Automatic Image Retargeting

Vidya Setlur
Northwestern University and Nokia Research Center

Saeko Takagi
Wakayama University

Ramesh Raskar
Mitsubishi Electric Research Laboratories (MERL)

Michael Gleicher
University of Wisconsin, Madison

Bruce Gooch
Northwestern University

Figure 1: *Preserving functional realism rather than photo-realism by image retargeting. (a) The source image containing three areas of higher importance, the two boys, and the ball. (b) The source image retargeted to fit a PDA display. (c) The source image retargeted to fit a cell phone display. In the retargeted images, our algorithm is able to keep both boys in the image and maintain the relative positions of all shadows.*

Abstract

We present a non-photorealistic algorithm for *retargeting* large images to small size displays, particularly on mobile devices. This method adapts large images so that important objects in the image are still recognizable when displayed at a lower target resolution. Existing image manipulation techniques such as cropping works well for images containing a single important object, and down-sampling works well for images containing low frequency information. However, when these techniques are automatically applied to images with multiple objects, the image quality degrades and important information may be lost. Our algorithm addresses the case of multiple important objects in an image. The *retargeting* algorithm segments an image into regions, identifies important regions, removes them, fills the resulting gaps, resizes the remaining image, and re-inserts the important regions. Our approach lies in constructing a topologically constrained epitome of an image based on a visual attention model that is both comprehensible and size varying, making the method suitable for display-critical applications.

CR Categories: I.3.3 [Computer Graphics]: Picture/Image Generation—Display Algorithms;

Keywords: small displays, image adaptation, image attention, visual perception, non-photorealistic rendering, image cropping.

MUM 2005 Christchurch, New Zealand
ISBN 0-473-10658-2

1 Introduction

Increasingly, our computing and communications infrastructure is evolving to support images and video. Visual content is becoming more important for sharing, expressing, and exchanging information on devices such as, cell phones and hand-held PCs [Liu et al. 2003], PDAs with video capabilities, home-networked media appliances, and "heads up" informational displays in automobiles and helmets. Image retargeting is also useful for WYSIWYG directory icons for the efficient selection of images from directories and large image databases. Regardless of whether the resolution of the screen is high or whether the bandwidth is high, retargeting addresses the issue of displaying images on screen sizes with limited display real-estate.

Simply scaling images reduces the size of important features. If there is a single important feature in the image, the image can be cropped and scaled to fit. Images with multiple, important features present a more challenging case for retargeting. In such cases, valuable image area in the target image may be wasted with unimportant regions between important features. For example, in Figure 1 there are important features on both sides of the image and cropping cannot remove the unimportant area between them. For many images, the key content is a small set of objects. To effectively display such images on a small displays, these objects must be displayed at a sufficient size that they can be easily recognized. Other objects in the image, as well as the precise relationships between objects, are less important.

To assist in generating these increasingly important small images, we introduce a novel method for *Automatic Image Retargeting.* The goal is to provide *effective* small images by preserving the recognizability of important image features during downsizing. The premise of our method is that if the important objects in a given image can be identified, their size can be exaggerated in the target image such that they are more recognizable. Such exaggeration necessarily comes at the expense of realism: we intentionally distort less important parts of the image to make more important parts clearer. In contrast to traditional resizing and down-sampling techniques, the results of our retargeting method show important objects exaggerated in size, by reducing the spacing between objects, so that they are easier to recognize.

Contribution: The contribution of this work is an algorithm for retargeting images to small displays. When a large image is resized to a small display distortion is inevitable. Uniquely, this algorithm distorts images such that the important objects are all recognizable. The algorithm has application where recognizable small versions of large images are required, for example in creating thumbnails for image browsing and for displaying images on small devices such as PDAs and cellular phones.

2 Related Work

Image resizing can be performed manually using standard tools. Commercial products [Ado n. d.; Gim n. d.] enable the manual resizing of images using cropping and scaling operations. However, this process is often tedious, especially with large data sets. Cropping tends to work well for images containing single objects of importance. Scaling tends to distort important regions. Also, performing retargeting operations other than simple cropping and scaling requires a great deal of skill and effort.

A few researchers have explored automating image retargeting through automatic cropping processes. For example, Suh et al. [Suh et al. 2003] proposed two techniques for automatic cropping based on using a visual attention model to detect interesting areas in an image. Their first method is based on saliency maps [Itti et al. 1998], while the second is based on face detection [Rowley et al. 1996; Bregler 1998; Schneiderman and Kanade 2000]. In both cases the output is a thumbnail, created by cropping and scaling the source image to capture a single object. However, neither method can handle cases where there are multiple important features in an image.

There has been some work on image resizing that explicitly considers mobile devices. Chen et al. [Chen et al. 2002] and Liu et al. [Liu et al. 2003] have worked on image adaptation where the most important region is delivered to the client on the small screen. However, the user scrolls between 'pages' of an image to view different important regions and the multiple objects in the image cannot be seen at the same time on the screen.The method we propose, tends to maintain the recognizability and spatial relationships of multiple important regions in an image.

Jojic et al. [Jojic et al. 2003] proposed a model of image representation, called an *'epitome',* that attempts to encode the image's essence. This is similar to the retargeting goal. The epitome of an image is its miniature, condensed version containing most constitutive elements needed to reconstruct the original image. The main idea is to uniquely map every patch in the epitome to a corresponding patch in the original image. This works well when the original image contains several small, repetitive unit patterns. This technique would not be suitable for obtaining a more comprehensible image where the neighborhoods between important regions are required to be maintained. Epitomes do not always give an idea of the overall picture because neighborhood relationships between regions are significantly violated. The topology of an image is characterized by the neighborhood relationships among different elements that comprise the image. Our approach lies in constructing an epitome of an image that is topologically constrained, based on a visual attention model that is both comprehensible and size varying, suitable for display-critical applications.

