left or the right column is removed depending on which has the lower cost. Scaling can support separable and dis-

(a) (b) (c) (d) (e) (f)

Figure 3: Different multi-operator sequences create a variety of results for retargeting the width of an image: (a) original image, (b-f) results using the following sequences respectively: $\langle 0 \cdot \text{SC}, -178 \cdot \text{CR}, 0 \cdot \text{SL} \rangle$, $\langle 0 \cdot \text{SC}, 0 \cdot \text{CR}, -178 \cdot \text{SL} \rangle$, $\langle +48 \cdot \text{SC}, -149 \cdot \text{CR}, -77 \cdot \text{SL} \rangle$, $\langle -30 \cdot \text{SC}, -148 \cdot \text{CR}, 0 \cdot \text{SL} \rangle$, $\langle -32 \cdot \text{SC}, 0 \cdot \text{CR}, -146 \cdot \text{SL} \rangle$.

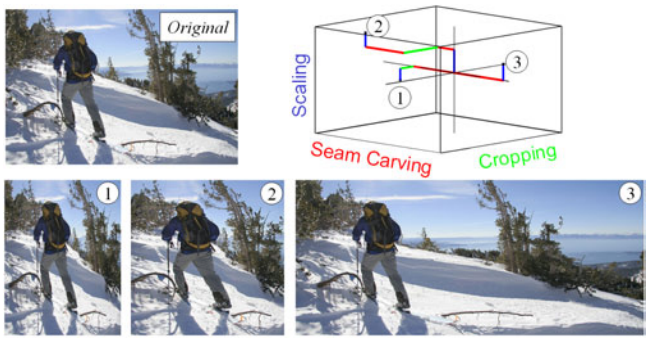


Figure 4: An example of a resizing space of an image using only changes in width by scaling, cropping and seam carving. Different retargeting results can be achieved using different multi-operator sequences represented by paths in the space.

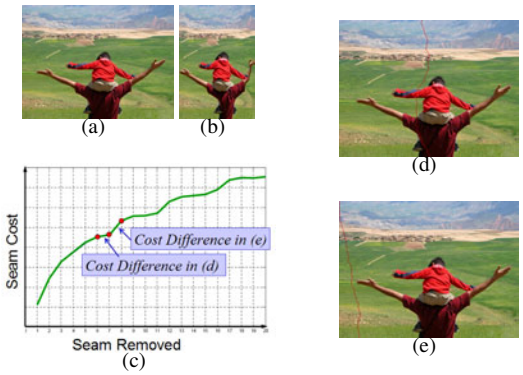


Figure 5: Retargeting using seam carving can destroy image content (a-b). Examining the seam cost function (c) as in [Avidan and Shamir 2007] one cannot anticipate when foreground objects are distorted. If we track not only the actual cost, but also the difference in the cost of removing seams, we can find seams that pass through the main object (d) but have smaller cost and smaller difference in cost than seams that pass only through the background (e). We experienced similar behavior when using the maximal pixel energy along a seam instead of average. Hence, seam cost cannot be used as a measure for retargeting quality.

crete resizing, but scaling an image by one pixel k-times is inferior to scaling by-k at once. Hence, whenever applicable we perform a scale by-k instead of applying k 1-pixel scalings.

Combining several operators together in an ordered sequence is a *multi-operator sequence*. Note that a certain type of operator can appear multiple times in different places in the sequence; in some it can be used to enlarge the image and in others to reduce it, and also in different directions (width and height). Figure 3 shows examples of different multi-operator sequences that create different valid variations for retargeting an image.

4.2 The Resizing Space

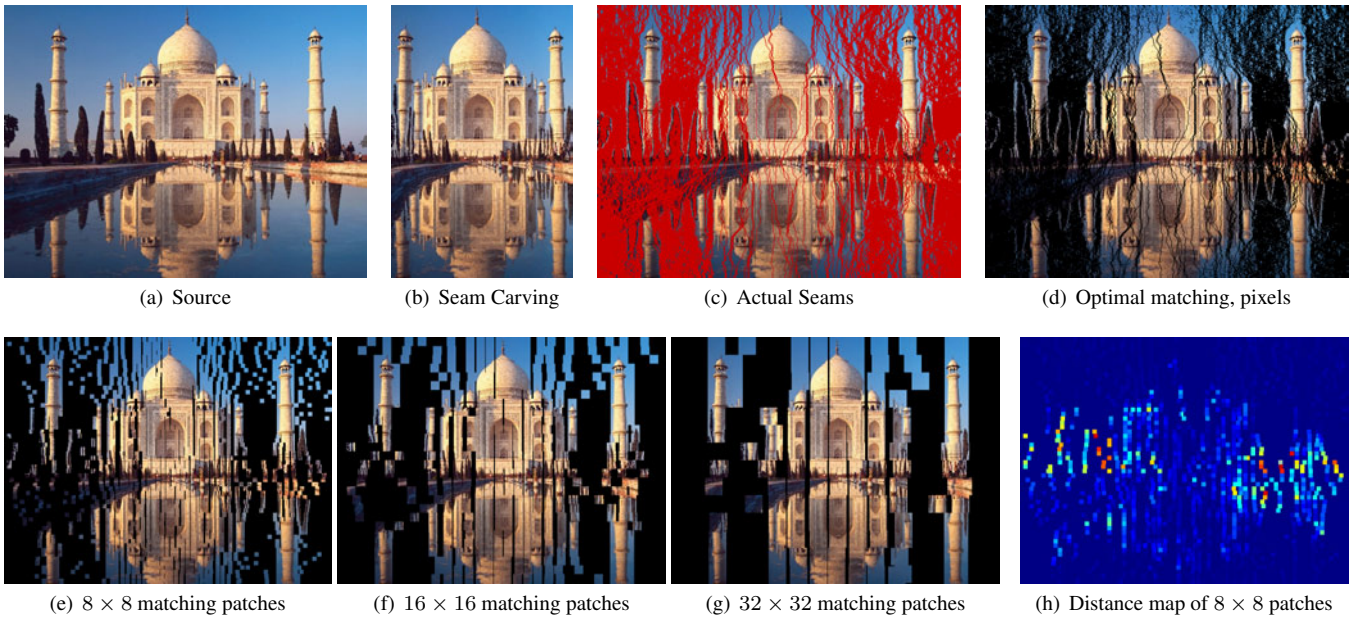
For a given image I of size (w, h) we define the *resizing space* Φ as the space spanned by any subset of n types of retargeting operators, each one in two directions - width and height. Hence, the dimension of this space is at most $2n$. A multi-operator sequence defines a directed path in this space beginning at the origin and following the path's operator sequence using integer steps. One step in the operator sequence is equivalent to a step either in the positive or negative direction of the respective operator axis, which can change either the width or the height of the image. Since only integer steps are used, we treat the resizing space as a lattice rather than a continuous space (Figure 4).

