Summarizing Visual Data Using Bidirectional Similarity

Denis Simakov Yaron Caspi Eli Shechtman* Michal Irani

The Weizmann Institute of Science, Rehovot, ISRAEL
Adobe Systems Inc. & University of Washington
The Weizmann Institute of Science, Rehovot, ISRAEL

Abstract

We propose a principled approach to summarization of visual data (images or video) based on optimization of a well-defined similarity measure. The problem we consider is re-targeting *(or summarization) of image/video data into smaller sizes. A good "visual summary" should satisfy two properties: (1) it should contain as much as possible visual information from the input data; (2) it should introduce as few as possible new visual artifacts that were not in the input data (i.e., preserve visual coherence). We propose a bi-directional similarity measure which quantitatively captures these two requirements: Two signals S and T are considered visually similar if all patches of S (at multiple scales) are contained in T, and vice versa.*

The problem of summarization/re-targeting is posed as an optimization problem of this bi-directional similarity measure. We show summarization results for image and video data. We further show that the same approach can be used to address a variety of other problems, including automatic cropping, completion and synthesis of visual data, image collage, object removal, photo reshuffling and more.

1. Introduction

Given a large image/video, we often want to display it in a different (usually smaller) size – *e.g.*, for generating image thumbnails, for obtaining short summaries of long videos, or for displaying images/videos on different screen sizes. This smaller representation (the *visual summary*) should faithfully represent the original visual appearance and dynamics as best as possible, and be visually pleasing.

The simplest and most commonly used methods for generating smaller-sized visual displays are *scaling* and *cropping*. Image scaling maintains the entire global layout of the image, but compromises its visual resolution, and distorts appearance of objects when the aspect ratio changes. Cropping, on the other hand, preserves visual resolution and appearance within the cropped region, but looses all visual information outside that region.

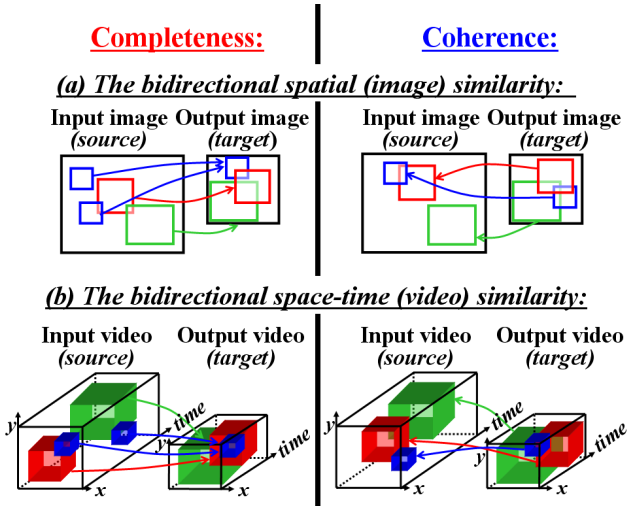

Figure 1. **The bidirectional similarity (Completeness + Coherence)**. *Two signals are considered visually similar if all patches of one signal are contained in the other signal,* and vice versa. *The patches are taken at multiple scales: spatial scales in the case of images (a), space-time scales in the case of video sequences (b).*

More sophisticated methods have been proposed for automatic retargeting by reorganizing the visual data (image or video) in a more compact way, while trying to preserve visual coherence of *important (usually sparse) regions* [3, 8, 10, 11, 12, 14, 15, 16, 17]. These methods can roughly be classified into three families: (i) *Importance-based scaling methods* [10, 14, 16] first identify important regions within the image (*e.g.*, salient regions, faces, high-motion regions). The outputs of these methods are characterized by scaling-down of unimportant regions (*e.g.*, the background), while the important regions are preserved as close as possible to their original size (*e.g.*, foreground objects). Nice results are obtained when there are only a few "important" objects within an image. However, these methods reduce to pure image scaling in case of uniform importance throughout the image. (ii) *Importance-based cropping methods* [3, 11, 15] provide nice results when the interesting information is concentrated in one region (spatial or temporal). (iii) *Importance-based bin-packing meth-*

*This research was done when the author was at the Weizmann Institute.

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

ods [8, 12, 14, 17] further account for the main deficiency of cropping – the inability to capture spatially or temporally separated objects – by compact packing (spatial and/or temporal) of segmented important/salient regions/blobs.

Importance-based methods require the important regions to be relatively compact and sparse within the visual data. In contrast, the elegant "Seam Carving" approach [1] does not rely on compactness/sparseness of important information. It removes uniform regions scattered throughout the image, by carving out vertical and horizontal pixel-wide seams with low gradient content. It beautifully shrinks images as long as there are enough low-gradient pixels to remove. Our experiments suggest, however, that when the image gets too small (i.e., all low gradient pixels have been removed), or when the interesting object/s span the entire image, "Seam Carving" deforms important image content.

An image retargeting example was also shown in a concurrent work [5]. It is based on choosing patch arrangements that fit together well. However, this method does not impose completeness of the visual data, i.e., does not require that all source patches/information be represented in the output. As such, it is not designed to generate complete/faithful visual summaries. Note that neither "Seam Carving", nor "importance based summarization", nor [5], exploit repetitiveness or redundancy of visual data in the retargeting/summarization process.

In this paper we propose a measure for quantifying how "good" a visual summary is. Such a measure is useful for two purposes: (i) As an objective function within an optimization process to generate good visual summaries; (ii) To quantitatively compare and evaluate visual summaries produced by different methods.