The use of deformations to exaggerate portions of images and displays has been used in various techniques, as found in presentation literature. See Carpendale and Montagnese [Carpendale and Montagnese 2001] for a survey. Our method is the first to combine non-photorealistic deformation to an image retargeting application, and the cut-and-paste algorithm is uniquely suited to maintaining the recognizability of key image objects.

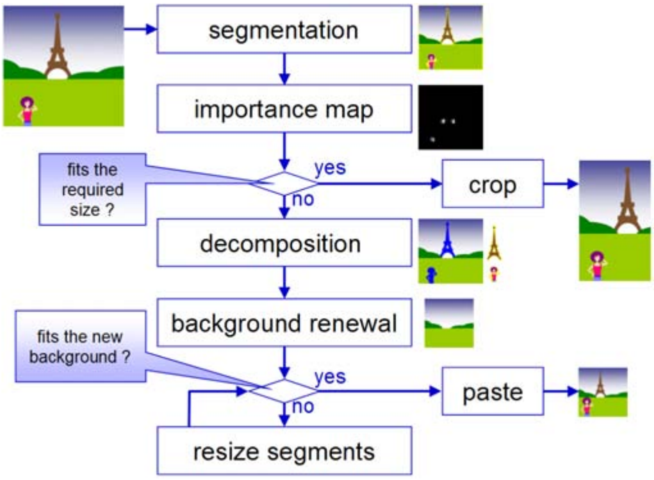


Figure 2: *Flowchart of our retargeting algorithm.*

3 The Retargeting Process

Our algorithm takes as input, a source image and a specification for the size of the output image. Figure 2 summarizes the algorithm. We first segment the source image into regions. We then use an importance map to select a set of *Regions of Importance* (*ROI*) to exaggerate in the result. Alternatively, the algorithm can be applied in a semi-automatic fashion by having the user specify the *ROI*. In Section 4 we discuss the techniques involved in segmenting the image, and combining adjacent regions based on their spatial distribution of color/intensity. In order to identify important regions, we generate an importance map of the source image using saliency and face detection as described in Section 5. If the specified size contains all the important regions, we simply crop the source image. Otherwise, we remove the important regions from the image, and fill the resulting "holes" using a background creation technique as described in Section 6. We then resize the updated background to fit the input specification. Regions of importance are then "pasted" back onto the updated background based on their importance, and relative topology within the scene. If all the important regions are not able to fit within the new image, we resize these regions inversely proportional to their importance. The "pasting" process is covered in detail in Section 7.

4 Image Segmentation

In order to identify important regions in the image, we must first segment the image. We use mean-shift image segmentation [Meer and Georgescu 2001] to decompose the given image into homogeneous regions (refer to Appendix A for more detailed explanation). The advantages of this approach include flexible modeling of the image and noise processes and consequent robustness in segmentation. The segmentation routine takes as input, the parameters:

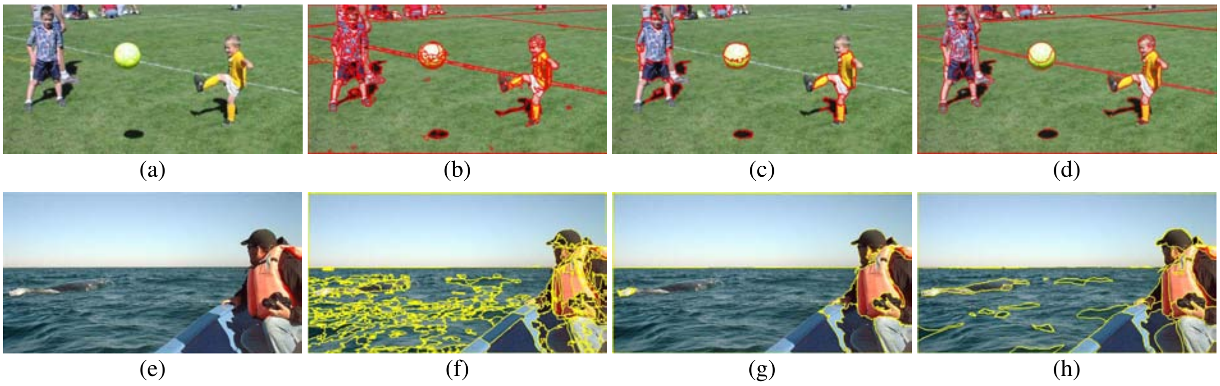


Figure 3: *Image segmentation. a) The original image. b) Applying mean-shift with parameters $h_s = 7$, $h_r = 6$, and $M = 50$. c) Applying mean-shift with parameters $h_s = 32$, $h_r = 30$, and $M = 150$. d) Performing region simplification on (b). e) The original image. f) Applying mean-shift with parameters $h_s = 7$, $h_r = 6$, and $M = 50$. g) Applying mean-shift with parameters $h_s = 22$, $h_r = 22$, and $M = 200$. h) Performing region simplification on (f).*

spatial radius h_s, color radius h_r, and the minimum number of pixels M that constitute a region. As with other segmentation methods, choosing optimal parameter values is often difficult. Therefore we over-segment the image using lower values of h_r and M and merge adjacent regions based on color and intensity distributions in a perceptually uniform color space, CIE-Luv. In practice, values of $h_s = 7, h_r = 6$, and $M = 50$, tends to work well for over-segmentation for most images.

We then compute a color similarity measure called histogram intersection [Swain and Ballard 1991] to determine color similarity between regions, and perform region simplification by merging adjacent regions. Figure 3 illustrates an example of this technique. Histogram intersection matches the image color histogram of a given segmented region with histograms of each of the adjacent regions. Given a pair of histograms, Q and T, each containing n buckets, the intersection of the histograms is defined to be:

$$\sum_{j=1}^{n} min(Q_j, T_j).$$

where j ranges over each color in the histograms.

