Image Seam Carving Based on Content Aware Resizing by Gradient Method

Matheel Emaduldeen* , Ph.D (Asst.Prof.)
Rana Mohammed Hassan*

Abstract

In this paper, bilateral filter and seam carving is implemented to get an image that is retargeted to a new size and has a clear appearance. Bilateral filter is used to enhance images by smoothing texture and preserving edges. The implemented seam carving for content aware image resizing gives better results in resizing images than the standard scaling and cropping techniques. Also, six types of edge detection filters are used in this paper to preserve the energy of image objects, and evaluation for these filters is done to find the best energy preserver among those six types.

Keywords: Bilateral filter , Seam Carving, Edge Detection.

*University of Technology

1. Introduction

Effective resizing of images should not only use geometric constraints but consider the image content as well. For exemple, making an image fit with a web page layout, magazine cover or displaying devices without affecting the content is very important. Applications of content aware resizing allow us to change the general aspect of images while preserving important image content for both reduction and enlargement of images [1]. The goal of this paper is to retarget an image to a new size without effecting on the content of the image using operator called “seam carving” and choosing the path that goes through sections in the image that have the least amount of information. Standard image scaling is not sufficient since it is oblivious to the image and typically can be applied only uniformly. Cropping is limited since it can only remove pixels from the image periphery.

The image operator, seam-carving, can change the size of an image by gracefully carving out or inserting pixels in different parts of the image. Seam carving uses an energy function defining the importance of pixels. In this paper, six types of edge detection filters were used to detect the energy of objects in an image.

When using seam carving in resizing a noisy image, seam paths will carve along the least important pixels of the image (least energy) so the noise will be preserved since it has high energy. Pre-processing using bilateral filter to enhance noisy images is necessary in this paper. The reason behind using bilateral filter with seam carving in image resizing is that it has the property of smoothing texture while preserving edges. Seam carving can be used for image content enhancement and object removal [1].

2. Related Work

Resize the image is a standard tool in a lot of image processing applications. It works by uniformly resize the image to a target size. A recent days, there is a growing interest in image retargeting that seeking to change in the size of the image while keeping the important features intact, where these features may be as for detected top-down or bottom-up. Top down methods used tools such as face detectors to detect important regions in the image [2], while that bottom-up methods depends on visual saliency methods for the construction a visual saliency map of the image. Once the saliency map is building, cropping can be used for display the most important region of the image [3]. Avidan and Shamir [1] suggested change resizing of still images by using “Seam Carving for Content-Aware Image Resizing”, the seam define is least-energy paths. Seam carving for image used dynamic programming to find the optimum seam iteratively and retargeting. Seam removal is used to reduce the size of an image; also expansion is the selected lowest-cost seam to inserting in order to expand an image. Lastly, the user represents the region for object removal and the system can automatically calculate seams to remove from the image even if all marked pixels disappear . F.Shafieyan, N.Karimi, Ebrahim N.Esfahani and S.Samavi [4] proposed seam carving algorithm by using new combined energy map (Gradient energy map, Saliency map, Depth map) and the output is energy map formation, and then selected seam to obtain the target image. The proposed method gets well visual appearance, preserving strong edges, reducing distortion and maintaining geometrical important contents and structures. Bilateral filter grid was proposed by Chen.J, et al. [5] to combine the notion of bilateral filter and content aware. By working in the bilateral grid, algorithms such as edge-aware painting, bilateral filtering and local histogram equalization to be simple manipulations that are both independent and local.

3. Bilateral filter

“Bilateral Filter is a non-linear technique that can blur an image while preserving strong edges” [6]. Bilateral filter was used in this project in order to enhance the image and remove the noise when needed so the retargeting results a clear image.Bilateral filter was implemented according to the following formula:

$$\text{BF}\ [I]_p = \frac{1}{W_p} \sum_{q\in S} G\sigma s\ (\|p-q\|)\ \ G\sigma s\ (|I_{p-}\ I_q|)\ I_p \qquad (1)$$

Where: $\|p-q\|$ is Space Denoise, $|I_{p-}\ I_q|$ is Range Feature Preserving and W_p is normalization.

