Generalized Multiscale Seam Carving

David D. Conger#1, Mrityunjay Kumar*2, and Hayder Radha#3

Dept. of Electrical & Computer Engineering, Michigan State University
East Lansing, MI, USA
1 congerd2@egr.msu.edu, 3 radha@egr.msu.edu

*Eastman Kodak Company
Rochester, NY, USA
2 mrityunjay.kumar@kodak.com

Abstract—With the abundance and variety of display devices, novel image resizing techniques have become more desirable. Content-aware image resizing (retargeting) techniques have been proposed that show improvement over traditional techniques such as cropping and resampling. In particular, seam carving has gained attention as an effective solution, using simple filters to detect and preserve the high-energy areas of an image. Yet, it stands to be more robust to a variety of image types. To facilitate such improvement, we recast seam carving in a more general framework and in the context of filter banks. This enables improved filter design, and leads to a multiscale model that addresses the problem of scale of image features. We have found our generalized multiscale model to improve on the existing seam carving method for a variety of images.

I. INTRODUCTION

As widespread use of mobile media devices continues to increase there remains a pressing desire for an effective means of resizing images to fit arbitrary screen sizes. Traditional methods such as cropping or resampling either remove important image features or introduce significant visual distortion. Avidan and Shamir [1] generated new interest in this area with the introduction of a content-aware image resizing technique called seam carving which iteratively removes/adds connected paths of pixels to achieve a desired target size. This was followed by the development of several other techniques ([4][11][13]) which employed global resizing rather than the iterative method used in seam carving. While these techniques produced good results, they are less scalable – an image must be reprocessed for each resolution change. Seam carving on the other hand enables one to store the locations of all seam paths removed/added in achieving some minimum/maximum resolution, allowing any resolution in between to be achieved by simply recalling the path locations and removing/repeating those pixels. For this reason and others, seam carving remains an effective image resizing tool. And although Rubinstein et al. [9] did improve on the original technique (as well as extend it to video), there are still cases where it has problems (cf. Figure 1 (b)).

In this paper we present a generalized seam carving algorithm from the perspective of filter banks and develop a multiscale analysis model. This enables the use of many different families of filters and leads to an improvement in seam selection for many images (cf. Figure 1 (c)). Section II details the proposed utility of filter banks for seam carving; section III discusses filter selection. Section IV describes the proposed multiscale model for seam carving. Section V presents our results and section VI gives concluding remarks.

II. GENERALIZED SEAM CARVING MODEL

To facilitate a more general seam carving model, we recast previous seam carving implementations in terms of 2D filter banks. Section II-A gives an overview of the most essential elements of the seam carving algorithm based on [1]. Section II-B develops the filter bank framework. Section II-C redefines backward-energy seam carving in this framework; section II-D redefines forward-energy seam carving.

A. Seam Carving Overview

A seam is defined as an 8-connected path of pixels from either top to bottom (vertical seam) or left to right (horizontal seam). In order to maintain an image’s rectangular shape, a seam contains exactly one pixel per row (vertical seam) or per column (horizontal seam). The optimum path is determined by generating a cumulative energy map $M$ and traversing from the bottom edge (or right edge) back to the top edge (or left edge)
edge), maintaining a connected path and selecting the minimum value at each row (or column). This path of pixels is the seam and is removed. The process then starts over, continuing until the desired resolution is met. While there are several seam carving operations – vertical seam removal/insertion, horizontal seam removal/insertion, combined vertical and horizontal seam removal/insertion – they are all based on the process of identifying a single seam. Therefore, for the sake of brevity, development of our model will be based on the selection of vertical seams only, but can be easily extended to the other listed operations.

B. Filter Bank Framework

Formally, let $\mathcal{B} = \{W^k\}_{k=0}^{K}$ be a filter bank where $W^k$ is an $h \times w$ mask. We define the result of filtering an $m \times n$ image $f$ with the mask $W^k$ as

$$d^k(x, y) = \sum_{i=-w/2}^{w/2} \sum_{j=-h/2}^{h/2} f(x + i, y + j)W^k(i, j)$$

which we will refer to as the $k$-th subband (or coefficient map). Note that this is the cross-correlation of $f$ and $W^k$ and not a convolution – a crucial distinction for the forward-energy seam carving implementation defined in section II-D.