Our system creates a *DualGraph*, defined by nodes and edges to store the spatial region information of the segmented image. A node in the dual graph corresponds to a region in the segmented image, and an edge between two nodes indicates that two regions are adjacent to one another. Each node also contains a histogram of the RGB color information of the region, and is later used in the retargeting process.

5 Importance Map

The main issue with traditional resizing or downsampling is that by uniformly throwing away information, it ignores the fact that some parts of the image are more important than others. Intelligent retargeting should attempt to make an informed choice as to what regions of an image are determined to be important. These regions should be large enough to be recognized by giving more space to these regions and less space to other parts of the image. Truly understanding what is important in an image requires a thorough understanding of what the image contains and what the viewer needs. Some recent results [Chen et al. 2003a; Fan et al. 2003; Ma and Zhang 2003; Itti et al. 1998] suggest that some heuristics work well (albeit imperfectly) on a broad class of images. The two heuristics that are used in most (if not all) systems determining importance in imagery are: specifically recognizable objects (faces) are usually important; and regions of the image that are most likely to attract the low-level visual system are likely to be important. The remainder of this section describes how we have realized these heuristics and describe how we employ them to re-create automatic retargeting.

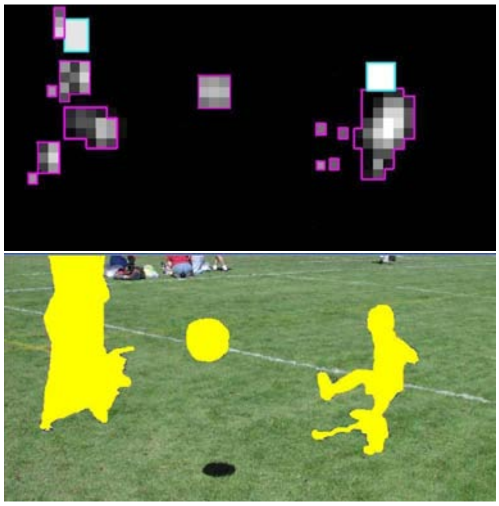


Figure 4: Importance map. *Top) Saliency regions are outlined in magenta, face regions in cyan. Bottom) Regions of importance.)*

To identify the *ROI*, we first compute an importance map that assigns a scalar value to each pixel estimating the importance of that image location based on an attention model. Like previous methods [Suh et al. 2003; Chen et al. 2003b] we use measures of visual salience (e.g. image regions likely to be interesting to the low-level vision system) and high-level detectors for specific objects that are likely to be important, such as faces and signs. Our implementation computes the importance map as a scaled sum of a visual saliency algorithm [Itti et al. 1998] and a face detection algorithm [Niblack et al. 1993].

5.1 Image Attention Model

We use the saliency-based image attention model [Itti et al. 1998] to generate the first contribution to the importance map. The saliency model is used to extract attended locations in complex scenes based on a low level visual model that uses color, intensity and edge orientation as visual cues. The technique uses Gaussian pyramids to compute several 'feature maps' for three low level features: color C, intensity I, and orientation O, which represent the visual scene. Such feature extraction is achieved through linear filtering for the given feature type, followed by a center-surround operation which extracts local spatial discontinuities for each feature type. Spatial discontinuity locations are then combined into a unique 'saliency map' represented as:

$$S = \frac{1}{3}(N(I) + N(C) + N(O)) \tag{1}$$

where N denotes normalization.

The two-dimensional topographical saliency map is used to determine the importance values within the original image. We binarize the saliency map to find the *ROI*. The *IV* can be calculated as:

$$IV_{saliency} = \sum_{(i,j \in R)} B_{i,j} \cdot W^{i,j}{}_{saliency} \tag{2}$$

$B_{i,j}$ denotes the gray-scale value of pixel (i, j) in the saliency map. Since people pay more attention to the region near the center of an image, a normalized Gaussian template centered at the image is used to assign the positional weight $W^{i,j}{}_{saliency}$.

5.2 Face Attention Model

Images of people are popular as well as important in many application areas. However, saliency map generation relies only on low-level features, and it might not be able to recognize faces correctly. The face is a highly important characteristic of human beings, and dominant faces in images certainly attract viewers' attention. Therefore, we use a face attention model in addition to the image attention model. By applying face detection [Schneiderman and Kanade 2000; Niblack et al. 1993], we obtain information about faces in the image such as position, region, and pose. The size and position of a face usually reflects its importance. Hence, the importance value in this model is calculated:

$$IV_{face} = \sqrt{Area_{face}} \cdot W^{i}_{pos} \tag{3}$$

where $Area_{face}$ denotes the size of the detected face region, and W^{i}_{pos} is the weight of its position and $i \in [0, 8]$ is the index of the position, as defined in [Ma et al. 2002].

5.3 Calculating Regions of Importance (ROI)

The saliency and face detection algorithms take color images as input return gray-scale images whose pixel values represent the importance of the corresponding pixel in the input image. The importance map, which is the attention model for the image, is built up by combining a series of importance measures. This allows the system to be adapted to differing image creation goals, and to be easily extensible. The importance map computation can accommodate other attention models as desired. A semi-automatic version of the algorithm can be easily implemented by allowing a user specify important regions. We normalize pixel values from the attention model output images and sum them then re-normalize to create the combined importance map.

Rather than using an exhaustive search to find the best possible regions of importance (*ROI*), we use a greedy algorithm. Using the combined importance map, our method finds initial candidate *ROI* and grow them until they meet the requirements for being the final *ROI* of the image. The process is described in two steps as follows:

Step 1: Identify candidate *ROI*. A candidate *ROI* is defined as a minimal region that identifies key important parts of the image. Each importance value from the importance map is mapped to the segmented regions of the image. In order to this, we calculate an importance value for each node of the *DualGraph* by summing pixel values in the corresponding region of the importance map.