The proposed similarity measure is simple yet intuitive, and can be used to compare two images or two videos *of different sizes*. We say that T is a good visual summary of S if both: (1) T is visually *complete* w.r.t. S (i.e., T represents all the visual data in S); and (2) T is visually *coherent* w.r.t. S (i.e., T does not introduce new visual artifacts that were not observed in S). These two requirements are formulated and captured in our *patch-based bi-directional similarity measure* (Fig. 1): Two signals S and T are considered visually similar if as many as possible patches of S (at multiple scales) are contained in T, *and vice versa*.

We further show how this similarity measure can be used to solve the following problem: "Find a visual summary T (of user-defined dimensions) which maximizes the bi-directional similarity measure when compared to the input source S." We show results of applying this approach for generating visual summaries of images and videos, as well as for other applications (image montage, image synthesis, object removal, and auto-cropping). Our algorithm produces visually coherent small-sized summaries which are impossible to obtain with currently existing methods (since they do not exploit data redundancy). Moreover, we show how non-uniform importance can also be incorporated into our measure and optimization process, if desired.

The main contributions of this paper are therefore:
(i) A *bi-directional similarity measure* between visual data of different sizes (with or without importance weights).
(ii) A summarization/retargeting algorithm of image/video data, which optimizes this measure.
(iii) Application of this approach to a variety of problems: image summarization, image synthesis, image collage, photo reshuffling, object removal, and auto-cropping.

The rest of this paper is organized as follows: Sec. 2 formulates the bidirectional similarity measure between two signals. Sec. 3 presents the retargeting algorithm for generating visual summaries by optimizing this measure. Results, applications and comparison to other methods are provided in Sec. 4. Finally, Sec. 5 shows how non-uniform importance can be incorporated into our framework.

2. The Bidirectional Similarity Measure

We consider two signals S and T to be "visually similar" if as many as possible patches of S (at multiple scales) are contained in T, *and vice versa*. This bi-directional similarity is illustrated in Fig. 1, and is formulated below. Denote by S, T two visual signals of the same type (images, videos, etc.) In the case of visual summarization or retargeting, S will be the input *Source* signal, and T will be the output *Target* signal. S and T need not be of the same size: T may be smaller than S (data summarization), or larger than S (data synthesis). Let P and Q denote patches in S and T, respectively, and let N_S and N_T denote the number of patches in S and T, respectively.

We define the following error (**dis**similarity) measure:

$$d(S,T) = \overbrace{\frac{1}{N_S}\sum_{P\subset S}\min_{Q\subset T} D(P,Q)}^{d_{\text{complete}}(S,T)} + \overbrace{\frac{1}{N_T}\sum_{Q\subset T}\min_{P\subset S} D(Q,P)}^{d_{\text{cohere}}(S,T)} \quad (1)$$

Namely, for every patch $Q \subset T$ we search for the most similar patch $P \subset S$, and measure their distance $D(,)$, and vice-versa. The patches are taken *around every pixel* and *at multiple scales*, resulting in significant patch overlap. The spatial (or spatio-temporal) geometric relations are implicitly captured by treating images (or videos) as *unordered sets* of *all* their overlapping patches. The distance $D(,)$ in Eq. (1) may be any distance measure between two patches. In our current implementation we used SSD (Sum of Squared Distances), measured in CIE $L^*a^*b^*$ color space and normalized by the patch size.

We can further introduce different relative weights of the two terms in Eq. (1), depending on the application:

$$d(S,T) = \alpha \cdot d_{\text{complete}}(S,T) + (1-\alpha)\cdot d_{\text{cohere}}(S,T) \quad (2)$$

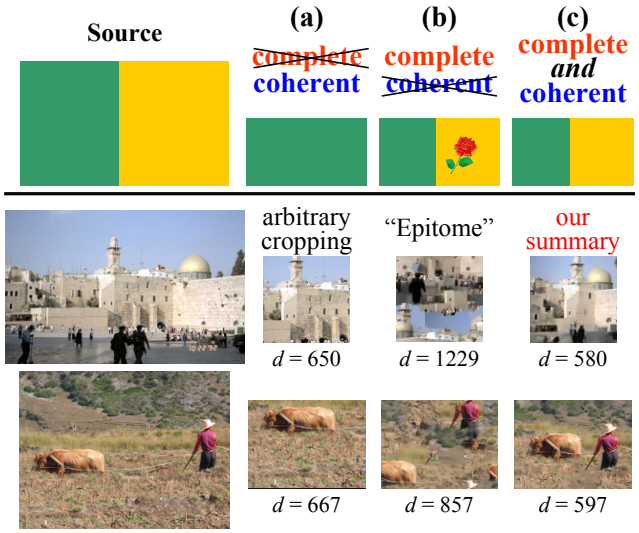

Figure 2. **Importance of "completeness" and "coherence".** ***Column (a)*** *shows images which are fully coherent w.r.t. their source image, but are not complete (*e.g.*, an arbitrary cropping – all its patches are contained in the source).* ***Column (b)*** *shows images that are fully complete w.r.t. the source, but are not coherent w.r.t. it (*e.g.*, "Epitome" [4, 7] – contains all patches from the source).* ***Column (c)*** *shows images that are both complete and coherent. The bidirectional* ***dis****similarity (error) measure $d(S,T)$ of Eq.* (1) *is displayed below each image.*

We used $\alpha = 0.5$ in all our summarization examples.