For a k-dimensional resizing space, $k \leq 2n$, a point on this lattice $p \in \Phi, p = (p_1, \ldots, p_k), p_i \in \mathbb{Z}$, represents the set of images $\{I'\}$ whose dimensions are (w', h') where $w' = w + \sum_i p_i^w$ and $h' = h + \sum_j p_j^h$, and p_i^w, p_j^h are the coordinates in $(p_1, \ldots, p_k)$ representing operators that change the width or height respectively. These coordinates can be positive as well as negative to signify enlargement or reduction of size. There is an infinite number of such images for each point p in the resizing space since there is an infinite number of paths starting at the origin and ending at p. All such paths define multi-operator sequences where the change of width or height by each specific operator i is fixed and amounts to the coordinate p_i. However, the order of applying the operators can be different. To complicate things further, there is an infinite number of points $q \in \Phi$ that represent images of dimensions (w', h'). For r horizontal operators and t vertical operators $(r + t = k)$, these are points for which $\sum_{i=1}^{r} p_i^w + w - w' = 0$ and $\sum_{j=1}^{t} p_j^h + h - h' = 0$. Each of these equations represent a $r + t - 1 = k - 1$ dimension hyperplane in the k-dimensional resizing space, the intersection of which is a hyperplane of dimension $k - 2$ (for $r, t > 0$). All points in this subspace represent images of the required size, where the difference between them is in the amount of applying specific operators (i.e. the ratio between them). Our main goal therefore, is to find the best path from the origin to one of these points, subject to some global image similarity measure.

5 Bi-Directional Warping

5.1 Motivation

As a motivating example, consider the task of combining seam carving and scaling. One way to combine the two is to start with seam carving and then switch to scaling when the cost of a



(a) Source (b) Seam Carving (c) Actual Seams (d) Optimal matching, pixels

(e) 8×8 matching patches (f) 16×16 matching patches (g) 32×32 matching patches (h) Distance map of 8×8 patches

Figure 6: Finding the optimal match using Asymmetric-DTW of image (b) to image (a) using different patch sizes (d)-(g). The element-wise distance function $d()$ was taken as the L_1-norm of RGB differences. Note that black pixels represent gaps in the matching. The distance itself is defined as the average or maximum of the cost of matching each element (h).

seam goes above a certain threshold, as this might indicate that seam carving starts introducing visual artifacts. In fact, Avidan and Shamir [2007] used the seam cost to find the optimal "multi-operator" sequence using just seam-carving to change both the width and the height of an image. Unfortunately, in our experiments we found that the seam cost is not very indicative of the quality of retargeting. It is a monotonically increasing function (with some local fluctuations), and usually does not contain steps that indicate when "bad" seams are removed. Figure 5 illustrates this with a specific example. Hence, to combine seam carving with scaling the seam cost should not be used. In the case of other operators, such as cropping and scaling, even the definition of an effective cost for the operator is not immediately clear. So instead of dealing with each potential operator independently, we need a global objective function that will allow us to combine operators in a principled manner.

We define the cost of applying an operator as the difference between the resulting image and the original image. Although the definition of distance between pair of images is an ongoing research problem for many years, in our setting there are several simplifying factors. First, we know that the target image is a resized version of the source image and aspires, by definition, to preserve its content as much as possible. Second, each application of an operator changes the image size in one direction. We define our objective function as a bi-directional relation between the images [Simakov et al. 2008] that conforms to the above constraints. Specifically, we account for the first factor by enforcing order on this relation.

The resulting similarity measure, which we term Bi-Directional Warping (BDW), uses a variant of Dynamic Time Warping (DTW) that is geared specifically to the problem of media retargeting.

5.2 Dynamic Time Warp

Dynamic Time Warping (DTW) [Sakoe 1978], is an algorithm for measuring similarity between two one-dimensional signals or time-series. It has been previously applied to various applications in video and images, and is extensively used with audio signals for speech recognition. The DTW algorithm finds the optimal matching between two 1D sequences t and s by non-linearly warping the one to the other, under several constraints: (1) boundary constraints: the first and last elements of t must be matched to the first and last elements of s, respectively, (2) all elements of t and s must be used in the warp path, and (3) the warp must be *monotonic*, meaning that matching cannot go backward, thus preserving the sequence order. It is easy to see that the warp is symmetric, that is, DTW(s,t)=DTW(t,s), and can contain both one-to-many and many-to-one matchings. This algorithm can be solved efficiently using dynamic programming, in $O(|s||t|)$ time and space.