Step 2: Grow the *ROI*. We extend the method of Swain and Ballard [1991] to include the additional dimension of importance and grow the *ROI* by combining nodes in the *DualGraph*. The candidate *ROI* grow by using a clustering algorithm that considers the unexplored or unattached node with the highest importance. Regions with small importance that are adjacent to regions with higher importance, but which cannot be combined because of color differences, are treated as unimportant. The clustering algorithm is applied recursively until all nodes have been explored. Examples of importance maps and *ROI* are shown in Figures 4 and 8.

If all the *ROI* fit inside the new image target specification we simply crop the input image to the target size. The automatic cropping method we provide is similar to the successful methods of previous papers. However, our method differs in some of the details of how the *ROI* are computed. If the *ROI* do not fit the target size, then the image undergoes additional processing as described in Sections 6 and 7.

6 Background Creation

Once the *ROI* are detected, the background is created by removing the *ROI* from the source image, storing the centroids of the *ROI*, filling the resulting gaps, and down sampling the result to the target size. Storing the centroid positions of the *ROI* aids in minimizing visual artifacts in the pasting process(Figure 6 c-d). Inpainting is necessary because an *ROI* may not cover its original image area when pasted back onto the down sampled background. We reconstruct the missing gaps with a plausible texture, using the inpainting algorithm of Harrison [Harrison 2001]. This method reconstructs an image with the same texture as the given input image by successively adding pixels from the input image in a particular order. The procedure is capable of reproducing large gaps even with the interaction of neighboring pixels, and transfers large complex features of the input to the output image. We note that this method could easily be replaced by other techniques [Pérez et al. 2003; Drori et al. 2003]. A second dual graph is then created for the background image using the image segmentation process.

Figure 5 shows the masks in the image, and the results after inpainting is applied. Removal of large regions from the image sometimes leads to some visual artifacts. Since the removed *ROI*' centroid positions are maintained while pasting back onto the updated background, most of these artifacts tend to be minimized in the final retargeted image.

Figure 5: *Background creation. Demonstrating masking, removal of important objects, and inpainting. Left: Original image. Center: Mask areas shown in yellow. Right: After inpainting.*

7 Pasting

The key step in our approach is *pasting* where the *ROI* are inserted onto the newly created background. This step is a novel algorithm added to the above existing techniques in the retargeting process. The algorithm is designed to preserve the relative positions of the *ROI* as this is important to recognizability [May 2000]. The algorithm greedily pastes each *ROI*, beginning with the one with the largest importance. The process of best fitting the importance objects onto the resized background for maximizing the functionality of these *ROI*, can be formulated as a constrained optimization algorithm. The constraints of this algorithm impose certain restrictions on the placement and size of the *ROI* on the resized background in order to maintain certain perceptual cues in pictorial images. These constraints are described as follows:

- *ROI*' positions stay the same. The goal of the retargeting process is to maintain the representation of the original image as much as possible. This involves maintaining object ordering in the scene [Rogers 1995].

- Aspect ratios of the *ROI* must be maintained. This constraint helps minimize the loss of information of important regions in the image, thus maintaining retargeted image's functionality.

- *ROI* must not overlap in the retargeted background, if they are not overlapping in the original image. Interposition (or partial occlusion) happens when objects are overlapping. The object that is partially covered by another one appears to be at the back. Since we cannot determine whether a given object is behind or in front of another object in the original image, we cannot determine which object needs to be in front of or behind another object, in the case of overlap. Hence, we apply this constraint.

- Background color of the *ROI* must not change. In order to keep the retargeted image consistent with the original, it is important to ensure that the object is pasted on approximately the same textural color. The dual graphs of both the original image as well as the newly created background, contain the region information for the objects regarding adjacency, and histograms. We use this information to check the colors of the surrounding areas. While pasting the object onto a sub-region of the re-targeted area, we compute color similarity between that sub-region and the regions adjacent to the object. We use the same color dissimilarity measure [Swain and Ballard 1991] that we used during segmentation. By applying this constraint, we tend to avoid major artifacts such as a person standing on grass in the original image, to appear 'flying in the air'.

The objective function for this optimization process is to minimize the change in size, and change in position of the important objects, proceeding in the decreasing order of importance. To determine the position and scale of each *ROI*, we seek to place them near the position in the target image that they would have appeared were they not removed and at a scale as close to the source image as possible. Each object is first placed, at full scale, such that its centroid is on the centroid position saved in the background step (Figure 6 e,g and i). This assures that all *ROI* maintain their relative positions in the target image. Two resizing steps shrink the *ROI* uniformly to fit in the target image. All resizing is done uniformly to preserve the aspect ratios of the *ROI*, as this is important to recognizability [Rushmeier et al. 1997].

The first resizing step shrinks each *ROI* to avoid overlap with any already placed *ROI* because partial occlusions might give a false depth cue [Rogers 1995]. The *ROI* being placed is checked to make sure that it is being pasted in a region of similar color as its background in the original image using the original images dual graph and the background images dual graph. The shrink amount for each object is determined as follows:

Variables:
$width$ = Width of *ROI*'s bounding box.
$height$ = Height of *ROI*'s bounding box.
$newWidth$ = new width of shrunk *ROI*'s bounding box.
$newHeight$ = new height of shrunk *ROI*'s bounding box.
$overlapLeft$ = amount of overlap to the left of bounding box.
$overlapRight$ = amount of overlap to the right of bounding box.
$overlapTop$ = amount of overlap at the top of bounding box.
$overlapBottom$ = amount of overlap at the bottom of bounding box.
$maxOverlap = MAX(overlapLeft, overlapRight, overlapTop, overlapBottom)$.