In order to capture bi-directional similarity *locally* and *globally*, the completeness and coherence terms need to be computed at *multiple scales*. In our current implementation, signals S and T are compared at multiple (corresponding) resolutions within a Gaussian pyramid. In each pyramid level, Eq. (1) is computed using patches of size 7×7 for images and $7 \times 7 \times 5$ (spatio-temporal) for video data.

While the two terms in Eq. (1) above seem very similar to each other, they have important complementary roles. The first term, $d_{\text{complete}}(S,T)$, *measures the deviation of the target T from "completeness" w.r.t. S*. Namely, it measures if all patches of S (at multiple scales) have been preserved in T (or how well S can be reconstructed from T). The second term, $d_{\text{cohere}}(S,T)$, *measures the deviation of the target T from "coherence" w.r.t. S*. Namely, it measures if there are any 'newborn' patches in T which have not originated from S (i.e., new undesired visual artifacts). These are the two properties we expect from a good visual summary: (i) to represent the input well (be *complete*), and (ii) to be visually pleasing (*coherent*).

Neither completeness nor coherence on its own suffices to provide a good visual summary, as illustrated in Fig. 2. Arbitrary image cropping (Fig. 2.a) provides a perfectly coherent image with respect to the input (all its patches can be found among the input patches), but it is clearly not a complete representation of the input image, hence not a good visual summary. To illustrate the effect of exploiting only completeness, we use the "Epitome" [4, 7] method. The "Epitome" results[1] in Fig. 2.b contains all the input patches (thus complete), but introduces new undesired visual artifacts (has additional patches which were not in the input image), thus being incoherent with respect to the input image. In this paper we show that combining these two constraints provides a useful measure for visual summarization. Fig. 2.c shows results of our algorithm, which optimizes both terms (completeness and coherence) simultaneously.

Each term *separately* had been previously employed for other purposes. The completeness term alone (when $\alpha = 1$ in Eq. (2)) resembles the objective function optimized in the "Epitome" work of [4, 7]. The coherence term alone (when $\alpha = 0$ in Eq. (2)) is similar to the objective function optimized in the data completion work of [18].

Other completeness or coherence measures have also been introduced. The "Jigsaw" work of [9], like "Epitome", generates a concise complete representation by learning non-regular shaped image parts – 'jigsaw pieces'. Although more coherent than "Epitome", its output is still scrambled. The similarity measure proposed by [2] can be regarded as maximizing coherence with respect to the input, but is indifferent to preserving completeness (it is a single-directional measure).

Although not psychophysically verified, our similarity measure is quite intuitive and simple to use for comparing images or video sequences of different sizes. Moreover, its simple mathematical formulation is convenient for analytical derivations, making it easy to use within an optimization algorithm, as will be shown next.

3. The Summarization (Retargeting) Algorithm

Given a source signal S, we want to reconstruct a target signal T (of pre-defined dimensions) that optimizes the similarity measure of Eq. (1) w.r.t. S. Formally, we search for T_{output} such that:

$$T_{output} = \arg\min_{T} d(S,T). \tag{3}$$

Below we describe the algorithm we use to solve this optimization problem. Sec. 3.1 shows how to update target pixel colors T in order to decrease $d(S,T)$ at each iteration (the *update rule*). Sec. 3.2 shows how to achieve good convergence of the iterative process (*coarse-to-fine gradual resizing* algorithm).

3.1. The Iterative Update rule

In this section we derive *an iterative-update rule in the color space* of pixels of the target T, that minimizes $d(S,T)$. Let $q \in T$ be a pixel in T, and let $T(q)$ denote its

[1] Code from `http://research.microsoft.com/~jojic/epitome.htm`.

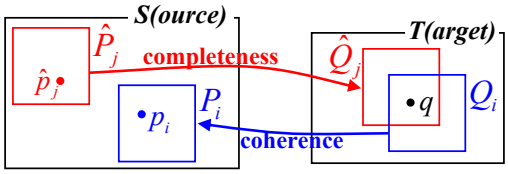

Figure 3. **Notations for the update rule.**

color. We first isolate the contribution of the color of each pixel $q \in T$ to the error $d(S,T)$ of Eq. (1).

The error a pixel $q \in T$ contributes to $d_{\text{cohere}}(S,T)$: Let $Q_1,\ldots,Q_m$ denote all the patches in T that contain pixel q (*e.g.*, if our patches are 7×7, there are 49 such patches). Let $P_1,\ldots,P_m$ denote the corresponding (most similar) patches in S (i.e., $P_i = \arg\min_{P \subset S} D(P,Q_i)$). Let $p_1,\ldots,p_m$ be the pixels in $P_1,\ldots,P_m$ corresponding to the position of pixel q within $Q_1,\ldots,Q_m$ (see Fig. 3). Then $\frac{1}{N_T}\sum_{i=1}^{m}(S(p_i)-T(q))^2$ is the contribution of the color of pixel $q \in T$ to the term $d_{\text{cohere}}(S,T)$ in Eq. (1).

The error a pixel $q \in T$ contributes to $d_{\text{complete}}(S,T)$: Let $\hat{Q}_1,\ldots,\hat{Q}_n$ denote all the patches in T that contain pixel q *and* serve as "the most similar patch" to some patches $\hat{P}_1,\ldots,\hat{P}_n$ in S (i.e., $\exists \hat{P}_j \subset S$ s.t. $\hat{Q}_j = \arg\min_{Q \subset T} D(\hat{P}_j,Q)$). Note that unlike m above, which is a fixed number for all pixels $q \in T$, n varies from pixel to pixel. It may also be zero if no patch in S points to a patch containing $q \in T$ as its most similar patch. Let $\hat{p}_1,\ldots,\hat{p}_n$ be the pixels in patches $\hat{P}_1,\ldots,\hat{P}_n$ corresponding to the position of pixel q within $\hat{Q}_1,\ldots,\hat{Q}_n$ (see Fig. 3). Then $\frac{1}{N_S}\sum_{j=1}^{n}(S(\hat{p}_j)-T(q))^2$ is the contribution of the color of pixel $q \in T$ to the term $d_{\text{complete}}(S,T)$ in Eq. (1).