$$W_p = \sum_{q \in S} G\sigma s\,(\|p-q\|)\;\; G\sigma s\,(|I_{p-}\, I_q|) \tag{2}$$

Gaussian formula:

$$\frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(X-\mu)^2}{2\sigma^2}} \tag{3}$$

Where, μ is the mean. The parameter σ is its standard deviation with its variance then σ^2.The space part (Gσs) can be found by applying Gaussian formula to each entry in the Euclidean matrix. Euclidean matrix can be found by computing Euclidean distance for each center pixel position (p) and each of its neighbor positions (q) in a kernel of size 5*5.For the range part (Gσr), Gaussian formula can be applied on the difference between each center pixel (Ip) in the 5*5 kernel and each of its neighbors (Iq). Parameters σs and σr will specify the amount of filtering for the image I.Each neighbor is weighted by a spatial component that penalizes distant pixels and range component that penalizes pixels with different intensity. The combination of both components ensures that only nearby similar pixels contribute to the final result [6].

4. Seam Carving for Content-Aware Image Resizing Algorithm

Seam carving was implemented according to algorithm proposed by Avidan and Shamir [1]. The main functions related to this part of the project are:

4.1 Energy Gradient

The goal in this section is to find the least noticeable pixels so those pixels can be used to be removed or inserted and blended with their surroundings. The least amount of information in an image can be computed using three methods:

1- Gradient (Faster).

2- Entropy.

3- Histogram of Oriented Gradients (HOG).

Edge detection is the process lack of continuity or sudden change in some visual properties (such as the color, intensity of lighting, composition) and treatments and the processes is very important to the understanding of images and the analysis can be used in the process of distinction objects in digital images[7].

Edge detection is traditionally performs by convolving in computer vision the signal with some kind of linear filter usually a filter that approximation a first or second derivative operator [8]. There are six methods to obtained edges using gradient G of intensity change [I(x,y)].

1- Sobel.

2- Prewitt.

3- Discrete Laplacian.

4- Frei – Chens.

5- Pyramid.

6- First order derivative of Gaussian.

When convolving these filters with an image, the obtained result is gradient along x and y direction. Once getting those gradients, magnitude is computed using the following formula

$$e_1(\mathbf{I}) = |\frac{\partial}{\partial x}\mathbf{I}| + |\frac{\partial}{\partial y}\mathbf{I}| \qquad (4)$$

4.2 Select the seam [9] [10]

The user must choose whether wants carve horizontal seams, vertical seams, or a combination of both horizontal and vertical seams. Let L be an n × m image.

A vertical seam is computed as:

$$\mathbf{s}^{\mathbf{x}} = \{s_i^x\}_{i=1}^{n} = \{(x(i), i)\}_{i=1}^{n}, \text{ s.t. } \forall i, |x(i) - x(i-1)| \leq 1, \tag{5}$$

Where x is a mapping x: [1, . . . , n] [1, . . . ,m], a vertical seam is an 8-connected path of pixels in the image from top to bottom. And Containing one, and only one, pixel in each row of the image. The pixels of the vertical seam {Si} thus be

$$L_S = \{L(s_i)\}\, i^n = 1 \;\; \{L(x(i), i)\}\, i^n = 1 \tag{6}$$

A horizontal seam is computed as:

$$\mathbf{s}^{\mathbf{y}} = \{s_j^y\}_{j=1}^{m} = \{(j, y(j))\}_{j=1}^{m}, \text{ s.t. } \forall j |y(j) - y(j-1)| \leq 1. \tag{7}$$

Where y is a mapping y: [1, . . . , n] [1, . . . ,m], a horizontal seam is an 8-connected path of pixels in the image from left to right. And containing one, and only one, pixel in each Column of the image.The pixels of the horizontal seam {Si} is computed as:

$$L_S = \{L(s_j)\}\, j^m = 1 \; \{L(y(j), j)\}\, j^m = 1 \tag{8}$$

4.3 Compute Seam Costs [1]

The cumulative minimum energy was computed for all possible connected seams at each entry (i,j) using dynamic programming to find lowest energy approach according to the following:

1- Forward Pass (top to bottom for finding seam)