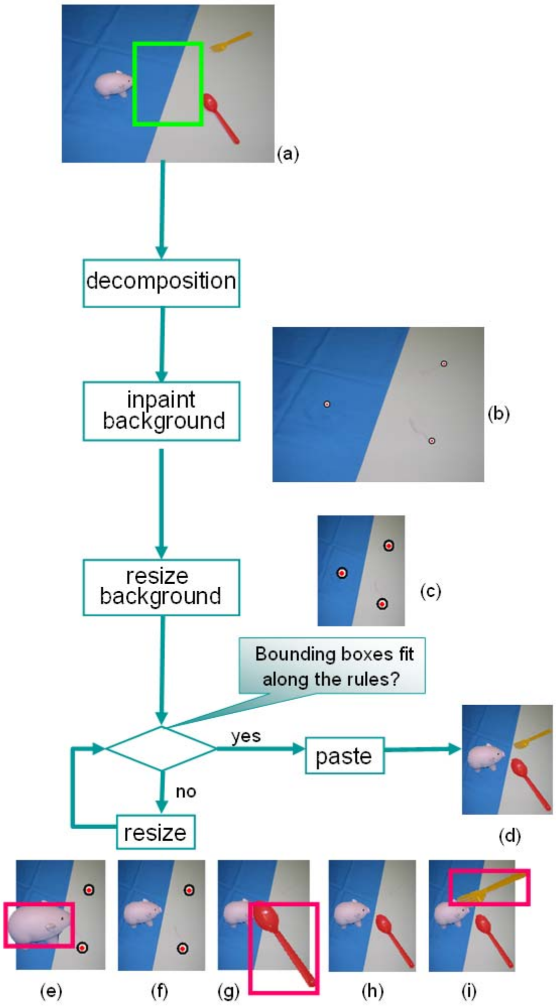


Figure 6: Pasting *A flowchart for the pasting algorithm.*

procedure *ShrinkROI*(*ROI*)

Compute the bounding box and centroid for each *ROI*.
if all *ROI* placed on their centroids, fit within target size **then**
    Perform automatic cropping.
else if the bounding boxes overlap with each other OR with the target image's boundary or with a different textured background **then**
    Starting with the most important *ROI*, compute overlap of its bounding box s.t. shrink factor is proportional to its aspect ratio:
    if $maxOverlap == overlapLeft$ OR $maxOverlap == overlapRight$ **then**
        $newWidth = (width - maxOverlap)$.
        $newHeight = \frac{newWidth}{width} \times height$.
    else if $maxOverlap == overlapTop$ OR $maxOverlap == overlapBottom$ **then**
        $newHeight = (height - maxOverlap)$.
        $newWidth = \frac{width}{height} \times newHeight$.
    end if
end if

The *DualGraph* of the original image provides the regions adjacent to the *ROI* in the original image. If these regions could not be merged, using the metric of Swain and Ballard [Swain and Ballard 1991], with the regions of the background dual graph that the *ROI* covers in the target image the *ROI* is uniformly scaled until this condition is met. This consistency is important because we want *ROI* to be in similar surroundings, but also because matching backgrounds helps avoid visual artifacts [Kosslyn 1978]. A flowchart of this process is shown in Figure 6.

8 Evaluation

The results indicate that retargeting tends to preserve the recognizability of important objects, when compared to traditional resizing techniques. Our retargeting method has worked reasonably well in many cases, especially when important objects are far apart from each other. Though we might be deforming the original image, we are able to better allocate the source images' important features in the target images. In addition, we maintain some important cues concerning the information in the retargeted image. The retargeting algorithm works reasonably well in many cases, especially when important objects are far apart. Though the algorithm may deform the original image, we are able to better allocate the source images' important features in the target images. In addition, we maintain some important visual cues in the retargeted image. For example, we preserve topological relationships between objects and objects tend to remain on their original background as shown in Figure 9.

The improved recognizability of important objects comes at the expense of distorting the overall image. While this design is based on the assumption that the recognizability of important objects is the key criteria for small images, we were concerned that the acceptance of distortions by viewers is a subjective assessment. Therefore, we conducted two preliminary studies to assess the viewers' reactions to the retargeted images.

Table 1: Percentage of selection in cell phone size images.

Scale vs. Crop	Retarget vs. Scale	Retarget vs. Crop
86.1%	83.3%	89.1%

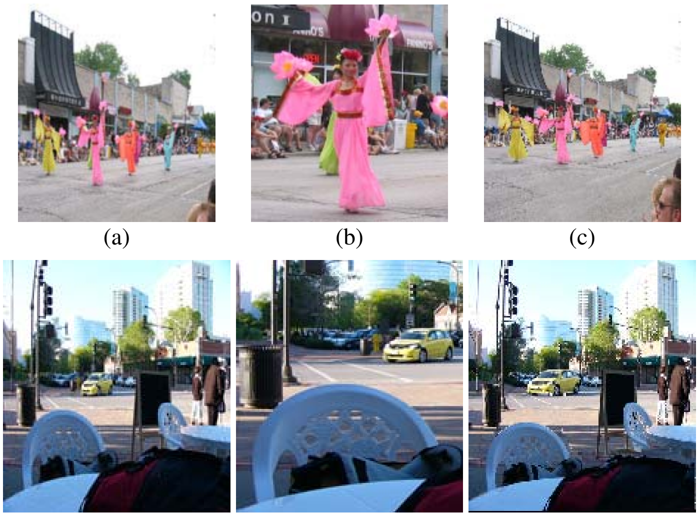
Table 2: Percentage of selection in PDA size images.