Thus, the overall contribution of the color of pixel $q \in T$ to the global bidirectional error $d(S,T)$ of Eq. (1) is:

$$\text{Err}(T(q)) = \frac{1}{N_S}\sum_{j=1}^{n}(S(\hat{p}_j)-T(q))^2 + \frac{1}{N_T}\sum_{i=1}^{m}(S(p_i)-T(q))^2. \quad (4)$$

To find the color $T(q)$ which minimizes the error in Eq. (4), $\text{Err}(T(q))$ is differentiated w.r.t. the *unknown color* $T(q)$ and equated to zero, leading to the following expression for the optimal color of pixel $q \in T$ (the **Update Rule**):

$$T(q) = \frac{\frac{1}{N_S}\sum_{j=1}^{n} S(\hat{p}_j) + \frac{1}{N_T}\sum_{i=1}^{m} S(p_i)}{\frac{n}{N_S}+\frac{m}{N_T}} \quad (5)$$

This entails a simple iterative algorithm. Given the target image $T^{(l)}$ obtained in the l-th iteration, we compute the colors of the target image $T^{(l+1)}$ as follows:

1. For each target patch $Q \subset T^{(l)}$ find the most similar source patch $P \subset S$ (minimize $D(P,Q)$). Colors of pixels in P are votes for pixels in Q with weight $1/N_T$.

2. In the opposite direction: for each $\hat{P} \subset S$ find the most similar $\hat{Q} \subset T^{(l)}$. Pixels in $\hat{P}$ vote for pixels in $\hat{Q}$ with weight $1/N_S$.

3. For each target pixel q take weighted average of all its votes as its new color $T^{(l+1)}(q)$. (Color votes $S(p_i)$ are found in step 1, $S(\hat{p}_i)$ in step 2.)

3.2. Convergence by Gradual Resizing

As in any iterative algorithm with a non-convex error surface, the local refinement process converges to a good solution only if the initial guess is "close enough" to the solution. But what would be a good initial guess in this case? Obviously the "gap" in size (and hence in appearance) between the source image S and the target image T is too large for a trivial initial guess to suffice: A random guess would be a bad initial guess of T; Simple cropping of S to the size of T *cannot* serve as a good initial guess, because most of the source patches would have been discarded and would not be recoverable in the iterative refinement process; Scaling down of S to the size of T is *not* a good initial guess either, because the appearance of scaled-down patches is dramatically different from the appearance of source patches, preventing recovery of source patches in the iterative refinement process.

If, on the other hand, the "gap" in size between the source S and target T were only minor (*e.g.*, $|T| = 0.95|S|$, where $|\cdot|$ denotes the size), then subtle scaling down of the source image S to the size of T could serve as *a good initial guess* (since all source patches are present with minor changes in appearance). In such an "ideal" case, iterative refinement using the update rule of Eq. (5) would converge to a good solution.

Following this logic, our algorithm is applied through a *gradual resizing process*, illustrated in Fig. 4. A sequence of intermediate target images $T_0, T_1, \ldots, T_K$ of gradually decreasing sizes ($|S| = |T_0| > |T_1| > \ldots > |T_K| = |T|$) is produced, where $T_0 = S$. For each intermediate target T_k ($k = 1,\ldots,K$) a few refinement iterations are performed. The target T_k is first initialized to be a scaled-down version of the previously generated (and slightly larger) target T_{k-1}, i.e., $T_k^{(0)} := \mathit{scale\ down}(T_{k-1})$. Then iterative refinement is performed using the update rule of Eq. (5) until convergence is obtained for the k-th target size: $T_k^{(L)} = \arg\min d(S,T_k)$. This gradual resizing guarantees that at each intermediate output size ($k = 1,\ldots,K$) the initial guess $T_k^{(0)}$ is always close to the desired optimum of T_k, thus guaranteeing convergence at that output size. Note that the bidirectional distance measure $d(S,T_k)$ is always minimized *w.r.t. the original source image* S (and not w.r.t. to the previously recovered target T_{k-1}), since we want to obtain a final desired output summary $T = T_K$ that will minimize $d(S,T)$. An example sequence of im-

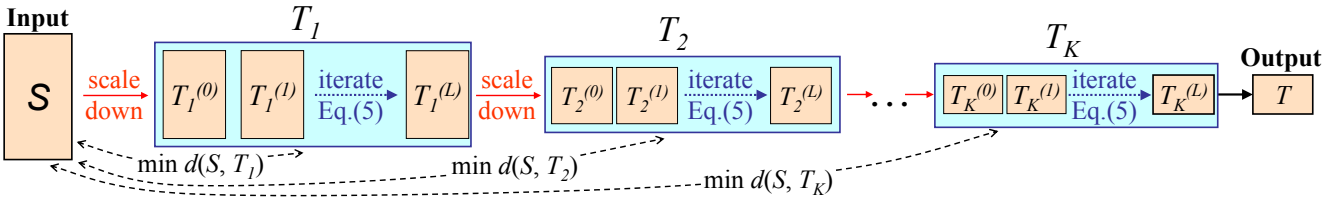

Figure 4. **Gradual resizing algorithm.** *Target size is gradually decreased until it reaches the desired dimensions. For each intermediate output T_k ($k = 1, \ldots, K$) the iterations $T_k^{(0)}, \ldots, T_k^{(L)}$ (inside blue rectangles) are initialized by scaling down the final result of the previous output: $T_{k-1}^{(L)}$. Fig. 5 shows intermediate outputs in such gradual resizing. Please view video in* `www.wisdom.weizmann.ac.il/~vision/VisualSummary.html`.