Forward formula computes the least energy to be removed .forward formula is:

$$C_L(i,j) = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j-1)| \tag{9}$$

$$C_V(i,j) = |I(i,j+1) - I(i,j-1)| \tag{10}$$

$$C_R(i,j) = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j+1)| \tag{11}$$

$$M(i,j) = \min \begin{pmatrix} M(i-1, j-1) + C_L(i,j) \\ M(i-1, j) + C_V(i,j) \\ M(i-1, j+1) + C_R(i,j) \end{pmatrix} \qquad (12)$$

2-Backward Pass (bottom to top for finding seam)

Backward ward formula computes the cost of removing seams that insert the least energy to an image. Backward formula is

M(i, j) = Energy(i, j) + min(M(i-1, j-1),M(i-1, j),M(i-1, j+1)) (13)

5. Proposed Algorithm Implementation

As a result, the goal is to implement the seam-carving for content aware resizing to know how the seam - carving operator is working and study the effect of using different edge detection filters in choosing the connected path along columns or rows. To summarize, the main contributions in this paper are:

- Bilateral filter as a pre-processing and post processing.
- Seam carving for content-aware image resizing algorithm
- Evaluation for the six edge detection filters.

As show, figure (1) represent the implementation of the proposed algorithm .

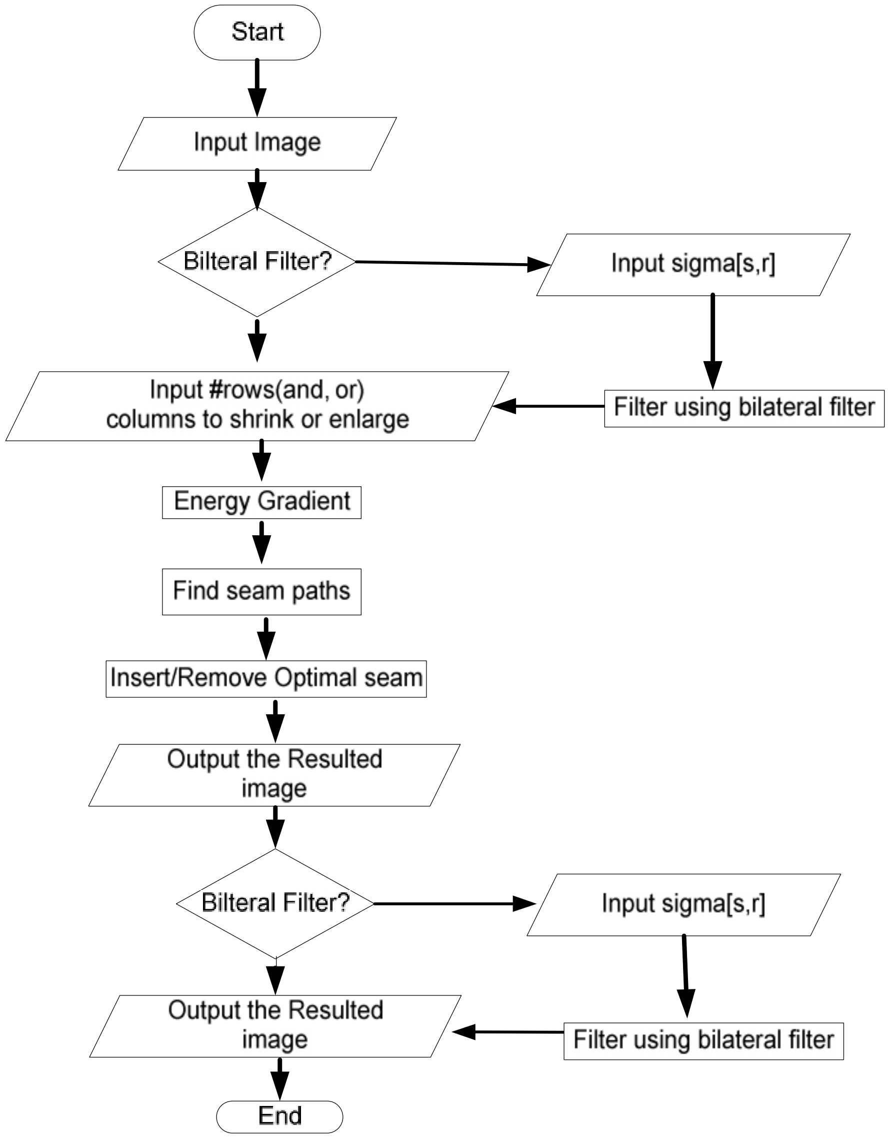


Figure (1): Flowchart describes procédures for the proposed implementation algorithm

