Image Resizing using Seam Carving

Kristin Rachor
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA 56267
racho008@morris.umn.edu

ABSTRACT

Image resizing has become more necessary with the increased popularity of cell phones, tablets and other electronic devices with varying screen sizes. This paper presents methods for resizing images and videos while attempting to preserve the important content of that image or video. An algorithm called seam carving can expand or reduce the size of an image while typically maintaining quality and content. Seam carving is not always effective however and there have been recent developments and modifications on this algorithm. This paper presents two advancements on seam carving, one that optimizes image retargeting on images with many repeated objects or patterns. The other applies the method of seam carving to video resizing.

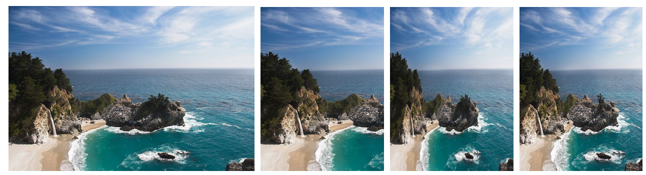
Keywords

seam carving, image resizing, object carving, video resizing, SeamCrop, retargeting

1. INTRODUCTION

Screens today come in a variety of sizes and dimensions. Images and videos are displayed on devices as small and square as smart watches and as large and wide as movie theater screens. The number of pixels in a screen is known as the display resolution and is measured as width by height. With the need to view the same media on multiple devices, researchers have developed image and video resizing algorithms that give the viewer a similar visual experience across multiple devices. Another situation where images may need to be resized is on webpages. HTML supports dynamic changes of all content on a webpage, including images and text, as a user resizes their window.

The most common types of image resizing used today are image cropping and image scaling. These forms of resizing are not optimal because they are not aware of the content in the image. They either distort the image or crop out important content. Shown in Figure 1 the original image, (a), is cropped to 50% of it's width in (b), removing almost all of the ocean and rock content. Figure 1 (c) is reduced to 50% of (a)'s size by using homogeneous scaling [9]. This method retains all of the original image's content but causes an obvious distortion.



(a) Original image (b) Cropping (c) Scaling (d) SC

Figure 1: Image (a) is reduced to 50% of its original width using cropping, scaling and seam carving. [9]

Content aware resizing is a way to change the aspect ratio of an image or video that recognizes the most important content and prevents the distortion and removal of that content. This paper focuses on the method of seam carving [1] and its improvements to achieve content aware image resizing.

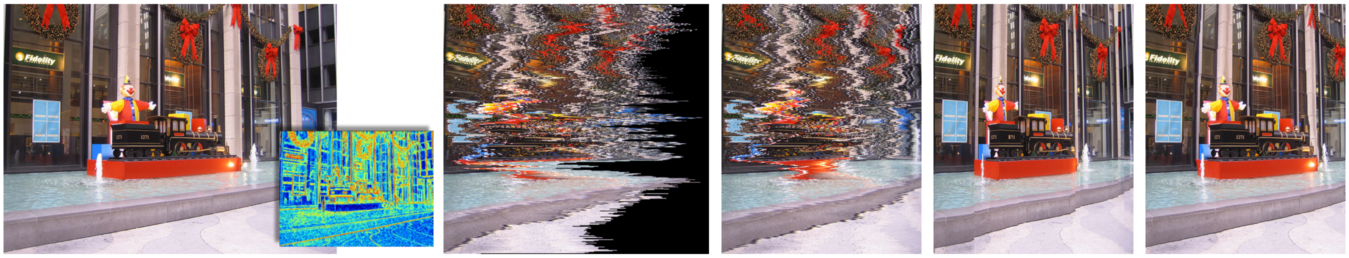
2. BACKGROUND

There were two primary ways that content aware image resizing, also known as image retargeting, was achieved prior to the discovery of seam carving. Important features were detected using a top-down or bottom-up approach. Using top-down methods, important content is detected using tools such as face detectors. Bottom-up methods construct visual saliency maps of an image. These saliency maps topographically highlight the visual saliency, or qualities that stands out, in an image. Then the important aspects of the image, identified by either using top-down or bottom-up approaches, are included in the window that is to be cropped to the desired size. These methods yield impressive results but still rely on traditional cropping and resizing techniques. [1]

A more recent approach to content aware image resizing presented by Wang, Tai, Sorkine, and Lee uses what they call a Scale-and-stretch method [8]. This method breaks the image up into a grid and computes the optimal scaling factor for each region of the grid and reduces the entire image accordingly [8].