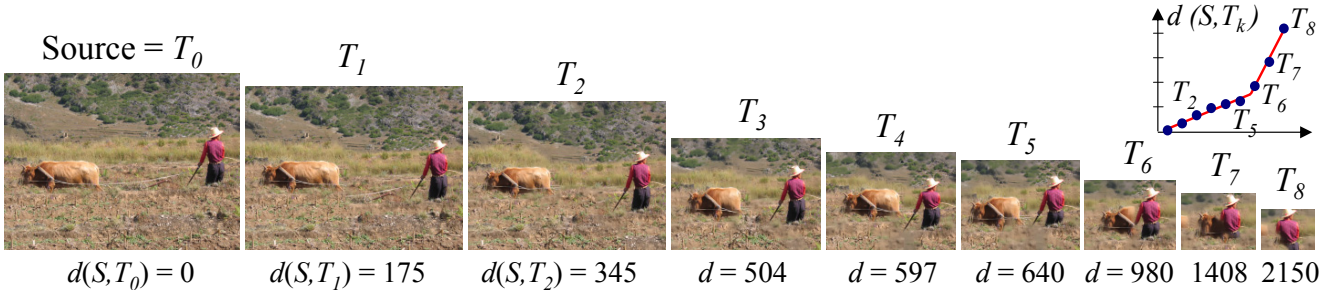

Figure 5. **Gradual image resizing.** *Results of our algorithm for gradually decreasing target size, with corresponding dissimilarity values $d(S,T)$ given by Eq.* (1)*. Note that when the target size is so small that we must compromise completeness (such as in T_6, T_7, T_8, where is not enough space to contain all source patches), the algorithm still preserves coherence. It selects the most coherent result among the equally incomplete ones.*

age summaries of gradually decreasing sizes is shown in Fig. 5. Please view video in `www.wisdom.weizmann.ac.il/~vision/VisualSummary.html`.

This gradual resizing procedure is implemented *coarse-to-fine* within a Gaussian pyramid (spatial in the case of images, spatio-temporal in the case of videos). Such a multi-scale approach allows to escape local minima and speeds up convergence, having a significant impact on the results.

4. Results and Applications

Our similarity measure (Sec. 2) and retargeting algorithm (Sec. 3) can be used for a variety of applications. Results are shown below for image/video summarization, image montage, image synthesis, photo reshuffling, and automatic cropping. Video results can be found in `www.wisdom.weizmann.ac.il/~vision/VisualSummary.html`.

Image Summarization: Figs. 5 shows results of our gradual resizing algorithm (Sec. 3) and the corresponding similarity measure $d(S,T)$ of Eq. (1). In the first few images, $T_1, ..., T_5$, the loss of visual information is gradual, accompanied by a slow increase in the dissimilarity measure $d(S,T)$. Starting from T_6 a significant amount of visual information is lost from image to image, which is also supported by a sharp increase in $d(S,T)$ (see graph in Fig. 5). This may suggest an automatic way to identify a good stopping point in the size reduction. Developing such a criterion is part of our ongoing work.

Fig. 6 presents results of image summarization obtained with our algorithm (Sec. 3) for several input images and for several different output sizes. It shows how the algorithm performs under different space limitations, down to very small sizes. These results are compared side-by-side with the output of the "Seam Carving" [1] algorithm[2].

Our algorithm exploits redundancy of image patterns (*e.g.*, the windows in the building) by mapping repetitive patches in the source image to a few representative patches in the target image (as in "Epitome" [4, 7]), thus preserving their appearance at the original scale. *"Seam Carving"* [1] removes vertical and horizontal paths of pixels with small gradients, nicely shrinking large images as long as there are enough low-gradient pixels to remove. It first removes uniform regions (such as the sky in the building image of Fig. 6), while maintaining faithful appearance of all objects in the image. However, when the image gets too small and all low-gradient pixels have been removed, "Seam Carving" starts distorting the image content. Moreover, since its decisions are based on *pixel-wide* seams, it does not try to preserve larger image patterns, nor does it exploit redundancy of such patterns. As such, it is less adequate for highly structured images (like images of buildings). Please view video on demo website, exemplifying these behaviors.

Our algorithm as described in Sec. 3 is significantly slower than "Seam Carving". However, the computationally heavy nearest-neighbor search may be significantly

[2]Code from `http://www.thegedanken.com/retarget`.