5.1 Bilateral filter as a pre-processing and post processing.

Use the Bilateral filter before working seam carving and after using seam carving because to smooth image. The reason of choosing kernel size of 5 is because kernel size of 3*3 shows less smoothing effects while kernel sizes larger than 5*5 leave high smoothing effects.

When retargeting an image that has some sort of noise that is not uniformly distributed, the seam carving retargeting results in a worse image because noise is considered high energy so the seam operator will bypass noisy pixels.As a result pre-processing using bilateral filter is necessary in preserving edges since edges are important in retargeting.

5.2 Seam carving for content-aware image resizing algorithm

What has been used in this project is the gradient method because it is fast. The edge detection filters that were used to find the energy are Sobel , Prewitt, Discrete Laplacian, Frei – Chens, Pyramid, First order derivative of Gaussian.When convolving these filters with an image, the obtained result is gradient along x and y direction, magnitude is computed energy, as show in Figure (2):

Input: Image , Sobel mask horizontal and vertical operators // im, H_x , H_y
Output: Magnitude image // mag.
Step1: Start Step2: Read image matrix and convolution with H_x and returns the centric portion of the convolution of the similar size as image to find image gradient = g_x Step3: Read image matrix and convolution with H_y and returns the centric portion of the convolution of the similar size as image to find image gradient = g_y Step4: mag = sqrt ($g_x * g_x + g_y * g_y$) Step5: Return mag ; Step6: End.

Figure(2): The algorithme of compute magnitude

5.3 Compute Seam Costs

The second stage is to pass the frame from the second row to the last row the cumulative minimum energy was computed for all thinkable related seams at all entry (i,j) using dynamic programming approach according to the following forward .After computing M seams keep track the minimal values. The algorithm that illustrated steps to finding forward energy, as show in Figure (3):

Input: energy matrix for image // cost(x)
Output: forward energy calculated matrix for frame // [seam]

Step1: Start
Step2: Read size of energy matrix and return number of rows, number of columns// [m n]=size(x)
Step3: Set matrix on all rows and columns equal to zeros// seam=zeros(m,n)
Step4: The first in the energy matrix put in the seam matrix // Seam (1, :) =x (1, :);
Step5: for i = second row to the last row
Step6 : for j = first column to the last column
Step7: Checking if j-1=0, find minimum from north and north east pixels // Seam(i,j)=x(i,j)+min([Seam(i-1,j),Seam(i-1,j+1)]);
Step 8: Else
Step 9 : Checking if j+1>n, find minimum from north and north west pixels // Seam(i,j)=x(i,j)+ min([Seam(i-1,j-1),Seam(i-1,j)]);
Step 10 : Else
Step 11: Find the least inserted energy
Column left=abs(x(i,j+1)-x(i,j-1))+abs(x(i-1,j)-x(i,j-1)) // Cl
Column right=abs(x(i,j+1)-x(i,j-1))+abs(x(i-1,j)-x(i,j+1)) // Cr
Column vertical=abs(x(i,j+1)-x(i,j-1)) // Cv
Compute the s(i,j)=min([(s(i-1,j-1)+cl),(s(i-1,j)+cv), (s(i-1,j+1)+cr)]);
Step12: Cl=0; Cr=0; Cv=0;
Step 13: return to step 5 to find all the rows in the seam array.
Step 14: End.