Scale vs. Crop	Retarget vs. Scale	Retarget vs. Crop
72.07%	72.4%	84.2%

User Studies: The first user study involved examining retargeting (640×480 pixel) images to cell phone display resolution (154×171 pixels), while the other involved examining retargeting (640×480 pixel) images to PDA display resolution (320×320 pixels). In both studies 40 images were resized using automatic uniform scaling, automatic cropping based on the image's importance map, and retargeting. 12 subjects each participated in the two sets of experiments. The cell phone experiment comprised 6 naive subjects; 2 secretaries, 1 person from a book store, and 2 people at a coffee shop. The other 6 subjects were computer researchers. 6 naive subjects participated in the PDA experiment; 4 clinical psychologists, 1 lawyer, and 1 artiste. The other 6 were computer researchers. Subjects had either normal or corrected-to-normal vision.

In the cell phone and PDA experiments, each subject participated in a number of trials in random order. Within each trial, a participant was first shown a larger image, and then randomly shown two of the three scaled, cropped, retargeted images. The cropped images were generated based on the importance map used to generate the retargeted images. The instruction that was shown to each subject was, “Thank you for agreeing to participate in this experiment. The experiment consists of several trials. For each trial you will be first shown an image that will be displayed for 6 seconds. Then the first image will be replaced by a pair of images. Please select the right or left image from the pair of images that you think best represents the first image. Press the ‘C’ key to choose the left image, or press the ‘M’ key to choose the right image. Please go as fast as you can without making mistakes. Now, please position your left forefinger on the ‘C’ key and your right forefinger on the ‘M’ key, and press the space bar to begin.” The order in which each subject performed the selection was randomized among subjects. The data shows that the subjects preferred retargeted images over scaled images 83.3% of the time and preferred retargeted images over cropped images 89.1% of the time for cell phone images.

In the PDA experiment, data shows that subjects preferred retargeted images over scaled images 72.4% of the time and preferred retargeted images over cropped images 84.2% of the time. The other comparisons are shown in Tables 1 and 2. Using a paired T-test, we found that there is a significant statistical difference in image selection between all of the different comparisons: cell phone sized retarget vs. scale and retarget vs. crop images($p = 0.042$), cell phone sized scale vs. crop and scale vs. retarget images ($p = .016$), cell phone sized crop vs. scale and crop vs. retarget images ($p = 0.041$),PDA sized retarget vs. scale and retarget vs. crop images($p = 0.002$), PDA sized scale vs. crop and scale vs. retarget images ($p = .005$), PDA sized crop vs. scale and crop vs. retarget images ($p = 0.007$).

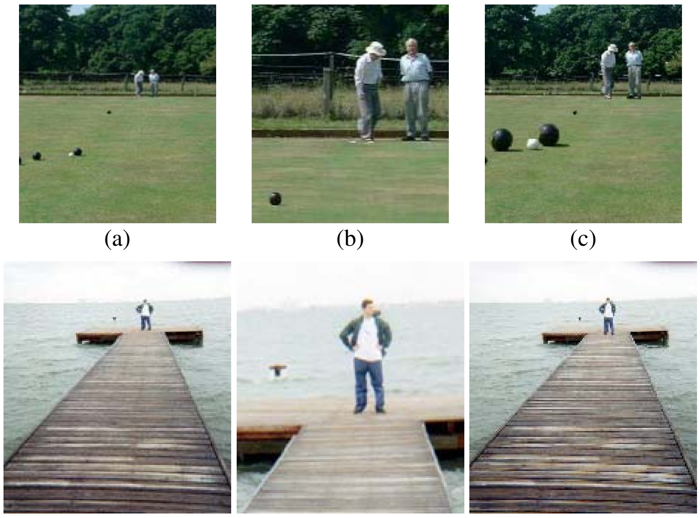


(a) (b) (c)

(e) (f) (g)

Figure 7: *During exit polls, subjects said that the difference between the above scaled and retargeted images was indeterminate, and often chose one or the other. Left) Scaled image. Center) Cropped image. Right) Retargeted image. In the top row, the dancers are very close to each other, leaving little room for pasting, to prevent the objects’ bounding boxes to intersect. Similarly, in the bottom row, several objects including the car and chair are important, leaving little room for them to be pasted larger.*

We note that the image set includes some images where our retargeting method has no advantage (e.g. there is a single region of interest), so the results are close to being as good as can be expected. From exit interviews we were able to determine that there



(a) (b) (c)

(d) (e) (f)

Figure 8: *Subjects prefer crop over these scaled and retargeted images, as the people are shown bigger. Left) Scaled image. Center) Cropped image. Right) Retargeted image. The importance map identifies the ball and the people to be important in the top row, and the man and part of the dock in the bottom row.*

are cases where the method fails to provide preferable images. The most common failure mode is when the importance map contains too many important objects as shown in Figure 7. Viewers also felt a single important object at a large scale better represented the image in some cases as shown in Figure 8, this case always involved people as the most important object.

During exit polls conducted with the subjects, some of the comments were: “For deciding on better representation, I chose the image which had all the objects from the larger image, so I chose the scale over crop, and retarget over crop”, “In images with people, I preferred the image that did not distort the people so much.”. The data values however, cannot be generalized for all types of images.

Most of the 40 images used in the poll had multiple objects of interest placed on uniformly textured backgrounds (Figure 9). Also, the performance of the retargeted approach is more apparent with the *ROI* in the images separated by a reasonable distance, rather than them being immediately adjacent to one another. The retargeting method is limited by the performance of the components that are used: image segmentation and the importance model must succeed at identifying the important objects. If the performance of these components is insufficient, a semi-automatic version of our method can be applied where the user manually identifies the important object. A more fundamental limitation in our approach is that we have explicitly designed it for images where the recognizability of a small set of key objects is critical. Our method may be ill-suited to images for which this is not the case, for example when there are too many (or no) distinct, important objects, or when the relationships between the objects are significant.