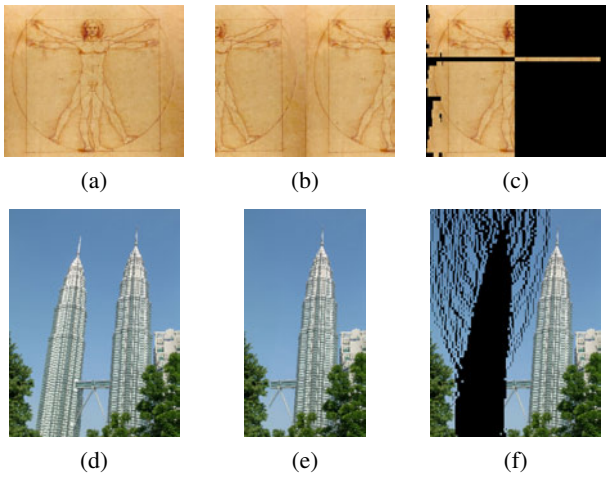
5.3 Bi-Directional Warping

We relax the first two constraints of DTW. First, we allow the algorithm to insert gaps in the warp, which also removes the boundary constraints. Second, for each element in the source we want a *single* match that minimizes the warping cost under the ordering constraint. Therefore, one-to-many matchings from the source to target image are disallowed. We do allow many-to-one matchings from the source to target image as it assists better matches and does not violate the ordering constraint. This creates an Asymmetric-DTW measure (A-DTW) detailed in Algorithm 1. The signals s and t can be either $1D$ arrays of pixels, or $1D$ arrays of patches, and the distance $d(s[i], t[j])$ between an element of s and element of t is taken to be the sum-of-square-differences of pixel values[1]. Given images S and T of height h, let S_i, T_i denote row i in images S and T, respectively. The BDW distance is given by:

$$\text{BDW}(S,T) = \frac{1}{N_S}\sum_{i=1}^{h} \text{A-DTW}(S_i, T_i) + \frac{1}{N_T}\sum_{i=1}^{h} \text{A-DTW}(T_i, S_i) \quad (1)$$

We found that using the $\max$ operator works better than the mean in equation 1, because in retargeting most elements are usually well

[1] We have experimented with numerous measures, such as the L_1 and L_2 norms of the intensity differences in both grayscale, RGB and CIE L*a*b* colorspaces.



(a) (b) (c)

(d) (e) (f)

Figure 7: Comparing Bidirectional Warping (BDW) and Bidirectional Similarity (BDS) of [Simakov *et al.* 2008]. In image (b) we switched the left and right parts of image (a), and image (e) is missing some repetitive structure (a tower) found in image (d). In both cases the BDS is one order of magnitude smaller than BDW as measured in equation 1 (it is two order of magnitudes smaller if we use max instead of mean). This is because every patch in one image will have, with high probability, a similar patch in the other image, and vice versa. On the other hand, the BDW measure is order-preserving, thus recovering the best alignment (as shown in images (c) and (f)) resulting in larger gap errors.

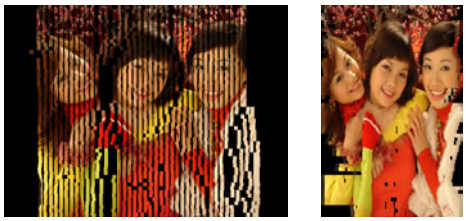
Algorithm 1 Asymmetric-DTW($s[1..|s|], t[1..|t|]$)

allocate $M[|s|+1][|t|+1]$
$M[0,0] := 0$
for $i = 1$ to $|s|$ **do**
  $M[i,0] := \infty$
for $j := 1$ to $|t|$ **do**
  $M[0,j] := 0$
for $i := 1$ to $|s|$ **do**
  for $j := 1$ to $|t|$ **do**
    $M[i,j] := \min(M[i-1,j-1] + d(s[i],t[j]),$
    $M[i,j-1],$
    $M[i-1,j] + d(s[i],t[j]))$
return $M[|s|,|t|]$

aligned, yet a small number of deformed elements are enough to cause a visual artifact. To find the maximum distance between elements from S and T, we need to recover the elements' alignment created by the asymmetric-DTW. To do this, we keep track of our path while filling the table M, and backtrack from $M[|s|,|t|]$ to $M[1,1]$ according to the optimal decisions made along the path. Figure 6 illustrates the results of aligning an image with its retargeted version (by seam carving) using A-DTW and several patch sizes. In practice, to calculate the BDW we combine the scores of four scales of patch size (Figure 8). For images S of size $h \times w$ and T of size $h \times w'$, $w' < w$, BDW is $O(hw^2)$ in time, and $O(w^2)$ in space. For example, the BDW between two 640×480 images and using 8×8 patches takes about 3 seconds to compute. Various methods exist to further optimize this computation (see e.g. [Salvador and Chan 2007]).

Both BDW and bidirectional similarity (BDS) explain patches in one signal using patches from the other. However, there are two main differences between these measures. First, BDW searches for

(a) S (original) (b) T (retarget)



(c) T→S (d) S→T

Figure 8: BDW uses two alignments of the source image to the retargeted image and vice versa. The distance would be the maximal matching cost of elements in the two. Note how the alignment reveals that the retargeting used a combination of cropping (gaps on the sides of the alignment), scaling (uniform spacing between patches in the middle) and seam carving (large gaps in the middle).

matches along a single direction (column or row) as opposed to the entire image in BDS. This is sufficient in our settings for assessing results created by operators which are separable in dimension. It is also more efficient to compute as we replace the computationally intensive nearest-neighbor search in BDS with an efficient matching algorithm. Second, BDW achieves optimal alignment that is order preserving. Order is important when assessing retargeting results, because we prefer as few as possible structural modifications of the media. The many-to-one alignment also supports repetitive content, albeit in an order-preserving manner. Figure 7 highlights the difference between BDW and BDS in the context of image retargeting.

6 The Optimization

Suppose we want to reduce the width w of input image S by m pixels, using a collection of n operators $\{O_1, \ldots, O_n\}$, and given some similarity measure $D()$ (e.g. BDW). This means that we seek a target image T of width $w' = w - m$ that minimizes $D(S,T)$. Even if we use monotonic sequences (e.g. do not reduce, then extend, then reduce back again), there are still $O(n^m)$ different multi-operator sequences that retarget S to width w'. This means the search space is exponential in the size change m.