Figure (3): The algorithm of Forward energy

5.4 Optimal Seam Path and remove

The last stage is to select the optimal seam path from the minimum value in the last row that refer to end of the minimum of connected vertical seam, and remove the whole column with lowest energy in each row. Backtracking about that minimum energy was posted to find the optimal path seam for deleted. This idea shown in Figure (4) illustrates the steps to find the optimal seam path and illustrates the steps to delete this seam as show in Figure (5):

Input: **:** forward energy matrix for image // [seam]
Output: Optimal seam path // k and store the index of minimum column //y is track

Step1: Start
Step2: Read matrix for forward energy and return number of rows, number of columns// [m n]=size(mag)
Step3: Create matrix contains the one column and many rows equal zeros// y= zeros(1,m)
Step4: Find the minimum cost from the last row of cost array because the cost calculation is from top to down(forward) using dynamic programming
Step5: Compute minimum of (the north, north east, north west) for each pixel in order to store the optimal connected minimum path along the image .
for i=m-1 to 1 step -1
Step6: Index the current column (i=m-1 from bottom to up calculation to find optimal seam path)// j=y(i+1)
Step7 : Check if the col=1, take the south and south east of the pixels in column 1// (j==1)
if (seam_cost(i,j) <=seam_cost(i,j+1))
    y(i)=j;
else
    y(i)=j+1;
end
Step8: Check if the col=n, take the south and south west of the pixels in column n // (y(i+1)==n) // if (seam_cost(i,j) <=seam_cost(i,j-1))
    y(i)=j;
else
    y(i)=j-1;

```
                    end
Step 9 : if (seam_cost(i,j) <=seam_cost(i,j-1))
                  if (seam_cost(i,j) <= seam_cost(i,j+1))
                    y(i)=j;
                  else
                   y(i)=j+1
                  end
Step 10: if (seam_cost(i,j-1)<= seam_cost(i,j+1))
              y(i)=j-1;
           else
             y(i) = j+1;
Step 11: return to step 5 to find  the optimal seam row
Step 12: End.
```

Figure(4): The algorithm of optimal seam path .

Input: : Optimal seam path // k and store the index of minimum column //y is track
Output: Remove the optimal pixels with lowest energy //k
Step1: After saving the index of the current column, (i=m-1 from bottom to up calculation to find optimal seam path)// j=y(i+1) **Step2 :** for i = 1 to last row **Step3:** If the pixel to be removed is in the first column skip the first column and copy from the second column **Step4:** draw the seam store black pixel // rgb_img(i,1,:)=0; **Step5:** Else **Step6:** If the pixel to be removed is in the last column skip the last column and copy until last column. **Step7 :** draw the seam store black pixel // rgb_img(i ,n,:)=0; **Step 8:** Else **Step 9 :** copy until reaching the specified column of the pixel to be

removed and skip then restart the copy from the next column

Step 10: draw the seam store black pixel // rgb_img(i, y(i) ,:)=0;

Step 11: return to step 2 until to remove all optimal seam path.

Step 12: End.

Figure(5): The algorithm of remove seam

To shrink an image by removing K vertical seams, in the first iteration, find a path along the cumulative minimum energy and remove that path, then shift all the pixels in the image to the left. The output of one iteration will be the input of the next iteration. Also, to shrink an image horizontally, it is the same process of shrinking vertically but by transposing the input image and the output image.

To enlarge an image vertically, the same process of finding K seams is applied, but instead of removing those paths, new pixels are inserted to the image as the average of current pixel in the seam path and its right and left pixel in the image. Also, the same process of enlarging vertically is applied to enlarge horizontally, but by transposing the input image and the final output image.

Vertical seam path Horizontal seam path

Figure (6): an example of vertical and horizontal seam path to be removed or inserted

5.5 Evaluation Measuring Performance using Confusion Matrix

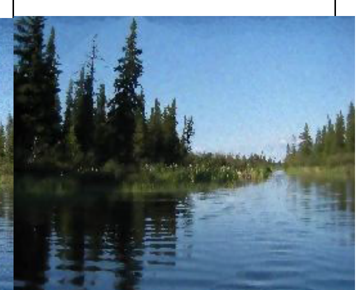
The definite table planning that allows conception of the presentation or the accuracy of an algorithm is called "Confusion matrix" . All column of the matrix represents the cases in a predicted class, though all row denotes the cases in an real class. "Confusion matrix" is a table that has four elements that determination the number of False Positives (FP), False Negatives (FP), True Positives (TP), and True Negatives (TN). Many further calculations can be taken from these elements. So

$$\text{Accuracy} = (TP + TN) \; / \; (Tp + TN + FP + FN) \qquad (14)$$

In this paper confusion matrix is used to evaluate the energy detected by the six energy detection filter.

