# Retargeting Techniques Using Seam Carving

Ramya Koneru, Miranda Aperghis and Jasen Hall

***Abstract*— In this paper we present two methods for image and video retargeting using seam carving. The purpose of retargeting is to resize images and videos for different aspect ratios, specifically reducing the dimensions while still retaining the important features of the image. Whereas cropping or scaling the image produces unwanted distortions in the important regions of the image, content-aware retargeting creates more aesthetically pleasing results by preserving the important content of the image. Seam carving is a technique applied in the field of retargeting and we will present two algorithms based on this method.**

***Index Terms*— Seam Carving, Graph Cuts, Image Retargeting, Video Retargeting, Backward Energy, Forward Energy, content-aware image resizing.**

## I. INTRODUCTION

Seam carving is used to perform resizing on images and videos by preserving important regions of the image or video. The challenge for retargeting is to perform resizing to produce aesthetically pleasing results, reducing distortions and maintaining the important features.

In this paper we present two different methods to perform seam carving for image and video retargeting. The first method uses dynamic programming based on the implementation in the paper by Avidan and Shamir, "Seam carving for content-aware image resizing"[1]. As we will show, our implementation of the algorithm performs well, however the algorithm used is limited to images. Therefore we will present a second method for seam carving using the theory of maximum-flow-minimum-cut for graph cuts. This video retargeting technique is based on the paper by Rubinstein, "Improved seam carving for video retargeting"[2].

There are many algorithms for image retargeting. Other papers present different techniques such as warping-based methods, patch-based methods and multi-operator methods[3]. Although there are many papers that claim that their algorithm performs the best for image retargeting, most results are based on qualitative comparisons with previous algorithms. The main goal of retargeting it to create aesthetically pleasing results and this can be determined by simple comparison of images produced by different methods. Therefore we compare our results with other seam carving methods.

This paper primarily focuses both on image and video retargeting. Although there has been plenty of research in the field of seam carving for image retargeting, there have been fewer papers written for seam carving for video retargeting. The additional challenge for video retargeting is to maintain the third dimension of temporal coherency. The state of the art paper is by Rubinstein that implements graph cuts. We will present the graph cut algorithm based on Rubinstein's work.

## II. SEAM CARVING

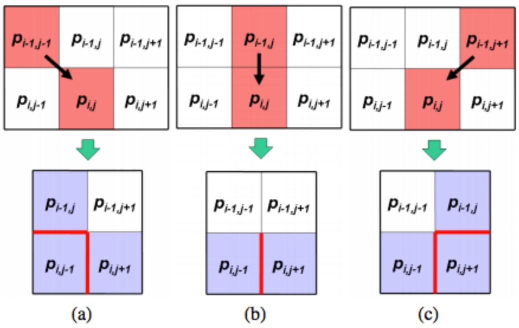
Seam carving is a popular approach in content-aware image and video resizing. The general method is to remove a seam of connected pixels from the image in either the horizontal or the vertical direction. A seam of pixels in the vertical direction is an 8-connected path of pixels from the top to the bottom of the image with a single pixel in each row. A horizontal seam is similar, but the seam follows the path from left to right and has a single pixel in each column. The removal of one vertical seam reduces the image size by one pixel in the horizontal dimension. When resizing a 480x320 image to 400x280 pixels, 80 vertical seams and 40 horizontal seams are removed.

To determine the best seam to remove we first calculate the gradient energy map of the image (Fig.1). This is done by using a filter such as the Sobel filter that measures the image gradient in both the x and y direction. The pixels with the lowest energy values are taken to be the least significant pixels in the image. For example, pixels within a region of blue sky have a much lower energy value than compared with pixels that make up a person's face in the foreground. The optimal seam to remove is determined using the backward energy as presented by Avian and Shamir[1]. The algorithm finds the seam with the lowest total from the gradient energy map and this is the seam that is cut from the image.

Figure1. Gradient Energy Map.