Future Directions

Our algorithm is non-photorealistic and does not maintain semantic relationships among the objects in the retargeted image. For example, the system does no establish semantic correlation between objects and their shadows. In Figure 1, the shape of the ball’s shadow in the retargeted image is not consistent with the shape of the ball. This is because the system identifies the ball and not its shadow to be important, leading to the resizing of the shadow along with the background. One interesting avenue would be the usage of composition rules such as Gestalt laws, balance, framing, rule of thirds, and diagonals to improve the algorithmic output. These are excellent directions to follow for improved re-layout results. Emotional connectedness between objects is another aspect that the system cannot address at this point. We believe that by making the system semi-automatic, the user can designate emotionally important objects in an image. The method may not be optimal if an important feature is on a similarly textured background, or the image contains one large object.

The main advantages of our method are:

- Unlike cropping, the retargeting method handles the case where there are multiple important features in an image by considering different regions of the image independently. Important features are extracted from the image and re-arranged such that they best fit in the target image.
- Unlike scaling, we tend to preserve image functionality of important regions in the image, by enhancing their recognizability in the retargeted image. For example, in Figures 1 and 9, the important image elements are brought closer together when the image is downsized; less of the unimportant background is shown so that more space can be used to show the important features.

However, our method has disadvantages and limitations:

- May not preserve semantic relationships between objects.
- Important regions in the image are resized independently based on saliency, and may convey wrong relative proportions between these regions.
- May not handle repeated textures correctly. If the saliency for a unit patch in the image is high, and similar such patches are repeated all over the image,such as a crowd of people, or a bed of flowers, the algorithm does not treat each of these patches equally, and may lead to visual artifacts.
- If there is a single large object occupying most of the area in the image, the result tends to be similar to that of cropping.

We ran the algorithm on an Intel(R) Xeon(TM) CPU 3.06GHz processor with 2GB RAM. The segmentation and importance map generation take the longest computation time ranging between 5 (20 image segments) to 40 minutes (160 image segments) for the image set we used. Once the *DualGraph* is computed, the pasting algorithm has a computation time ranging between (30 seconds to 1.2 minutes). We would like to investigate the possibilities of optimizing the algorithm further for porting to mobile phones with more restricted processing power.

9 Conclusion

Images are a powerful mode of communication. Because of this, our computing infrastructure is constantly evolving to deliver higher quality imagery. Ubiquitous high-speed networking provides imagery to home theater screens, cellular phone displays, networked PDAs, and even displays embedded in refrigerators, elevators and airplane seats.

Our vision is that increasingly ubiquitous displays can provide people with information when and where they need it, provide more effective channels of inter-personal communication, deliver educational media, and provide entertainment. Achieving this vision requires providing imagery for a variety of display devices.

This work in automatic image retargeting is a first step in that direction. We have demonstrated an algorithm that allows a user to author imagery once, and then automatically *retarget* that imagery for an assortment of display devices. Results generated by our method tends to minimize the loss of detail and distortion. In addition, the algorithm moves significant regions closer together while retaining key feature relationships in the image. One can imagine a variety of retargeting applications such as: entertainment images for cellular phones, training images for PDAs, status information for "heads up" displays, and image icons for files on a computer.

10 Acknowledgements

We would like to thank the reviewers for their valuable suggestions and feedback. Thanks to Amod Setlur, Amy Gooch and the rest of the Northwestern University graphics group for their comments and critical review. This material is based upon work supported by the National Science Foundation under grant numbers IIS-0097456 IIS-0416284. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

References

http://www.adobe.com/. Adobe Photoshop.

BREGLER, C., 1998. Tracking people with twists and exponential maps.

CARPENDALE, M. S. T., AND MONTAGNESE, C. 2001. A framework for unifying presentation space. In *UIST '01*, 82–92.

CHEN, L., XIE, X., FAN, X., MA, W.-Y., ZHANG, H.-J., AND ZHOU, H. 2002. A visual attention model for adapting images on small displays. Tech. Rep. MSR-TR-2002-125, Microsoft Research, Nov.

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

CHEN, L.-Q., XIE, X., MA, W.-Y., AND ZHOU, H.-Q. 2003. Image adaptation based on attention model for small-form factor devices. In *Proceeding of the 9th International Conference on Multimedia Modeling*, IEEE, 421–439.

DRORI, I., COHEN-OR, D., AND YESHURUN, H. 2003. Fragment-based image completion. *ACM Transactions on Graphics 22*, 3 (July), 303–312.

FAN, X., XIE, X., ZHOU, H.-Q., AND MA, W.-Y. 2003. Looking into video frames on small displays. In *Proceedings ACM Multimedia 2003*. Short Paper.

http://www.gimp.org/. The GNU Image Manipulation Program.

HARRISON, P. 2001. A non-hierarchical procedure for re-synthesis of complex textures. In *Proceedings of WSCG 2001*, 190–197.

ITTI, L., KOCH, C., AND NIEBUR, E. 1998. A model of saliency-based visual attention for rapid scene analysis. vol. 20, 1254–1259.

JOJIC, N., FREY, B., AND KANNAN, A. 2003. Epitomic analysis of appearance and shape. In *9th IEEE International Conference on Computer Vision*, 34–41.

KOSSLYN, S. M. 1978. On the ontological status of visual mental images. In *Theoretical issues in natural language processing*, Association for Computational Linguistics, 167–171.

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

MA, Y.-F., AND ZHANG, H.-J. 2003. Contrast-based image attention analysis by using fuzzy growing. In *Proceedings ACM Multimedia 2003*, 374–381.

MA, Y.-F., LU, L., ZHANG, H.-J., AND LI, M. 2002. A user attention model for video summarization. In *Proceedings of the tenth ACM international conference on Multimedia*, ACM Press, 533–542.

MAY, J. 2000. Perceptual principles and computer graphics. In *Computer Graphics Forum*, vol. 19.