Each coefficient map $d^k$ provides information about a particular image feature based on the structure of the mask $W^k$; image neighborhoods resembling the mask will result in large coefficient values while other neighborhoods will not. This leads to the idea that different combinations of coefficient maps can be used depending on the direction of a seam. Recall that, due to the nature of 8-connected paths, a seam can travel in one of three ways at each stop on its journey from edge to edge. For a vertical seam, at each row it can move down to the right, straight down, or down to the left. Therefore, it is possible to “consult” different coefficient maps for different directions. For convenience, we describe this discretionary ability using vector products. We define

$$\mathbf{D}_{ij} = q(d^0(i, j), d^1(i, j), \ldots, d^K(i, j))$$

where $q$ is an element-wise operator used for normalization. Unless otherwise stated, we define $q$ based on the $L_1$-norm: $(q(A))_{ij} = |a_{ij}|$ where $a_{ij}$ is the $i,j$-th element of the matrix $A$. This normalization is used to ensure that coefficients do not cancel and essentially typifies the energy metric.

How and when the different coefficient maps are used comes into play during the generation of the cumulative energy map $M(i, j)$ as defined in [2][9] and generalized here for vertical seam removal:

$$M(i, j) = \min \left\{ M(i-1, j-1) + \mathbf{D}_{ij}p_R, M(i-1, j-1) + \mathbf{D}_{ij}p_C, M(i-1, j+1) + \mathbf{D}_{ij}p_L \right\}$$

where $p_R$, $p_C$ and $p_L$ are $K \times 1$ vectors whose $k$-th elements define the weighting factor of the $k$-th coefficient map. Essentially, these guidance vectors – as we will refer to them – designate which coefficient maps are used in “steering” a seam down to the right, straight down or down to the left, respectively. The cumulative energy map can be defined similarly for horizontal seam removal; however, as mentioned above we will limit our discussion to the vertical case.

C. Backward-Energy Seam Carving Filters

The original seam carving algorithm presented in [1] (and which was later named backward-energy in [9]), uses a single energy map – primarily the $L_1$-norm of the image gradient – to generate the cumulative energy map $M$. As there is more than one way to implement the gradient for discrete-time signals, we define the backward-energy filter bank in terms of general derivative masks for the vertical ($H_V$) and horizontal ($H_H$) directions:

$$\mathcal{B}_{\text{back}} = \{H_V, H_H\}$$

with $p_R = p_C = p_L = [1, 1]^T \pm p$. One can easily verify that

$$\mathbf{D}_{ij}p = |d^0(i, j)| + |d^1(i, j)|$$

which is the $L_1$-norm of the image gradient at the point $(i, j)$. Notice that since $p_R = p_C = p_L$, there is no distinction between the coefficient maps for different seam directions.

D. Forward-Energy Seam Carving Filters

The forward-energy seam carving algorithm defined in [9], is described in terms of the absolute pixel differences between disjoint pixels brought together by seam removal. Specifically, this is implemented directly into their definition of the cumulative energy map for vertical seam removal\footnote{Note that we have presented this with slightly different notation in terms of what is left and what is right, and have separated the cost $C_L(i, j)$ from the left and right costs in order to facilitate developing the filters.}:

$$M(i, j) = \min \left\{ M(i-1, j-1) + \mathbf{D}_{ij}p_R, M(i-1, j-1) + \mathbf{D}_{ij}p_C, M(i-1, j+1) + \mathbf{D}_{ij}p_L \right\}$$

$$\mathbf{D}_{ij} = q(d^0(i, j), d^1(i, j), \ldots, d^K(i, j))$$

$$\mathcal{B}_{\text{back}} = \{H_V, H_H\}$$

with $p_R = p_C = p_L = [1, 1]^T \pm p$. One can easily verify that

$$\mathbf{D}_{ij}p = |d^0(i, j)| + |d^1(i, j)|$$

which is the $L_1$-norm of the image gradient at the point $(i, j)$. Notice that since $p_R = p_C = p_L$, there is no distinction between the coefficient maps for different seam directions.

D. Forward-Energy Seam Carving Filters

The forward-energy seam carving algorithm defined in [9], is described in terms of the absolute pixel differences between disjoint pixels brought together by seam removal. Specifically, this is implemented directly into their definition of the cumulative energy map for vertical seam removal\footnote{Note that we have presented this with slightly different notation in terms of what is left and what is right, and have separated the cost $C_L(i, j)$ from the left and right costs in order to facilitate developing the filters.}:

$$M(i, j) = \min \left\{ M(i-1, j-1) + \mathbf{D}_{ij}p_R, M(i-1, j-1) + \mathbf{D}_{ij}p_C, M(i-1, j+1) + \mathbf{D}_{ij}p_L \right\}$$

$$\mathbf{D}_{ij} = q(d^0(i, j), d^1(i, j), \ldots, d^K(i, j))$$

$$\mathcal{B}_{\text{back}} = \{H_V, H_H\}$$

with $p_R = p_C = p_L = [1, 1]^T \pm p$. One can easily verify that

$$\mathbf{D}_{ij}p = |d^0(i, j)| + |d^1(i, j)|$$

which is the $L_1$-norm of the image gradient at the point $(i, j)$. Notice that since $p_R = p_C = p_L$, there is no distinction between the coefficient maps for different seam directions.