To solve this problem we need to limit our search space and we consider two types of paths: *mixed* and *regular*. We will focus on *mixed* paths here and defer discussion on *regular* paths to section 7. Recall that we define a *mixed* path to be a path where we don't know, ahead of time, the order of the operators, nor the number of times each operator is to be used.

The basic assumption we use is that the ratio of operators in a sequence (i.e. the total amount each one is used) is more important than their order in the sequence. This leads to a dynamic programming formulation of the problem. In our search we always keep just one representative for each sequence with a given ratio of operators. We represent it by the point $(p_1, \ldots, p_n)$ in resizing space where p_i, the coordinate for operator i, denotes the total number of times of applying operator i. In a dynamic programming table we store

(a) (b) (c) (d) (e) (f) (g) (h)

Figure 9: An illustration of the dynamic programming table used to optimize the search for the best *mixed* path using two operators only - seam carving (SC) and scaling (SL). The colors in table (b) indicate the BDW distance of the best image in each step. The original image is shown in (a) and the retargeted result is shown in (c) - this is the best result using a *mixed* path (i.e. the algorithm automatically determines the order of operators and how much each should contribute). The optimal operator sequence found is $\langle -30SL, -30SC, -10SL, -20SC, -10SL, -10SC, -10SL, -20SC, -10SL\rangle$. For comparison, we show the results of using two *regular* paths (d) $\langle -70SL, -80SC\rangle$ and (e) $\langle -80SC, -70SL\rangle$, and the optimal *regular* path (f) $\langle -90SC, -60SL\rangle$. (g) uses scaling and (h) seam carving.

the optimal cost and optimal sequence $\sigma(p_1, \ldots, p_n)$ including the order of applying all the operators for this representative point.

We begin with an empty sequence denoted by the point $(0, \ldots, 0)$ and cost 0. Next, we apply each operator once and create n sequences denoted by the points $(1, 0, \ldots, 0), \ldots, (0, \ldots, 0, 1)$ with costs $\{D(S, \langle O_i\rangle(S))\}_{i=1}^{n}$ of applying operators $O_i,\ 0 \le i \le n$ respectively on the original image S. Next, we store the cost of sequences $\sigma(2, 0, \ldots, 0), \ldots, \sigma(0, \ldots, 0, 2)$, but for each sequence containing the application of two distinct operators $O_i, O_j, i \ne j$ we have 2 possible sequences: $\sigma = \langle O_i, O_j\rangle$ or $\sigma = \langle O_j, O_i\rangle$. We check the two options, and keep only the one whose cost is smaller in the table at position $(\ldots, 0, 1, 0, \ldots, 0, 1, 0, \ldots)$, where the 1s appear in positions i and j.

In general, to fill the entry $(p_1, \ldots, p_n)$ we examine all its predecessor sequences where the application of one of the operators was less by one. These correspond to points where one of the coordinates is less by one, which were already calculated and stored in the table. Denote them for abbreviation by $\sigma_i = \sigma(p_1, \ldots, p_i - 1, \ldots, p_n),\ 1 \le i \le n$. We append the operator O_i to sequence σ_i to get the new operator sequence denoted by $\langle \sigma_i \cup O_i\rangle$, apply this new sequence to the original image and choose the best one:

$$i^* = \arg \min_{1 \le i \le n} D(S, \langle \sigma_i \cup O_i\rangle(S))) \qquad (2)$$

The table structure is an n-dimensional simplex that is constructed in m stages. For example in Figure 9 the table is an equilateral triangle, which is a 2-simplex. In practice, we sample the search space in lower rates than 1 pixel (usually 5 or 10 pixels), meaning we apply each operator more than once between stages. At the last stage, all points $(p_1, \ldots, p_n)$ where $\sum_{i=1}^{n} p_i = -m$ represent target images of size $w' = w - m$. We choose the one that stores the smallest cost. To obtain the optimal sequence of retargeting operators $\langle O_{i_1}, \ldots, O_{i_m}\rangle$ we backtrack to the first entry and in each step recover the operator that had been chosen. The time and space complexities of the algorithm are $O(m^n)$ which is polynomial in the amount of size change, but exponential in the number of operators to be used. This approach can also be used to combine width and height operations in two directions.

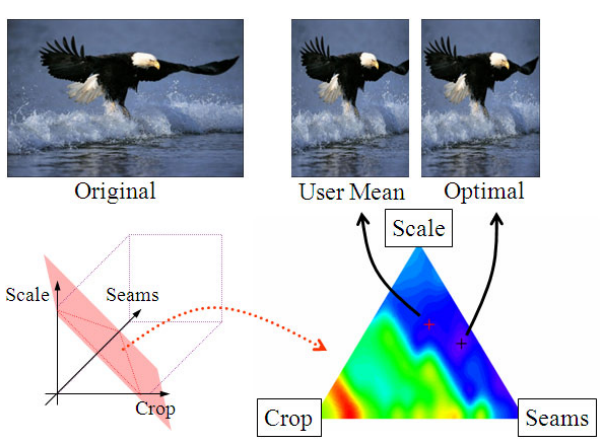
Subject to the assumptions outlined above, our discrete optimization is guaranteed to find the optimum multi-operator sequence. However, these assumptions mean we only search in a sub-space of the resizing space and do not reproduce all images of the desired target size. There might exist a retargeted image that is more similar to the source image. Furthermore, the definition of best results may change depending both on the user and on the goal for retargeting the image. In our study (Section 3) we found that all users tend to use a combination of operators (Figure 2). However, the variance between users choices was very large ($\sigma \approx \frac{1}{2}\mu$). Therefore, in addition to the automatic optimization method we present an interactive technique that allows users to explore a sub-space of retargeting possibilities in a simple manner.