Using seams for image editing was in effect prior to the existence of the seam carving algorithm. One way seams were used was to automatically compose a collage of images from a collection of photos. Optimal boundaries or seams were calculated to place these images together in fragments. However, seams had never been used to retarget image, re-

UMM CSci Senior Seminar Conference, December 2015 Morris, MN.



(a) Original Image (b) Optimal (c) Pixel (d) Column (e) Seam Carving

Figure 2: Four different resizing methods used to change the width of (a), the original image and its energy function. (b) shows the global removal of pixels with the lowest energy, regardless of their position, (c) shows the removal of the pixel with the least amount of energy in each row, (d) removes the columns with minimal energy, and (e) shows seam carving. [1]

searchers Avidan and Shamir were the first to do this in 2009 when they invented seam carving [1].

Since seam carving gained popularity there have been many researchers that have taken the algorithm and made improvements on it. In this paper I have defined the traditional seam carving method and go into detail on two improvements.

3. SEAM CARVING

The idea behind seam carving is to remove regions of an image who's removal is the least noticeable. Areas that are least noticeable when removed tend to be similar to their surroundings, the pixels in this area are considered to be low energy. A pixels energy is high if its color value is very different from its neighbors. A pixels energy is low if it is similar to its neighbors. This section presents seam carving following the algorithm of Avidan and Shamir. [1]

3.1 Energy Function

In this section I formalize the notion of low vs. high energy pixels. There are different energy functions that can be used to measure this. I describe a method that uses the surrounding pixels to measure how different the middle pixel is in comparison. If we have a color image, I, with each pixel having a location in the image (x, y) and a color value (r, g, b) we can calculate the partial derivative to find the energy. A way to approximate this value is by looking at the surrounding pixels $I(x-1, y)$, $I(x+1, y)$, $I(x, y-1)$, $I(x, y+1)$ and using those to find the partial derivatives in each direction:

$$dx = |I(x+1, y) - I(x-1, y)|/2$$

for the x direction

$$dy = |I(x, y-1) - I(x, y+1)|/2$$

for the y direction.

We will then sum the horizontal and vertical derivatives to get one value that we will call the energy level.

$$e(x, y) = dx + dy$$

3.2 Seams

Figure 2(a) shows an image and its energy function represented visually. If we wanted to reduce this image's width we could crop it but would lose some content. So instead we will want to remove the parts of the image with the lowest energy. A naive approach would be to just remove the pixels with the lowest energy but in doing so you will no longer have a rectangular image (Figure 2(b)). To maintain a rectangular image you must remove an equal number of pixels from each row (or column). However just removing the lowest energy pixel from each row over and over will break up the pieces of the image (Figure 2(c)). So the next logical step would be to just remove the lowest energy columns. However this still distorts our image and creates obvious jumps in the image, evident in the diagonal floorboard line (Figure 2(d)).

This leads us to seams. A seam is simply a path that is one pixel wide and runs from the bottom to the top of an image. This path does not have to be straight, it can veer to the left or right, but must stay connected [5]. Let image I be a $n \times m$ image where $x(i)$ is a point in the horizontal direction and i is a point in the vertical direction. A vertical seam is defined as:

$$s^x = \{s_i^x\}_{i=1}^n = \{(x(i), i)\}_{i=1}^n, \text{ s.t.}$$

$$\forall i, |x(i) - x(i-1)| \leq 1$$

Here s signifies the seam which will start at the bottom of the image (when $i = 1$) and traverse up to the top (when $i = n$). The 'for all i' condition states that when you move up you can be at most one pixel to the left or the right of position below you. This insures that the seam stays connected. This same definition holds for horizontal seams where you swap all occurrences of x for y and all occurrences of n for m. Then when a seam is removed all the pixels will shift left (or up) to fill in the missing seam.