D. Forward-Energy Seam Carving Filters

The forward-energy seam carving algorithm defined in [9], is described in terms of the absolute pixel differences between disjoint pixels brought together by seam removal. Specifically, this is implemented directly into their definition of the cumulative energy map for vertical seam removal\footnote{Note that we have presented this with slightly different notation in terms of what is left and what is right, and have separated the cost $C_L(i, j)$ from the left and right costs in order to facilitate developing the filters.}:
Another observation gleaned from our model is that the guidance vectors can take on different weights to produce interesting results; we discuss this in section III-C.

A. Texture Filters

A significant problem with seam carving is its sensitivity to texture. Often there may be an image that contains a relatively smooth object set against a busy texture such as in Figure 2 (a). Using seam carving, the texture is mostly avoided and many seams pass through the person’s hair and face, causing significant distortion (Figure 2 (b)). One way to address this problem is in terms of scale (which we discuss in section IV). Another approach is through the use of filters which are sensitive to texture, such as Laws’ texture masks or Gabor filters [3][7]. Another example is shown in Figure 2, where we used a Gabor filter with 90° phase, one octave bandwidth, and a center frequency of $\sqrt{2}/2^4$ (Figure 2 (c)) – let $G_1$ be the mask associated with this filter. By augmenting this onto the forward-energy filters (i.e., $\mathcal{B} = \mathcal{B}_{\text{fwd}} \cup \{G_1\}$) and assigning it a negative weight for diagonal directions, i.e.,

$$p_R = [1,1,0,1]^T \quad p_C = [0,1,0,0]^T \quad p_L = [0,1,1,1]^T$$

we can “guide” the seams through the texture and mostly avoid the person (Figure 2 (d)).

Of course, there are other Gabor filters that could have been used, but this example is mainly meant to illustrate how incorporating a texture filter can improve the results.

B. Roberts Masks

As hinted above, the Roberts masks can also be used as an alternative to the forward-energy filters. Here, the goal is more about using simple filters than necessarily improving the results. Formally, we define

$$\mathcal{B}_{\text{Roberts}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

with guidance vectors

$$p_R = [3,1,1,0]^T \quad p_C = [1,1,1,0]^T \quad p_L = [1,3,1,1]^T$$

As in section II-C, one can easily verify that

$$D_{ij}p_R = |d^0(i,j)| + |d^1(i,j)| = C_R(i,j) + C_C(i,j)$$

$$D_{ij}p_C = |d^1(i,j)| = C_C(i,j)$$

$$D_{ij}p_L = |d^1(i,j)| + |d^2(i,j)| = C_C(i,j) + C_L(i,j)$$

C. Guidance Vector Weighting

The guidance vectors control the weighting for each subband for a given direction; for general seam carving, they will usually be designed like the ones defined above. However, there may be cases where one might want to bias a particular direction. For example, compression methods using seam carving have been proposed which involve storing the
A. Energy Accumulation Map

Having filtered the image using the cascaded filter bank, this brings us to the challenge of utilizing the information across multiple scales in some meaningful way. Up to this point we have not explicitly required that $W^0$ be a low-pass filter or that the remaining masks be high-pass filters (or bandpass filters covering higher frequencies); however, to facilitate our discussion, we will assume that this is the case from here on out. At each scale, the high-pass coefficient maps characterize the energy of the different features corresponding to each filter. Finer scales represent the fine details of an image – the highest frequency content – while coarser scales represent broader changes in the image. Consequently, the finest scales will be more susceptible to fine textures and noise which may be visually unimportant.

This is exactly the case in Figure 1 (b) – the horse is much smoother than the grass and therefore makes for a less resistant path. Even though the edges of the horse produce high energy, it is not enough to make the cumulative path energy higher than a path through the grassy region. As we discussed above, the seam carving methods described in section II have the tendency to weight the fine details too heavily. On the other hand, it neither makes sense to give a lot of weight to the coarse details. Still, the goal is to give weight to significant edges and, as it is well known, such edges will span several scales. In particular, the well-established results by Mallat and Zhong have shown that edges appear across scales as modulus maxima, with the finest scale giving the precise location of the edge; as the scale increases, the edge diffuses [5][6]. This motivates the idea that the importance of some image point can be characterized not only by the significance of its energy at a certain scale, but also by the number of scales in which it is significant.
Consider the two different image regions in Figure 5 taken from the horse image. As the scale increases, the coefficients for the grassy region quickly decay, becoming nearly zero. In contrast, the coefficients for the region containing the horse’s head are significant at all four scales.