7 Interactive Multi-Operator Retargeting

Recall that *regular* paths fix the order of operators ahead of time and can conveniently be written as: $\langle k_1 \times O_{i_1}, \ldots, k_n \times O_{i_n}\rangle$, where $\sum_{j=1}^{n} k_j = m$ (see e.g. Figure 9(d)-(f)). So the only question is how much does each operator contribute to the image retargeting process? This creates a one-to-one mapping between a point in the resizing space and a *regular* path. Fixing m is equivalent to choosing a hyper-plane in the resizing space, and choosing how much each operator will contribute corresponds to choosing a point on this hyper-plane. This means that we can find the optimal solution for this problem using exhaustive search in $O(m^{n-1})$, which is polynomial in the size change m but exponential in the number of operators n. Since n is usually small (say, three or four operators) and we can sample m in discrete steps, this search is feasible.

Regular paths also lend themselves to a simple interface that assists users search in this sub-space for desired results. First, the order of operators in the sequence is chosen ahead of time (e.g. $\langle k_1 \times SC, k_2 \times CR, k_3 \times SL\rangle$ or $\langle k_1 \times SL, k_2 \times SC, k_3 \times CR\rangle$). Next, there is a slider governing the image size change m, and a slider for each operator separately. Since the contribution of all operators must sum to the total size change m, users must choose a coupling of a pair of sliders to change their value. Moving one in a positive direction will drive the other to move in the negative direction, and vice versa.

Note that such interface enables using both positive and negative amounts of specific operators. For example, we can bound the contribution of each operator O_{i_j} to $m \ge k_j \ge -m$ such that $\sum_{j=1}^{n} k_j = m$. In our user study of Section 3, we constrain $sign(k_j) = sign(m)$ and fix the order of operators to be $\langle k_1 \times SC, k_2 \times CR, k_3 \times SL\rangle$ to define a simpler interface us-



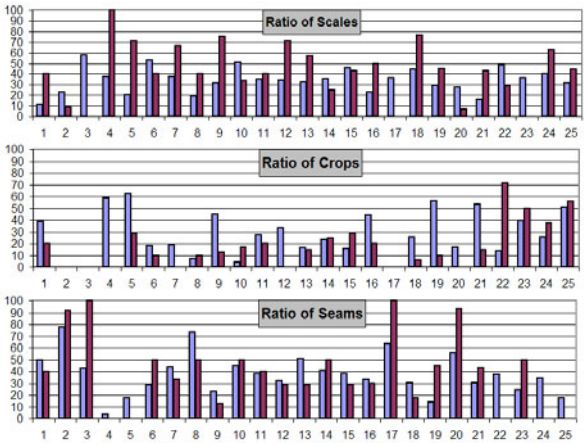


Figure 10: Using *regular* paths we are limited to searching on a plane in resizing space (left). We find the optimal multi-operator resizing sequence having the minimum BDW distance (0.241 in this case). The distance throughout the search space is colored from blue (small distance) to red (large distance) and was interpolated for visualization purposes. We compare our results to the results and score of the mean of the user study (where BDW = 0.355). On the right we show a summary of the comparison of the ratios of all results. Blue is the mean of the user study and Red is our results using optimization with BDW. The average difference is around 20%, well within the standard deviation of the user study which is 50%. More image results can be found in the supplemental material.

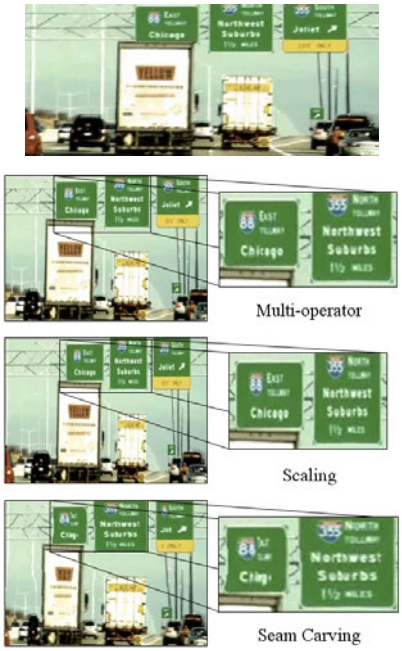


Figure 11: An example of a video key-frames where multi-operator retargeting achieves better results than scaling or seam carving.

ing just one slider for all operators (see accompanying video). This constrains the retargeting to *regular-monotonic* sequences, and further confines the search space to the intersection of the hyperplane with the axes (Figure 10). Although somewhat limiting, users found this method to be intuitive and productive.

8 Retargeting Video

We extend multi-operator image retargeting to video using key frames. For each key-frame, we find the best *regular* path. This defines a set of multi-operator sequences constraints for k time-steps. In-between the key-frames, we interpolate the amount of each operator in the sequences to obtain the multi-operator sequence for each intermediate frame. For scaling, this interpolation is trivial since we just change the scale factor linearly. Seam carving also supports linear interpolation between different amounts by interpolating the number of seams to be removed. For cropping, we separate the amount of cropping in each key-frame to the left and right cropping (or top and bottom) and interpolate linearly between each one separately.

Such sequence interpolations can sometime insert virtual camera motion into the resulting video. For example, interpolating between cropping k columns on the left to cropping k columns on the right introduces a *panning* effect, while interpolating between different scaling levels may introduce a *zoom in/out* effect. Some example results can be found in Figure 11 and the supplemental video.

9 Results

To validate our optimization and BDW similarity measure we compared the mean results of the user study to results obtained by our optimization on *regular* paths (Figure 10). The average difference between the automatic and mean user choice is 20%, which is well within the standard deviation of about 50% in the user choices. Figure 10 also illustrates that the visual results are comparable (more image results can be found in the supplemental material). Moreover, the mean user study result usually does not differ much in terms of the BDW score from the best score (i.e. our result). This demonstrates the effectiveness of the BDW measure itself.