3.3 Computing Seams

Now that we know the energy function and what constitutes a seam we need to traverse through the image to find the lowest cost seam. We will use dynamic programming to accomplish this. In this case dynamic programming uses sub-seams that have already been calculated when calculating longer seams. This means that we do not need to calculate the least energy sub-seam multiple times.

We will start from the top of the image and work our way down, computing the minimum energy seam from the top of the image to pixel (i, j) as shown in Figure 3. Pixel index $(0, 0)$ is the bottom left pixel of the image. The variable i represents the vertical placement and j represents the hor-

4	**2**	1
3	5	2

4	2	**1**
5(3+2)	5	2

4	2	**1**
5(3+2)	6(5+1)	2

Figure 3: Three pixel's energy values being updated (in red) based on the pixels above them (in grey). The top figure is an example of six pixels and their energy values. The proceeding two figures are updated versions of the pixels as the second row's energy values are added to the lowest energy value above them. For example, in the top figure, the value 3 is updated to 5 because $3+2=5$ and 2 is the lowest energy valued pixel above the pixel valued 3.

izontal placement in the image. $M(i,j)$ is the least energy value path from point (i,j) to the top of the image, which we can compute via

$$M(i,j) = e(i,j) + \min\left(M(i-1,j-1), M(i-1,j), M(i-1,j+1)\right)$$

So you will start by calculating the energy value, or $e(i,j)$, for each pixel in the top row. The next row down will then pick the minimum value from the three 'seams' above it and then add the energy value of the pixel at (i,j). This will happen recursively and stores each pixel's minimum seam value. After you have traversed the entire image, each pixel on the bottom row will have a value M. The lowest value M on the bottom row will be the endpoint of the minimum valued seam. Once the minimum valued seams are calculated you will traverse back up the seam and remove each pixel in that seam. Then all pixels to the right of that seam will be shifted one pixel to the left to fill in the missing seam. This entire process will be repeated each time a seam is removed. You can scale the image by both it's width and height by alternating removing horizontal and vertical seams until the image achieves the desired dimensions.

4. IMPROVED METHODS

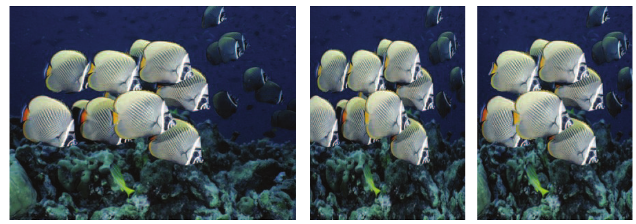
Seam carving does a good job re-sizing certain images such as landscapes. However it often distorts other types of images as seen in Figure 4's seam cropping output. No single method for resizing will be the best option for every image. This section describes three additions/improvements to the original seam carving algorithm.

4.1 Multi-Operator

After content aware image resizing gained popularity the original authors who presented seam carving, Avidan and Shamir with the addition of Rubinstein, published another paper that made the algorithm more effective on a larger

(a) Original (b) Cropping (c) SC (d) Multi-Op

Figure 4: Example where seam carving is not the best option. However when seam carving is combined with cropping it performs the best (d). [7]



(a) Original (b) Multi-Op (c) Object Carving

Figure 5: An example of a situation where object carving (c) is much more visually appealing than seam carving(b) after being resizied from (a). [2]

range of image types [7]. The algorithm combined seam carving, cropping, and scaling in an optimal manner depending on the image's content (Figure 4). They called this algorithm multi-operator, or multi-op, because it combined many image resizing techniques. It's improvement lead many algorithms to enhance seam carving to better fit a wider variety of images. Subsequent algorithms compare their results to multi-operator instead of traditional seam carving, as we will see in the next section.