6. Expérimental Results and Evaluations

a- Using Bilateral Filter as a pre-processing and then retarget images is show in Figure (7).

Input Image	**Retargeted without Applying BF**	**Retargeted after Applying BF**
 **(450*600)**	 **(250*300)**	 **[s=10 r=0.1] (250*300)**

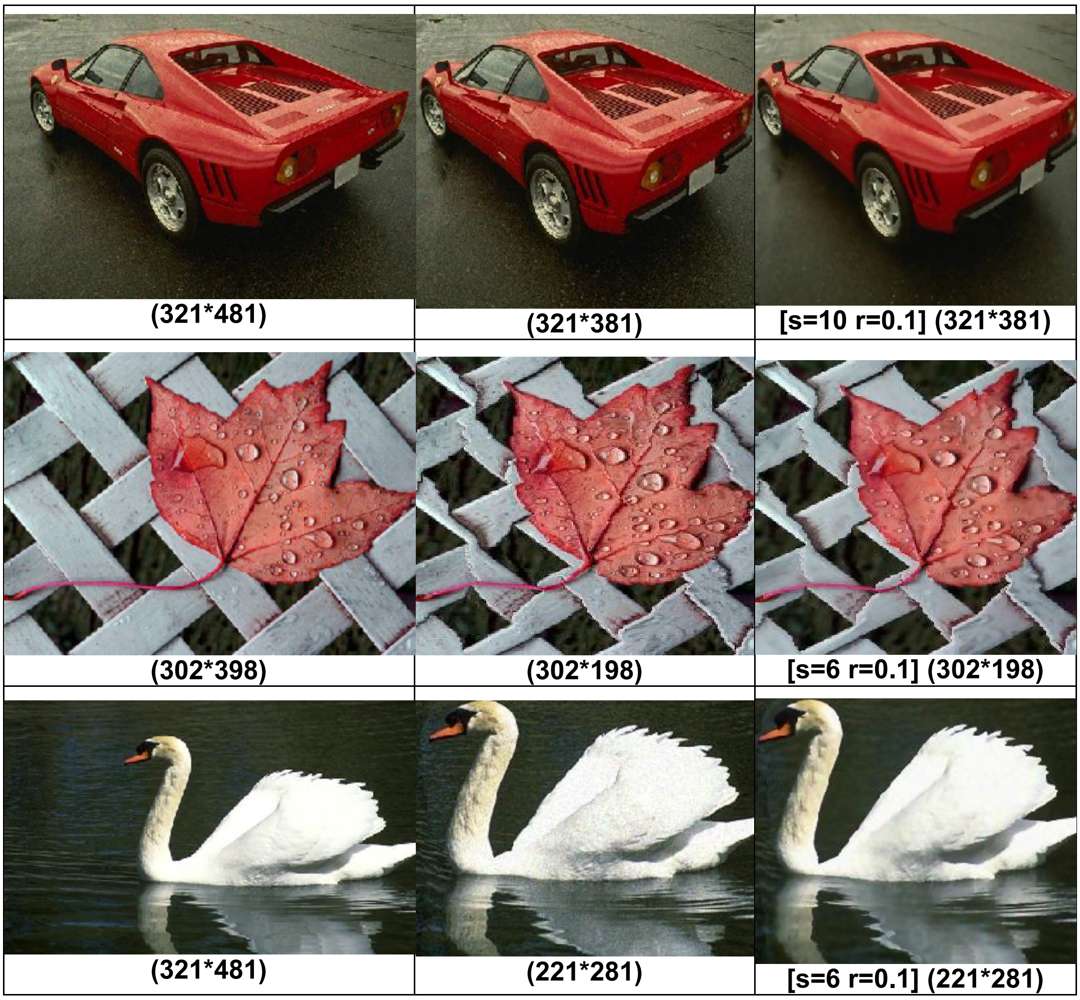
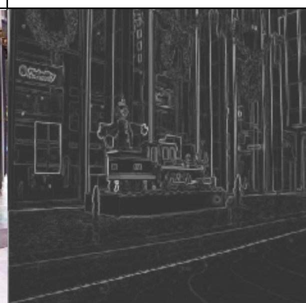
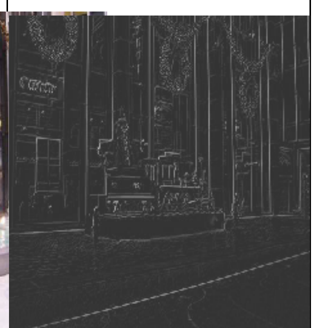
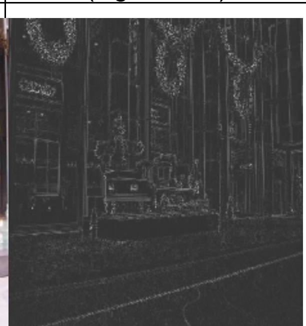