Our multi-operator framework supports various scenarios for finding the optimal combination for retargeting. We illustrate this by showing results of a number of cases where we change the set of operators and also the image similarity measure used. Figure 13 illustrates an example of optimally combining seam carving and scaling in a *regular* sequence. We find the best transition point between applying seam carving and scaling by measuring the BDW of the results. In Figure 15 we show results of computing the optimal *mixed* sequences using two operators (seam carving and scaling) by dynamic programming using the BDW score. The teaser figure (Figure 1) shows the result of finding the optimal *mixed* sequence

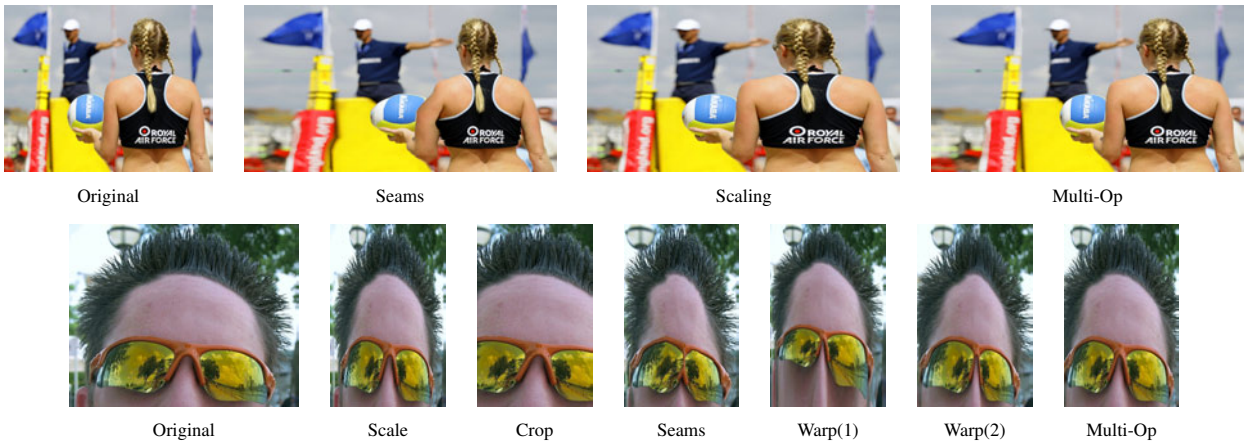


Figure 12: Comparison of retargeting results (expansion and reduction) using various image retargeting methods. These examples illustrate cases where using optimized multi-operator retargeting combining seam caving, cropping and scaling achieves better results. Seams is using seam carving from [Rubinstein *et al.* 2008], Warp(1) and Warp(2) are non-homogeneous scaling from [Wang *et al.* 2008] and [Wolf *et al.* 2007] respectively.

for changing the width of an image consisting of three operators (scaling, cropping and seam carving), subject to the BDW image similarity measure. In Figure 14 we find the best *mixed* path using the bidirectional similarity measure [Simakov et al. 2008] and four image retargeting operators (horizontal and vertical scaling and seam carving). As can be seen, the horizontal dimension is retargeted mainly with seam carving while the vertical dimension is mainly scaled. Figure 12 shows a comparison between several retargeting methods and our multi-operator results.

In some cases, the optimal multi-op result might reduce to a single operator (e.g. scaling). However, we should note that this reduction is achieved automatically by the algorithm in an *informed* manner. This is exactly the purpose of the suggested system. A good result is not necessarily one that utilizes all available operators, but rather one that achieves higher similarity of the required size. In particular, by attempting to use cropping, scaling and seam-carving, the algorithm chose the scaling approach for more structured media, as seam carving tends to insert artifacts in such cases, and cropping might remove too much important information. In fact, we deliberately used images that are difficult cases for non-uniform operators (such as seam carving) to test if our method can recognize this automatically.

In Figure 11 we show an example of a key-frame from a video that demonstrates why multi-operator retargeting provides the flexibility to achieve better results than a single operator in video. Lastly, our framework also enables utilizing a simple user interface for combining *regular* multi-operator sequences in an intuitive manner. In the accompanying video we also show several interactive sessions for image retargeting, and more retargeting results for images and video. Taken together, these results show that our multi-operator algorithm can combine multiple operators together using various similarity measures (e.g. BDW or BDS), various paths (either *regular* or *mixed*) and various operators (horizontal and vertical, seam carving, scaling and cropping).

All results were created either on a 1.8 GHz dual core laptop with 2GB memory or on a 2.2 GHz dual core desktop with 4GB memory. Several processing time statistics for computing the BDW at different patch sizes are detailed in Table 1. The average optimization times were around 2 minutes for 2-operator regular paths, 10 minutes for 2-operator mixed paths and 3-operator regular paths, and 20 minutes for 4-operator mixed paths. As for the interactive

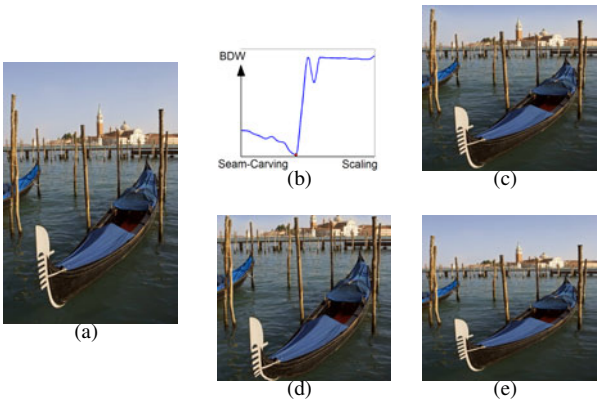
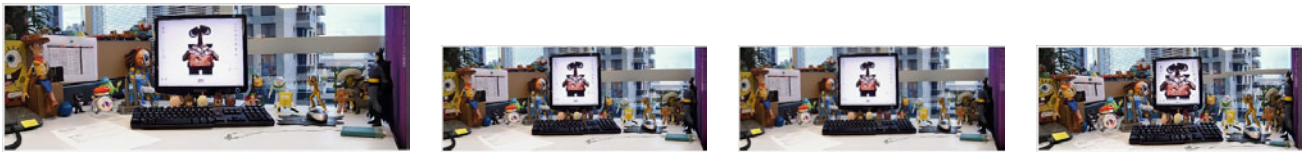