4.2 Object Carving

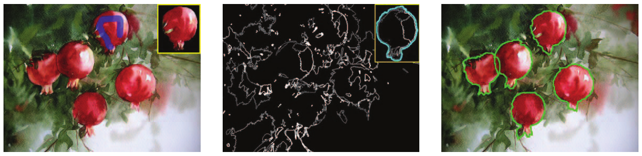
Dong, et al. found that images with many repeated objects could be resized by removing repeated material [2]. It has been found that humans are less likely to notice missing areas of an image than distortion in an image [6]. This is especially true in images with human faces but can also be seen in Figure 5. The fish look squished in Figure 5(b) but still look proportional in Figure 5(c) despite the loss of some fish. The object carving algorithm utilizes this observation. By removing repeated parts of an image there is more room to shrink that image without deformation.

As you might guess the best images to use the proposed object carving algorithm with are images whose primary contents are similar objects. This can be anything from a school of fish to a bouquet of flowers to a fence with a repeated diamond wire pattern.

4.2.1 Object Carving Algorithm

There are two main steps to successfully achieve visually appealing object carving results. First you must detect similar objects and calculate their visual importance in order to select which object will be removed. The removal step is next, where a revised version of seam carving called multi-operator (Section 4.1) is used to carve out these sections.

There are many different factors that make objects appear



(a) Paint Selection (b) Shape Information (c) Detection Result

Figure 6: To detect similar objects, the user must first manually select an object to get a template, then hierarchical segmentation is performed to extract shape information. [2]

(a) Original Image (b) Visual Importance (c) Multi-Op (d) Object Carving

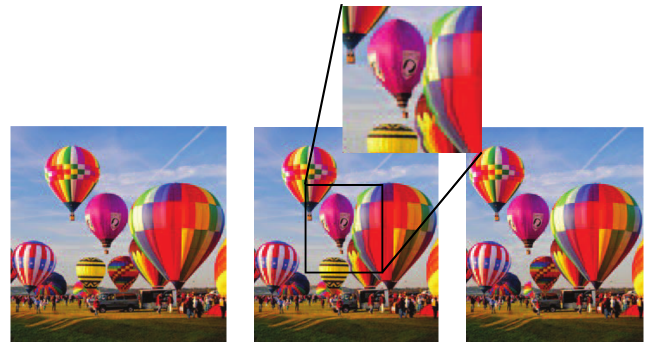
Figure 7: The visual importance of an objects is automatically calculated by combining several features. Brighter objects are more important and the darker objects are more likely to be carved out. The object carving image, (d), is favored by 60.32 percent of users over (c). [2]

similar: outer shape, inner texture, illumination, overlapping, and more [2]. There are also many methods used to detect similar objects in an image. Some are only good at detecting simple objects in a static background. RepFinder is a preexisting method for detecting similar objects, it is the algorithm that will be used as a starting point to the algorithm described in this section [2]. However RepFinder will only consider objects to be similar if their outer shape is very similar which is not always the case for the object carving algorithm because objects that are overlapping each other or have subtle shape differences should still be recognized as a repeated object. There is another option that uses a graph-based method to detect similar objects and does better with overlapping, however it relies heavily on the objects having very similar colors [2].

Dong et al. designed an algorithm that combined these methods in a more robust way [2]. The only cost is that the user will manually have to mark one of the objects as a guide to aid in finding the others. The next step is to use hierarchical segmentation, dividing the image into regions, on both the example object we marked and then again on the whole image. You can see the shape information being extracted in figure 6(b) in both the image as a whole and the sample object [2]. Then in Figure 6(c) the color and shape information is combined to detect the repeated objects. This method held up well in tests and out-performed RepFinder and others.

The next step is to give each of the objects a visual importance value. Visual importance is necessary in the resizing process because we will need to decide which of the repeated objects can be carved out. The importance of an object is measured by how hidden it is. The algorithm measures the layering relation between the different objects and assigns higher importance to images in the foreground, as documented in Figure 7.

We then apply seam carving to carve out the low importance objects. This is achieved using the same algorithm described in Section 3.2 with some slight modifications. An initial reaction would be to mark all the pixels in the object that we will remove and lower their energy values to ensure the seams will run through the object we plan to remove. However removing the object with the lowest visual importance isn't always the best object to remove, seen in Figure 8.