This brings us back to the challenge of combining the information across scales. While there are likely many ways to do this, we have found that a simple linear combination of the coefficient maps works quite well. The one challenge, of course, is that a coefficient map at level \( j \) is \( L \) times the size of one at level \( j + 1 \) and therefore must be upsampled. This is implemented as shown in Figure 7. At each level, the coefficient map \( d_j \) is normalized using \( q \), weighted by \( \alpha_j \), upsampled in both dimensions by \( L \), low-pass filtered, and added to the weighted, normalized coefficient map at level \( j - 1 \). Formally, the combination of levels \( j \) and \( j - 1 \) can be expressed as

\[
\tilde{d}_{j-1}(x, y) + \sum_{a=-w/2}^{w/2} \sum_{b=-h/2}^{h/2} d_j \left( \frac{x+a}{L}, \frac{y+b}{L} \right) W(a, b)
\]

where \( \tilde{d}_j \doteq \alpha_j q(d_j) \). If downsampling was implemented in the cascaded filter bank (see Figure 4), then we also upsample and filter at the last level to ensure that the output matches the size of the image \( f \). We refer to the final map \( D^k \) as the energy accumulation map corresponding to the filter \( W^k \) (cf. Figure 6). Since it represents \( N \) scales of the coefficient map \( d^k \), we now redefine

\[
D_{ij} \doteq [D^0(i,j), D^1(i,j), ..., D^k(i,j)]
\]

Note that the normalization function \( q \) is absent in this definition since we have included it in the generation of the energy accumulation map. From this point, the rest of the algorithm proceeds as described in section II.

C. Design

While this may appear to be a fairly rigid method of utilizing the energy across scales, there are a few things that give it some flexibility. First, while there are no hard and fast rules on the number of levels \( N \), the practical limit is given by

\[
N \leq \left\lfloor \log \min\{m, n\} - \log(\min\{h, w\} - 1) \right\rfloor / \log L
\]

where \( \lfloor \cdot \rfloor \) is the floor function. Second, the selection of the weight factors \( \alpha_j \) can be used to give more importance to certain scales if one is interested in preserving a certain spatial frequency. For our purposes, we have generally given each scale equal importance, which requires something of the form \( \alpha_j = L^{-j} \). Also, the operator \( q \) can be adjusted to include normalization of each coefficient map, which we have done in our simulations. Specifically,

\[
q(A) = \frac{1}{\max_{i,j} |a_{ij}|} \left[ \begin{array}{c} \vdots \\ a_{ij} \\ \vdots \end{array} \right]
\]

where \( a_{ij} \) is the \( i,j \)-th element of the matrix \( A \). Finally, another option is that instead of linearly combining each scale, one could employ a product or some other non-linear operator. However, as mentioned above, we have found a linear combination to work well and have thus used it throughout this paper.

V. RESULTS

In Figure 8 we compare uniform resampling and forward-energy seam carving to our multiscale seam carving method for the various filter banks defined above. Overall our method was more capable of avoiding important image features, and was less sensitive to fine texture. The top image shows a dramatic improvement: uniform resampling creates obvious distortion of the person’s face and forward-energy seam carving nearly removes the face entirely; with either Roberts filters at four scales or the forward-energy filters at three scales, our method was able to preserve the person’s face. The middle image showed similar results: with the forward-energy filters at either three or four scales, the white horse is preserved. The bottom image is a particularly difficult case because of the many different objects and textures in it;
uniform resampling and forward-energy seam carving again suffer the same fate—obvious distortion and undesirable seam selection, respectively. Using the backward-energy filters at four scales, the duck and the person’s face are avoided—the primary objects in the image—but the person’s side is removed. The result is similar for the Roberts filters at six scales, except that the removal of the person’s side is less noticeable because it happens near the image edge. In either case, this is a notable improvement over the prior methods. Overall, we have found the Roberts filters (at around five levels) and the forward-energy filters (at around three to four levels) to produce the best results.

VI. CONCLUSIONS

We have presented a new framework for seam carving based on filter banks. This generalized model encompasses the original implementations and provides for more design flexibility. In particular, it lays the groundwork for incorporating many different filters and their various combinations via the guidance vectors. Finally, we extended this model to multiple scales to reduce sensitivity to noise and improve seam selection. The result is a more robust seam carving algorithm.

REFERENCES