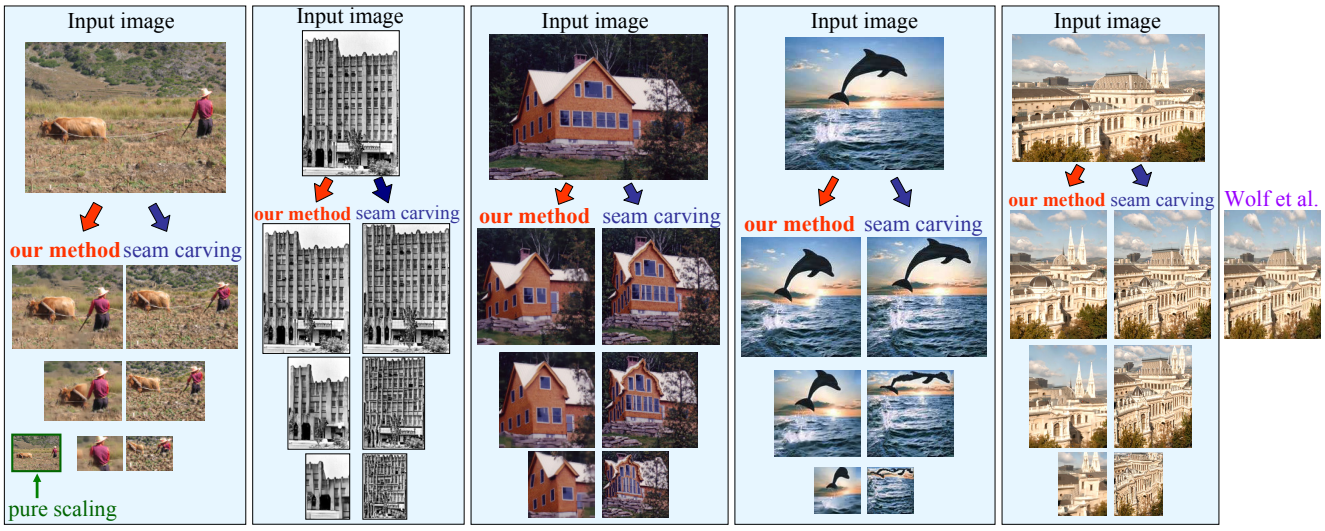

Figure 6. **Image summarization results.** *Our method exploits redundancies in the images (bushes, waves, windows of the buildings, etc.), often creating coherently-looking images even for extremely small target sizes. "Seam Carving" prefers to remove low-gradient pixels, thus distorts the image considerably at small sizes, when there are no more low-gradient pixels left. Please view video on demo website. The Dolphin image is from Avidan and Shamir [1], the right-most building image from Wolf* et al. *[16].*

sped up by using location from previous iteration to constrain the search. Our current implementation in Matlab takes 5 minutes to resize 250×200 to half its size, and scales linearly with the image size. Our algorithm can be parallelized on multiple CPU's/GPU for significant speedup.

After pure image scaling objects may become indiscernible if significantly scaled down, or distorted if the aspect ratio changes substantially from source to target (see an example in Fig. 6). The *importance-based retargeting* methods of [10, 14, 16] give very nice results when there are few "important" objects within an image, but reduce to pure image scaling in case of *uniform importance* throughout the image. For instance, in the right-most example in Fig. 6, the result of Wolf *et al*. [16] is very similar to pure image scaling. In Sec. 5 we further describe how non-uniform importance can also be incorporated into our summarization algorithm, and present comparison to Wolf *et al*. [16] on one of their examples containing faces.

Note that most existing summarization algorithms (*e.g*., [1, 14, 16, 3, 15, 10, 12, 8, 17]) *cannot* exploit repetitiveness or redundancy of the visual data. This leads to significant scaling down or distortion of image patterns by these methods when the target size gets very small.

Image/Video Montage. Image Tapestry/AutoCollage (*e.g*., [13]) and Video Montage (*e.g*., [8]) address the following problem: Given a set of inputs $S_1, S_2, ..S_n$ (images or videos), merge them into *a single output* T (image or video) in a seamless "sensible" way. Our algorithm of Sec. 3 can be applied to produce an image/video montage by taking multiple images/videos as a source (i.e., $S = \{S_1, S_2, ..S_n\}$). An example result is shown in Fig. 7.

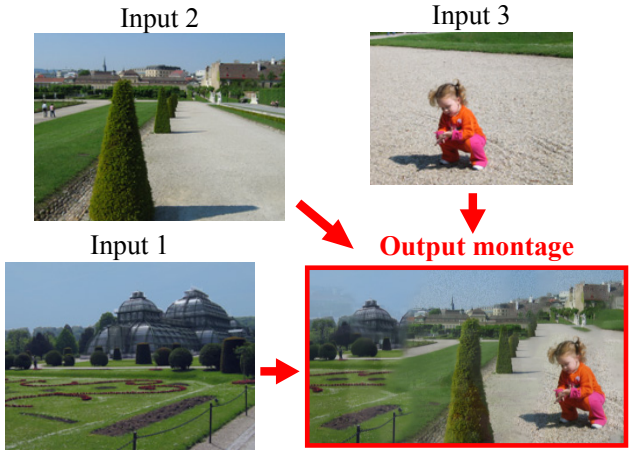

Figure 7. **Image montage result.** *Three different input images were automatically merged into a single output in a seamless coherent way. Simple side-by-side stacking of the three input images was taken as an initial guess, and then gradually resized to the target montage size using our algorithm from Sec. 3.*

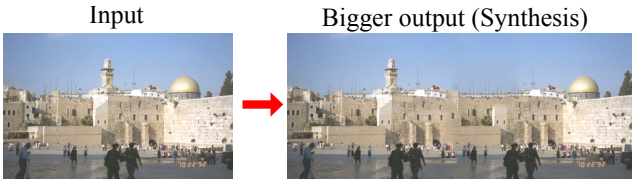

Figure 8. **Synthesis results.** *(See text for details.)*

Completeness guarantees that all patches from all input images are found in the output montage, while the *coherence* term guarantees their coherent merging. Note that no graph-cut nor post-processing blending has been used in this case.