MEER, P., AND GEORGESCU, B. 2001. Edge detection with embedded confidence. *IEEE Transactions on Pattern Analysis and Machine Intelligence 23*, 12.

NIBLACK, W., BARBER, R., EQUITZ, W., FLICKNER, M., GLASMAN, E. H., PETKOVIC, D., YANKER, P., FALOUTSOS, C., AND TAUBIN, G. 1993. The qbic project: Querying images by content, using color, texture, and shape. In *Vol. 1908*, SPIE, 173–187.

PÉREZ, P., GANGNET, M., AND BLAKE, A. 2003. Poisson image editing. *ACM Transactions on Graphics 22*, 3 (July), 313–318.

ROGERS, S. 1995. *Perceiving pictorial space. In Perception of Space and Motion, Chapter 4.* Academic Press, Inc.

ROWLEY, H. A., BALUJA, S., AND KANADE, T. 1996. Human face detection in visual scenes. In *Advances in Neural Information Processing Systems*, The MIT Press, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds., vol. 8, 875–881.

RUSHMEIER, H., BARRETT, H., RHEINGANS, P., USELTON, S., AND WATSON, A. 1997. Perceptual measures for effective visualizations. In *VIS '97*, IEEE Computer Society Press, 515–517.

SCHNEIDERMAN, H., AND KANADE, T. 2000. A statistical model for 3d object detection applied to faces and cars. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition - Volume 1*, IEEE, 746–751.

SUH, B., LING, H., BEDERSON, B. B., AND JACOBS, D. W. 2003. Automatic thumbnail cropping and its effectiveness. In *User interface software and technology*, ACM, 11–99.

SWAIN, M., AND BALLARD, D. 1991. Color indexing. *International Journal on Computer Vision 7*, 1, 11–32.

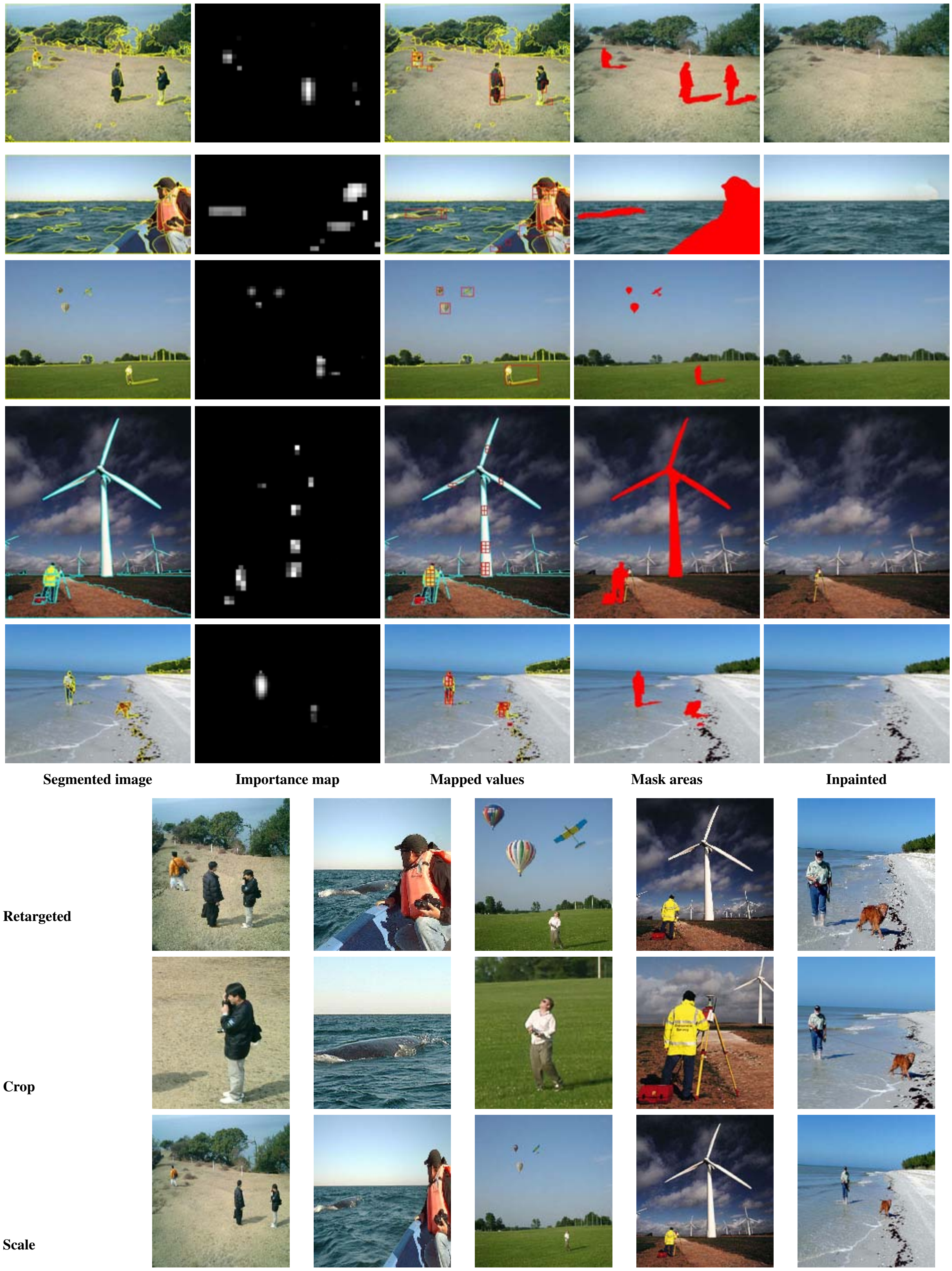


Figure 9: *A comparison between existing image resizing techniques and automatic image retargeting method.*