(a) Current Resizing (b) Object Carving 1 (c) Object Carving 2

Figure 8: This displays the importance of evaluating the energy that the seam will carve out as well as the object's energy. In the original (a) the lowest energy balloon is the checkered balloon in the far back. However removing the striped balloon in (c), yields a better result because that leads to less distortion in the balloon above (as shown in the enlarged portion of Figure (b)). [2]

To solve this problem Dong et al. present an algorithm that evaluates the information lost in the whole image when each object is removed [2]. The local information loss of each object is

$$L_i^l = p_i^A + I_i + J(t, O_i),$$

where p_i^A is the percent of the image left after the object is removed, I_i is the visual importance of that object, and $J(t, O_i)$ is how closely the object matches the template image that the user marked at the beginning. We will want to remove objects with a low L_i^l.

We also need to calculate the global information loss of the whole image when an object is removed. This equation is displayed below.

$$L_i^g = \left(\frac{\|V^{\vec{S}} - V_i^{\vec{S}'}\|}{V^{\vec{S}}} + \frac{\|V^{\vec{C}} - V_i^{\vec{C}'}\|}{V^{\vec{C}}}\right) \times I_i$$

$V^{\vec{S}'}$ and $V^{\vec{C}'}$ are the are the shape and color variance of the image, respectively. $V_i^{\vec{S}'}$ and $V_i^{\vec{C}'}$ are the shape and color variance of the objects still left on the image. In other words L_i^g is the difference of the color and shape of the new image compared to the original. We can then calculate the total information loss:

$$L_i = \beta \times L_i^l + (1 - \beta) \times L_i^g$$

They set the percentage of similarity, β, to 0.7. 0.7 was chosen by Dong et al. as the percentage of similarity because it is high enough to only identify objects as the same that really were similar but low enough to account for some shape differences due to overlapping and the like. We will then sort the objects by L_i and look at the two lowest valued objects. For each of those objects the pixels that would be carved out are measured and recorded. The object that removes less information during the seam carving phase is the one that is chosen to be removed. You can see in Figure 8 the yellow balloon in the back is larger and brighter and has nothing obstructing it compared to the smaller balloon to its right. So it would seem that the multicolored balloon would be the first choice to remove, however removing seams above the smaller multicolored balloon distorts the balloons above it. Removing the yellow balloon, however, allows seams to carve out the sky to the left, making it the better choice to remove despite it having more visual importance.

We will then continue with seam carving as presented in Section 3.2 until all the marked pixels are gone. The authors note that if after removing an object the image is smaller than desired the algorithm would simply revert back to the image prior to removing that object and from that point use seam carving to achieve desired size.

4.2.2 Results

A user study of 95 participants was conducted to evaluate the different image resizing method's results in comparison to object carving. They found that in general their algorithm out-performed many others including multi-op and cropping. For example in Figure 5, 52.69% of users selected object carving and only 17.2% of users chose the multi-op resized photo. The remaining users chose a number of other resizing techniques as their most preferred image. Dong et al. also ran the experiment again without showing the participants the original image. Again a majority of users favored object carving. One situation where object carving was particularity effective was where the resizing was extreme.

4.3 Video Seam Carving

The seam carving technique is not limited to static images, recent research has focused on resizing videos as well. This subject is of major importance with the need for movies to be visually appealing on all screen sizes, from theater screens to cell phone screens. The largest challenge is when the aspect ratio changes. The aspect ratio is the relationship between the width and height of a screen. Movies have adapted to different television ratios by adding black boarders on the top and bottom of the screen or by cropping the left and right sides of the video to fit it in the entire screen [4]. While both of these methods are effective, neither of them are ideal.

In this section I will be exploring a method for video resizing that authors Kiess, Guthier, Kopf, and Effelsberg called SeamCrop [4]. The first step is finding a window that contains all the useful content and cropping the sides to the desired size. The next step is to add back some area on each side and then use seam carving to reduce the video size back to the desired dimensions. See figure 9. Their goal was that SeamCrop would have fast computation times and improved results over other video resizing algorithms [4]. Many algorithms are used over an entire video sequence, which is very time consuming. SeamCrop only uses the whole sequence in the cropping portion of the algorithm but does the seam carving frame by frame.