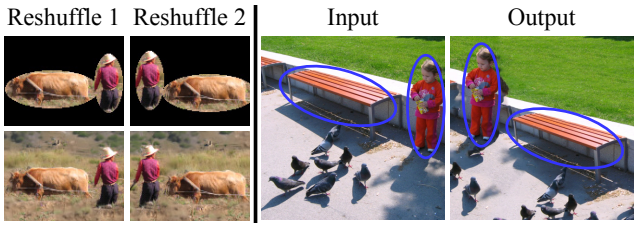

Figure 9. **Photo reshuffling.** *The user specifies new locations of objects (man and ox, girl and bench). The rest of the image is reconstructed in the most complete and coherent way.*

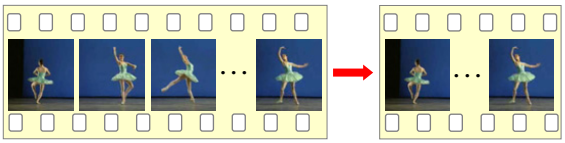

Figure 10. **Video summarization result.** *A complex ballet video sequence was summarized to half its temporal length by our algorithm (Sec. 3). Please view video on demo website.*

Image/Video Completion and Synthesis: In data synthesis/completion (*e.g.*, [6, 18]), the new synthesized regions must be visually coherent with respect to the input data. By setting $\alpha = 0$ in Eq. (2), our similarity measure reduces to the Coherence objective function of [18], and our optimization algorithm of Sec. 3 reduces to a synthesis algorithm similar to that of [18]. An example of an image synthesis result can be found in Fig. 8.

Photo reshuffling: We used our algorithm to reshuffle visual information according to user guidance (see Fig. 9). The user roughly marks objects and specifies their desired locations in the output. We then apply our similarity-optimization algorithm (Sec. 3) to fill in the *remaining* parts of the image in the most complete and coherent way. A different approach for photo-reshuffling appears in [5].

Video results: We further applied our optimization algorithm to generate a visual summary of video data (where S and T are video clips, and the patches are *space-time patches* at multiple space-time scales). We summarized a complex ballet video clip S by a shorter target clip T of half the temporal length – see Fig 10. The resulting output clip, although shorter, is visually coherent – it conveys a visually pleasing summary of the ballet movements at their original speed! On the other hand, it is also complete in the sense that it preserves information from all parts of the longer source video. Please view video on demo website, showing the input sequence and the resulting shorter output sequence (the video summary).

Automatic Cropping: The bidirectional measure described in Sec. 2 allows for automatic extraction of the best window to crop. Let S be the input image, and let T be the

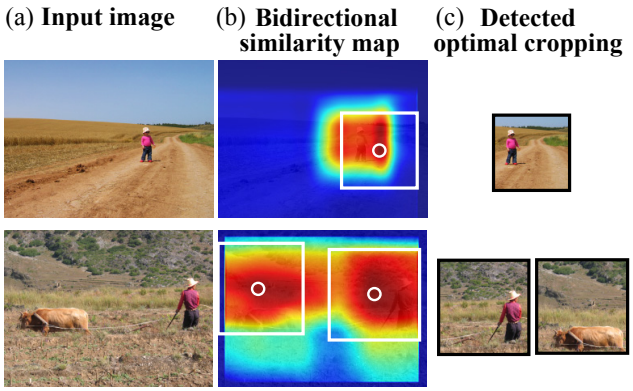

Figure 11. **Automatic optimal cropping.** *(a) Input images. (b) The bidirectional similarity measure maps color coded from blue (low similarity) to red (high similarity). The white circles mark the highest peaks and the white rectangles mark the corresponding best windows to crop. (c) The detected optimal cropping.*

(unknown) desired cropped region of S, of predefined dimensions $m \times n$. Sliding a window of size $m \times n$ across the entire image, we compute the bidirectional similarity score of Eq. (1) for each window, and assign it to its center pixel. This generates a continuous *bidirectional similarity map* for S – see Fig. 11.b. The peak of that map is the center pixel of the best window to crop in order to maintain maximal information (note that in this case only the "completeness" term will affect the choice, since all sub windows of S are perfectly "coherent" with respect to S). Often there is no single "good" place to crop (see lower row of Fig. 11). In that case the bidirectional similarity map contains multiple peaks, which can serve as multiple possible locations to crop the image (with their relative scores).

The same approach was used for temporal cropping in video data, using our bidirectional similarity measure with space-time patches. View video on the project website.

5. Incorporating Non-Uniform Importance

So far we assumed that all image regions are equally important. Often, however, this is not the case. For example, people are often more important than other objects. Such non-uniform importance is exploited in many retargeting and auto-cropping methods (*e.g.*, [8, 11, 12, 14, 15, 16]). Non-uniform importance can be incorporated into our bidirectional similarity measure by introducing "importance weights" in Eq. (1):

$$d(S,T) = \frac{\sum_{P \subset S} w_P \cdot \min_{Q \subset T} D(P,Q)}{\sum_{P \subset S} w_P} + \frac{\sum_{Q \subset T} w_{\hat{P}} \cdot \min_{P \subset S} D(Q,P)}{\sum_{Q \subset T} w_{\hat{P}}} \quad (6)$$

where w_P is the patch importance weight and $\hat{P} = \arg\min_{P \subset S} D(Q,P)$. Note that the weights in both com-