## III. Forward Energy

Forward energy considers the energy that will be added to the image once the seam is removed. When a seam is removed, neighboring pixels on either side of the seam are joined together creating new "pixel-edges". The cost of these pixel edges is measured as the forward differences between the pixels that become new neighbors after the seam is removed. There are three possible cases depending on the direction of the seam, whether it is the right pixel, left pixel or vertical pixel that is removed (Fig. 2).




$$(a)\ C_L(i,j) = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j-1)|$$
$$(b)\ C_U(i,j) = |I(i,j+1) - I(i,j-1)|$$
$$(c)\ C_R(i,j) = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j+1)|$$

Figure 2. Three possible vertical seam step costs for pixel $p_{i,j}$ using forward energy.

A cost matrix is calculated using dynamic programming as presented in Rubinstein's paper[2]. The seam with the minimum cost is then removed from the image.

Figure 3. Seam carving with backward energy (left). Seam carving with backward energy and importance diffusion (right).

Figure 4. Seam carving with backward energy and importance diffusion (left). Seam carving with forward energy (right).

## IV. Comparison of Forward Energy with Backward Energy

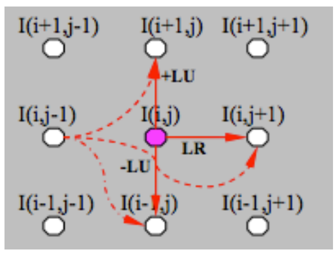
We performed seam carving with backward and forward energy on images (Fig. 3).

We further implemented backwards energy with using an importance diffusion map using the ideas from the paper by Cho and al[4]. The importance diffusion map adds weights to the new neighboring "pixel-edges" to decrease the chance that they will then be removed next. This is to prevent too many seams being cut from the same area that results in jittery and unpleasant seams. The results (Fig. 3) show that by using an importance diffusion map the image appears to be slightly better. However using forward energy in the results are greatly improved. This is also confirmed by Rubinstein's results.

## V. Video Retargeting

When implementing seam carving for video retargeting a new method has to be implemented. The dynamic programming approach that was implemented for images cannot be applied to video because we cannot resize each frame individually and still maintain temporal coherency. A new method presented by Rubinstein[2] uses the algorithm for maximum-flow-minimum-cut in graph theory.

Firstly we implemented the graph cut algorithm for single images. The implementation for the graph cut algorithm can be found in Rubinstein's paper and the paper by Yu and al[5]. A general idea of the algorithm is to find the minimum energy path from either left to right or top to bottom in the image for vertical or horizontal seams. Similarly to the dynamic programming approach, it assumes that the gradient of pixel values near the boundary of different objects is larger and therefore that pixels are more important to the image. The graph is created using the energy values of neighboring pixels using forward energy (Fig. 5).




$$\begin{cases} LR = |I(i,j+1) - I(i,j-1)| \\ +LU = |I(i,j-1) - I(i+1,j)| \\ -LU = |I(i,j-1) - I(i-1,j)| \end{cases}$$

Figure 5. Forward energy calculations for graph cut.

These values are then inserted into the graph structure where each pixel is taken to be a node in the graph, connected to neighboring pixels as shown in figure 6.

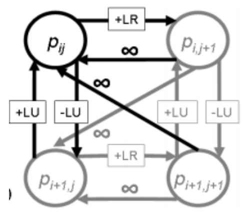



Figure 6. Graph construction using forward energy.