The algorithm has two parts, the first being content aware cropping. This process is pretty straightforward. First each pixel is given an energy value similar to the energy function that was presented earlier in this paper. Then each column's values will be summed up, assuming you are changing the width of the video, otherwise the rows will be summed. Next you will sum up the energies of each cropping window possibility and stored them in an array (Figure 10). This array will then be turned into a two dimensional array that will store the cropped window energy for each frame in the video. Then on that 2D array dynamic programming is used to find the optimal cropped windows for each frame. This method of dynamic programming is very similar to that in the seam carving algorithm. Because each frame's cropped window is different you can get some jittery movement, this is corrected with another method called Gaussian smoothing filter.

The next part of the algorithm is to extend the cropped window to compensate for the seams that will be removed. The method for seam carving is the same as was already

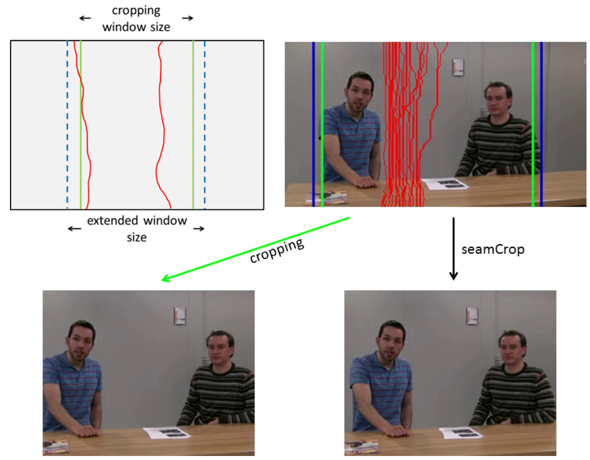


Figure 9: Top Row: optimal cropping window is computed (solid green straight lines). This window is extended(blue dotted lines) to account for the seams that will be taken out. Bottom images show the difference between optimal cropping and using seam carving combined with cropping. [4]

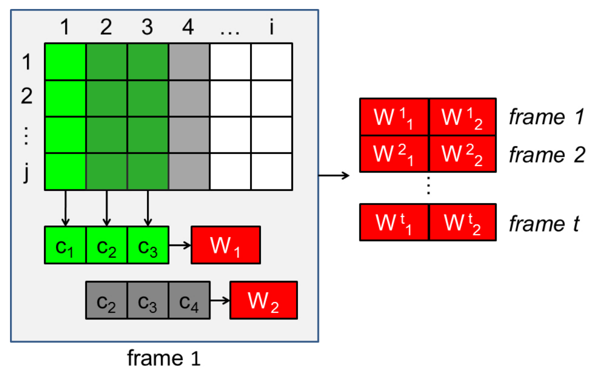


Figure 10: Each column's energy is summed in a value C_i. Then each of these values are summed into a value W_j for each cropped window. Finally the values for the positions of all frames are combined to a 2D array. [4]

presented. The issue at hand is when you remove seams from each frame independently, the video will jump and look jittery. To prevent this and ensure coherence we want to increase the chance of seams landing on or near the seams from the previous frame. Coherence costs based on the previous frame's seam locations are added to the energy map before determining where the next frame's seams will be.

SeamCrop succeeded in creating a video resizing algorithm that has fast computation time while still maintaining visually pleasing results. The authors of SeamCrop decided to conduct a user study to verify this conclusion. There were a total of 19 participants that each viewed twenty-four videos. Twelve were scaled using SeamCrop and the other twelve were the same videos using another video retargeting algorithm. The original unscaled video was not shown as a reference. Video categories varied from animations to sports videos. The conclusion was that no one technique was superior in all test videos. The studies did show however that audiences preferred SeamCrop in videos that had larger visible faces [4].