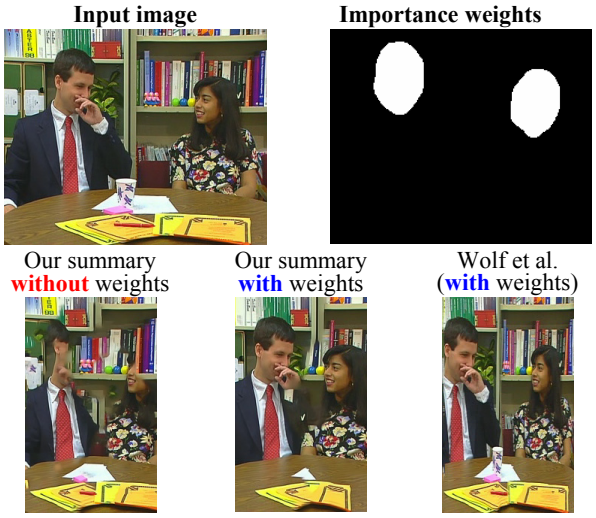

Figure 12. **Incorporating non-uniform importance.** *Without weights (left) our method prefers to preserve textured regions* (e.g., *books) over semantically important regions* (e.g., *faces). Importance weights on the faces solve this problem (center). Corresponding result of Wolf* et al. *[16] is shown on the right (the mask they used for this result may be different from ours). Note that the face of the girl has been preserved better with our method.*

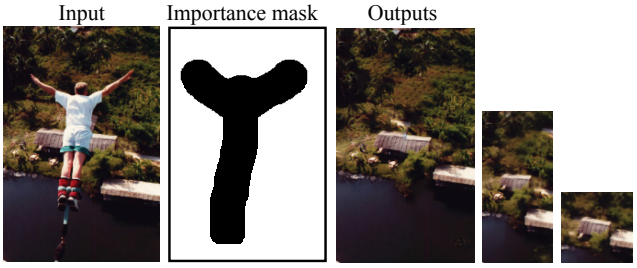

Figure 13. **Summarization with object removal constraints.** *White = high weights, black = low weights. This guarantees that the bungee jumper will not be included in the visual summary. No accurate segmentation is needed. Visual summaries of different sizes are shown.*

pleteness and coherence terms (w_P and $w_{\hat{P}}$, respectively) are defined over the source image.

Fig. 12 illustrates the contribution of using importance weights. Without importance weights, our method prefers the textured regions (*e.g.*, the books) over the relatively homogeneous regions (*e.g.*, the faces), which may be semantically more important. Introduction of importance weights solves this problem. Fig. 12 also includes comparison to the importance-based method of [16].

Importance weights can further be used to *remove/eliminate undesired objects* from the output image in the summarization process, as shown in Fig. 13.

6. Conclusion

We proposed a bidirectional similarity measure between two images/videos of *different sizes*. We described a principled approach to retargeting and summarization of visual data (images and videos) by optimizing this bidirectional similarity measure. We showed applications of this approach to image/video summarization, data completion and removal, image synthesis, image collage, photo reshuffling and auto-cropping.

Acknowledgments The authors would like to thank Lena Gorelick and Alex Rav-Acha for their insightful comments on the paper. This research was supported in part by the Israeli Ministry of Science.

References

[1] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *SIGGRAPH*, 2007.

[2] O. Boiman and M. Irani. Similarity by composition. In *NIPS*, pages 177–184, Vancouver, BC, Canada, 2007.

[3] L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H.-J. Zhang, and H.-Q. Zhou. A visual attention model for adapting images on small displays. *Multimedia Systems*, 2003.

[4] V. Cheung, B. J. Frey, and N. Jojic. Video epitomes. *IJCV*, Dec 2006.

[5] T. S. Cho, M. Butman, S. Avidan, and W. T. Freeman. The patch transform and its applications to image editing. In *CVPR*, 2008. To appear.

[6] A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *ICCV*, pages 1033–1038, 1999.

[7] N. Jojic, B. J. Frey, and A. Kannan. Epitomic analysis of appearance and shape. In *ICCV*, 2003.

[8] H.-W. Kang, Y. Matsushita, X. Tang, and X.-Q. Chen. Space-time video montage. In *CVPR (2)*, 2006.

[9] A. Kannan, J. Winn, and C. Rother. Clustering appearance and shape by learning jigsaws. *NIPS*, 2006.

[10] F. Liu and M. Gleicher. Automatic image retargeting with fisheye-view warping. In *UIST*, 2005.

[11] F. Liu and M. Gleicher. Video retargeting: Automating pan-and-scan. *ACM Multimedia 2006*, October 2006.

[12] A. Rav-Acha, Y. Pritch, and S. Peleg. Making a Long Video Short. In *CVPR*, June 2006.

[13] C. Rother, L. Bordeaux, Y. Hamadi, and A. Blake. Autocollage. *SIGGRAPH*, 25(3):847–852, 2006.

[14] V. Setlur, S. Takagi, R. Raskar, M. Gleicher, and B. Gooch. Automatic image retargeting. In *MUM*, 2005.

[15] B. Suh, H. Ling, B. B. Bederson, and D. W. Jacobs. Automatic thumbnail cropping and its effectiveness. In *UIST*, 2003.

[16] L. Wolf, M. Guttmann, and D. Cohen-Or. Non-homogeneous content-driven video-retargeting. In *ICCV'07*.

[17] Y.Pritch, A.Rav-Acha, A.Gutman, and S.Peleg. Webcam synopsis: Peeking around the world. In *ICCV*, 2007.

[18] Y.Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *PAMI*, 27(2), March 2007.