When applying the graph cut algorithm to video we want to maintain temporal coherency between frames. The seam to cut therefore depends on the next frames in the image. We calculate the new values to insert into the graph as a ratio of the current frame and the next frame gradient values. The ratios are determined by the value of alpha (see equations below).

$$LR_{pre} = \alpha \times LR_{pre} + (1-\alpha) \times LR_{next}$$
$$LU_{pre} = \alpha \times LU_{pre} + (1-\alpha) \times LU_{next}$$
$$-LU_{pre} = \alpha \times (-LU_{pre}) + (1-\alpha) \times (-LU_{next})$$

We tested different values of alpha but other papers suggest the optimal value to use is 0.7. Furthermore, we went on the test this algorithm by looking forward three frames instead of one (see equations below). We tested with alpha=0.5, beta=0.2, gamma=0.15 and delta=0.15.

$$LR_{pre} = \alpha \times LR_{pre} + \beta \times LR_{next} + \gamma \times LR_{next+1} + \delta \times LR_{next+2}$$
$$LU_{pre} = \alpha \times LU_{pre} + \beta \times LU_{next} + \gamma \times LU_{next+1} + \delta \times LU_{next+2}$$
$$-LU_{pre} = \alpha \times (-LU_{pre}) + \beta \times (-LU_{next}) + \gamma \times (-LU_{next+1}) + \delta \times (-LU_{next+2})$$

$$\alpha + \beta + \gamma + \delta = 1$$

## VI. Reults and Limitations

We found that for different values of alpha with a look ahead of one frame produced similar results for video retargeting. Furthermore a look ahead of three frames with varying ratios also produced very similar results. This leads us to conclude that the algorithm has some limitations and that a different method would have to be implemented to further improve our results for video retargeting.

When looking at our results for seam carving for image retargeting we found that the dynamic programming method with forward energy produced aesthetically pleasing results for images such as landscapes (Fig. 8). However as we have shown in the paper, images with high gradients throughout (Fig 1.) produce satisfactory but not perfect results (Fig. 4).

Figure 7. Original image.

a) Backward energy with importance diffusion

a) Forward energy

Figure 8

a) Our result for forward energy

b) Rubinstein's result for forward energy

Comparing our implementation for forward energy on images with Rubinstein's results are the same. These are the best results that we could find to compare with for forward energy using seam carving.

## VII. Conclusion

We have presented two methods for image retargeting and one method for video retargeting. We can clearly see that forward energy using dynamic programming produces better results than backward energy using dynamic programming.

Our implementation of graph cut for video retargeting produced satisfactory results however they were not as good as Rubinstein's results. In the future we would try to improve our graph cut algorithm for seam carving and look to include a better look-ahead algorithm that takes into account temporal coherency throughout a larger region of the video. Rubinstein suggests using preprocessing with pyramids to find the optimal regions of the frames to remove seams before performing the algorithm. We would look into implementing something similar to this.

We would also consider parallelizing the algorithm to make our code run faster. Graph cuts for video retargeting took a long time to compute.

In conclusion, we found that our implementation for seam carving for images to be more successful than seam carving for videos. Video retargeting is a much harder problem to solve because of the need for temporal coherency between frames.

## Acknowledgements

We used the code for graph cuts from the website http://pub.ist.ac.at/~vnk/software.html and followed the implementation in the paper "An Experimental Comparison for Min-Cut/Max-Flow Algorithms for Energy Minimization in Computer Vision.", by Yuri Boykov and Vladimir Kolmogorov.

## References

[1] Avidan, S. and Shamir, A., "Seam carving for content-aware image resizing," ACM Trans. on Graphics (Proc. of SIGGRAPH), (2007).

[2] Rubinstein, M., Shamir, A., and Avidan, S., "Improved seam carving for video retargeting," ACM Trans. on Graphics (Proc. of SIGGRAPH), (2008).

[3] Vaquero, D., Turk, M., Pulli, K., Tico, M., and Gelfand, N., "A survey of image retargeting techniques," University of California Santa Barbara, Nokia Research Center, (2010).

[4] Cho, S., Choi, H., Matsushita, Y., and Lee, s., "Image retargeting using importance diffusion", POSTECH, Microsoft Research Asia, (2009).

[5] Yu, S., Hua, S., and Fang, X., "Image and video retargeting using graph cuts", IEEE International Conference on Intelligent Control and Information Processing, (2010).