When using seam carving on videos rather than images one additional detail to pay attention to is the computation time. Since the algorithm essentially performs an advanced version of traditional seam carving on each frame, run times can get large pretty quickly. For example when reducing 400 frames of a 400 by 300 resolution screen by 50% of its original width, the processing time was 1 minute and 6 seconds. On a 1920 by 1080 resolution screen reducing 72 frames to 50% width took 10 minutes and 27 seconds. This is actually faster than previous video retargeting algorithms. The authors that made the SeamCrop algorithm just released another paper in 2014 that built on the SeamCrop algorithm and optimized it by making the program run in parallel [3].

5. CONCLUSION

No one algorithm to resize images is the best option for every image. As we have seen in this paper, traditional seam carving works well on landscapes and images with enough background to carve out [1]. The distortion from seam carving typically occurs in images with faces or images with too much foreground. Seam carving was improved to better handle these images by a method called multi-op that combined cropping and seam carving [7]. While this was an improvement, images, especially images with a lot of content in the foreground, were still getting distorted. Object carving solves this problem, but only when the objects in the foreground are relatively similar [2]. There have been other papers in the field that minimize distortion on other sub-sets of images [9, 8]. This means for optimal image retargeting photo editing software should be able to recognize the appropriate algorithm to use for any given image.

Seam carving was also applied to other media, such as videos. The algorithm is given another dimension to handle consistency between frames [4]. This type of resizing is no more effective than seam carving in images and should only be used when there are slight aspect ratio changes. Not only is seam carving used in other media, it is also used for purposes besides image resizing. It is able to remove an object that a user highlights in an image without changing the size of the image [1].

There are still challenges in the field that need to be solved. Protecting semantic meaning though automatic detection could use further research. Another area to look into is the combining of the top-down and bottom-up methods to work better together [9]. It has been found that if you shirk an image or video enough, eventually no algorithm will be able to keep all content without distortion. This problem is hardly solvable but more research on how small a given image can be shrunk relative to the overall energy is needed. There has been research about adding seams to an image to increase size but more could be done there as well [1].

Acknowledgments

Thanks to Nic McPhee for his help, time, and advice, to Elena Machkasova for her corrections and improvements, and to Kirbie Dramdahl for her feedback and guidance.

6. REFERENCES

[1] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *ACM Trans. Graph.*, 26(3), July 2007.

[2] W. Dong, N. Zhou, T.-Y. Lee, F. Wu, Y. Kong, and X. Zhang. Summarization-based image resizing by intelligent object carving. *Visualization and Computer Graphics, IEEE Transactions on*, 20(1):1–1, Jan 2014.

[3] J. Kiess, D. Gritzner, B. Guthier, S. Kopf, and W. Effelsberg. GPU video retargeting with parallelized SeamCrop. In *Proceedings of the 5th ACM Multimedia Systems Conference*, MMSys '14, pages 139–147, New York, NY, USA, 2014. ACM.

[4] J. Kiess, B. Guthier, S. Kopf, and W. Effelsberg. SeamCrop: Changing the size and aspect ratio of videos. In *Proceedings of the 4th Workshop on Mobile Video*, MoVid '12, pages 13–18, New York, NY, USA, 2012. ACM.

[5] J. Kun. Seam carving for content-aware image scaling, Mar. 2013. http://jeremykun.com/2013/03/04/seam-carving-for-content-aware-image-scaling/.

[6] M. Rubinstein, D. Gutierrez, O. Sorkine, and A. Shamir. A comparative study of image retargeting. *ACM Trans. Graph.*, 29(6):160:1–160:10, Dec. 2010.

[7] M. Rubinstein, A. Shamir, and S. Avidan. Multi-operator media retargeting. In *ACM SIGGRAPH 2009 Papers*, SIGGRAPH '09, pages 23:1–23:11, New York, NY, USA, 2009. ACM.

[8] Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee. Optimized scale-and-stretch for image resizing. In *ACM SIGGRAPH Asia 2008 Papers*, SIGGRAPH Asia '08, pages 118:1–118:8, New York, NY, USA, 2008. ACM.

[9] Z. Yan and H. Chen. A study of image retargeting based on seam carving. In *Measuring Technology and Mechatronics Automation (ICMTMA), 2014 Sixth International Conference on*, pages 60–63, Jan 2014.