Figure 13: We find the optimal *regular* path by finding the minimal BDW score (the red dot in (b)) for a combination of two operators (seam carving and scaling in this case) to retarget an image (a). Compare the resulting image (c) to using just seam carving (d) or just scaling (e).

interfaces, in most cases (as seen in the video) the interaction is performed in real time. There are waiting periods, for instance, when there is a switch in the direction of size change between height and width due to recalculation of the seams. For video, once seam carving has been pre-computed on the video frames, the video playback is instant. We store just the interpolation values of the operators in each frame and by keeping the ratio between the operators constant during resize, we can interactively change video size as well.

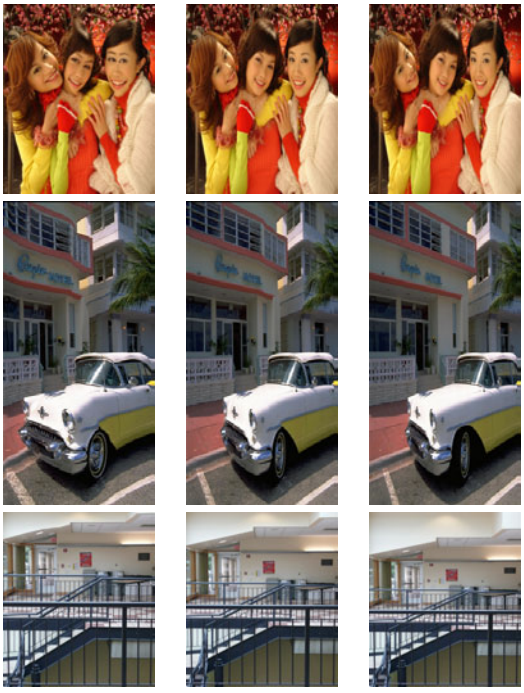
Patch size	Pixels	4×4	8×8	16×16	32×32
Max. overlap	–	5	3.2	2.4	0.9
No overlap	7.5	3.1	1.5	1.1	0.4

Table 1: Average calculation times (seconds) of BDW for the examples used in this paper. Maximum overlap means patches are taken in 1-pixel steps, and with no overlap means patches are disjoint. The larger the overlap the better the alignment of images.



Input Multi-op Scaling Seam Carving

Figure 14: Result of 2D retargeting. In this case we find the optimal *mixed* path using the bidirectional similarity measure and a combination of four retargeting operators (horizontal and vertical seam carving and scaling). The multi-operator result finds the best result by mainly applying seam carving in the horizontal dimension (compare the size of the monitor in the different methods) and scaling in the vertical dimension (look at the bottom of the desk and the face of Woody on the left). For comparison, we show the result of applying a uniform 2D scaling, or seam carving (running horizontal seam carving first, followed by vertical seam carving).



Seam Carving Multi-op Scaling

Figure 15: A comparison between seam carving (left), Multi-operator (center) and scaling (right). The multi-operator algorithm uses the BDW image similarity measure and finds the best *mixed* path using two image retargeting operators (seam carving and scaling).

10 Limitations

Although the number of operators is usually small, the optimization is still exponential in the number of operators. The overall complexity of the algorithm is measured by the number of entries in the dynamic programming table that we need to fill. In each iteration we need to apply one operator and compute the bidirectional warping, which is the major bottleneck as can be seen in Table 1. Still, in the context of retargeting it is asymptotically faster than [Simakov et al. 2008]. Additional optimizations may be considered such as early termination of the BDW evaluation in case the images are too different, or sampling of the resizing space in coarser resolution (which in our experiments still produces good results).

Figure 10 and the supplemental material show that our automatic results do not always agree with the users' preference. Still, one must remember that the users also did not agree on the "correct" result and our ground truth was just the barycenter of all users choices. Lastly, it is clear that the suggested algorithm is only as good as the operators used. Limitations imposed by the specific methods (cropping, scaling, seam-carving) will also be carried over to our solution. Towards this end we advocate that using a combination of several operators could alleviate limitations of specific ones.

11 Conclusions and Future Work

We presented an algorithm for combining multiple retargeting operators. We defined the *resizing space* as a space combining several resizing operators. We presented an algorithm to find the optimal path in resizing space, given a global objective function that measures the similarity between the source and target images. We further proposed the Bi-Directional Warping (BDW) function to measure this similarity. Remarkably, all levels of our algorithm benefit from dynamic programming. It is used to compute Seam Carving, used to compute a-symmetric alignment for BDW image similarity measure and finally, it is the basis of the algorithm to find the optimal multi-operator path. Our approach was tested on a large number of images and videos, many of which were difficult cases for previous single retargeting operators. We also validated our results by comparing them with ground truth data, collected in the user study. In addition, we described a simple and intuitive user interface to interactively explore the resizing space and achieve high quality results.

The BDW measure we presented is best suited to changes applied to the image in one direction. However, BDW can be extended to match 2D modifications in some cases by recursively applying asymmetric-DTW on the rows (or columns) of the image (Appendix A). In the future we plan to investigate other applications for BDW, as well as possible extensions to 2D and 3D. We also intend to combine other types of operators into the multi-operator framework. Finally, there are further ways to utilize the data gathered from the users and develop other similar experiments.

References

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

CHEN, B., AND SEN, P. 2008. Video carving. In *Short Papers Proceedings of Eurographics*.

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

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

KEYSERS, D., AND UNGER, W. 2003. Elastic image matching is np-complete. *Pattern Recogn. Lett. 24*, 1-3, 445–453.

LIU, F., AND GLEICHER, M. 2006. Video retargeting: automating pan and scan. In *Proc. of the 14th annual ACM international conf. on Multimedia*, ACM, 241–250.

LIU, H., XIE, X., MA, W., AND ZHANG, H. 2003. Automatic browsing of large pictures on mobile devices. *Proceedings of the 11tn ACM international conf. on Multimedia*, 148–155.

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

SAKOE, H. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing 26*, 43–49.

SALVADOR, S., AND CHAN, P. 2007. Toward accurate dynamic time warping in linear time and space. *Intell. Data Anal. 11*, 5, 561–580.

SANTELLA, A., AGRAWALA, M., DECARLO, D., SALESIN, D., AND COHEN, M. 2006. Gaze-based interaction for semi-automatic photo cropping. In *ACM Human Factors in Computing Systems (CHI)*, 771–780.

SETLUR, V., TAKAGI, S., RASKAR, R., GLEICHER, M., AND GOOCH, B. 2005. Automatic image retargeting. In *In the Mobile and Ubiquitous Multimedia (MUM)*, ACM Press.

SIMAKOV, D., CASPI, Y., SHECHTMAN, E., AND IRANI, M. 2008. Summarizing visual data using bidirectional similarity. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*.

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 Press, 95–104.

UCHIDA, S., AND SAKOE, H. 1998. A monotonic and continuous two-dimensional warping based on dynamic programming. In *ICPR '98: Proceedings of the 14th International Conference on Pattern Recognition-Volume 1*, 521.

WANG, Y.-S., TAI, C.-L., SORKINE, O., AND LEE, T.-Y. 2008. Optimized scale-and-stretch for image resizing. *ACM Trans. Graph. (Proceedings of ACM SIGGRAPH ASIA 27*, 5.

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)*, 1–6.

A Two-dimensional Asymmetric-DTW

The framework presented in the paper supports operators that resize images in one direction (either horizontally or vertically), yet, operators may distribute changes along two dimensions [Wang et al. 2008]. To measure such changes there is a need to develop a two-dimension BDW. The two-dimensional version of DTW is commonly known as *Dynamic Planar Warping* (DPW), and its definition is similar to its one-dimensional counterpart. Unfortunately, this problem was shown to be NP-complete [Keysers and Unger 2003], and several approximation methods have been proposed [Uchida and Sakoe 1998].

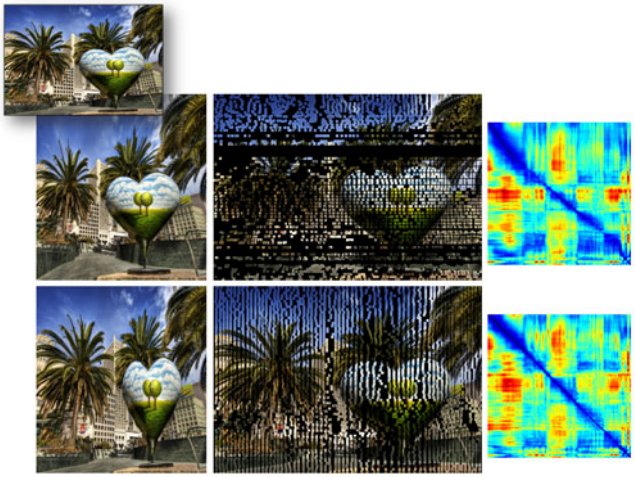


Figure 16: Examples of 2D A-DTW. Alignments of the Scale-and-Stretch result (top) and non-homogeneous warping result (bottom) to the original image (upper-left) using 4×4 patches. On the right, the pairwise 1D A-DTW row distances are shown for each result, colored from blue (small distance) to red (large distance). Notice that as the Scale-and-Stretch operator deforms the image in both directions, the optimal alignment inserts gaps in some rows, while for non-homogeneous warping, which works only along the resized dimension, every row in the retargeted image is matched to its corresponding row in the source.

A possible extension of our one-dimensional solution is to align the images using recursive evaluations of Asymmetric-DTW. That is, assume w.l.g we change the width of an image, and using our previous notation, we apply A-DTW on the signals $\{S_i\}_{i=1}^{h}$ and $\{T_i\}_{i=1}^{h}$ representing the rows of S and T respectively, taking $d(S_i, T_j) = \text{A-DTW}(S_i, T_j)$. This results in a two-dimensional order-preserving mapping between the two images that is optimal under a rigid row-to-row alignment. Although this method will not estimate correctly all possible transformations, we found it to produce good approximation for assessing image similarity (Figure 16). The running time of this algorithm is $O(h^2 w^2)$ using naive implementation, but can be further optimized using the techniques mentioned in Section 5.3.

Acknowledgements

We thank the anonymous SIGGRAPH reviewers for their comments. We thank Maya Yaniv for narrating our video. We thank the flickr members who have kindly made their media available for research purposes via the creative commons license: Ben Harris-Roxas (fishing), danorbit (desk), david.bunting (volleyball), g_magnan (italy), Greg Gladman (church, wheels), i am indisposed (snow), iboy daniel (mnm), Pandiyan (pond), romainguy (surfers), thomas23 (glasses), van swearingen (orchid), etrusia_uk (Bodiam castle). We also thank the users of publicdomainpictures.net and morguefile.com who have shared their images through public domain (tiger, eagle, stairs, islands). The Taj Mahal image is courtesy of ictopon2009.uwo.ca. The San Francisco heart image and results were borrowed from [Wang et al. 2008]. The bicycle, Buddha, car, malibu, foliage, face, mochizuki, venice and waterfall images are borrowed from [Avidan and Shamir 2007]. The osaka image and highway video are taken from [Rubinstein et al. 2008]. The birds video sequence is a snipped from "for the birds", courtesy of Disney/Pixar.