Figure (7): Retargeted images after preprocessing using bilateral filter b- Several outputs of retargeting (shrinking) images along columns and/or rows are show in Figure (8).

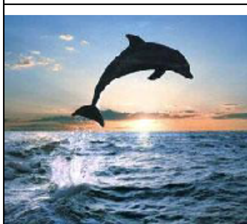
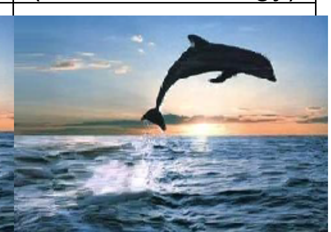
Input Image	Energy	Retargeted Image (Forward energy)	Retargeted Image (Backward energy)
 (375*500)	 Sobel Edge Detection Filter	 (375*300)	 (375*300)
 (375*500)	 Gaussian filter (sigma=0.2)	 (375*300)	 (375*300)
 (375*500)	 Discrete Laplacian Filter	 (375*300)	 (375*300)

 (500*400)	 Sobel Filter	 (300*400)	 (300*400)
 (500*400)	 Gaussian filter (sigma=0.2)	 (300*400)	 (300*400)

			
(500*400)	Discrete Laplacian Filter	(300*400)	(300*400)

Figure (8): Retargeted images by removing vertical and/or horizontal seam paths

c- Several outputs of retargeting (enlarging) images along columns and/or rows are show in Figure (9).

Input Image	Energy	Retargeted Image (Forward energy)	Retargeted Image (Backward energy)
			
(200*239)	Sobel Edge Detection Filter	(200*497)	(200*497)

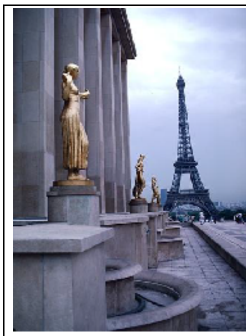
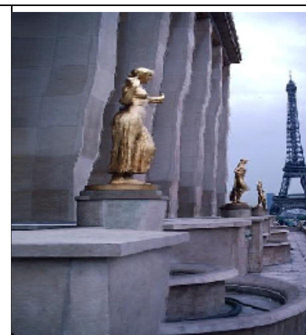
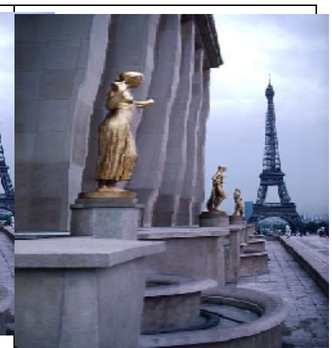
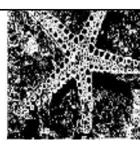
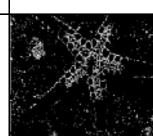
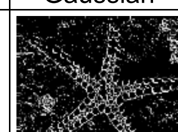
 (481*321)	 Discrete Laplacian Filter	 (481*521)	 (481*521)

Figure (9): Retargeted images by inserting vertical and/or horizontal seam paths

d- The following tables show statistics from matching outputs of six different edge detection filters for first and second tested images.

- **First image:**

Ground Truth	Sobel	Prewitt	Discrete Laplacian	Frei - Chens	Pyramid	First order derivative of Gaussian
						

Figure(10): First image with six edge detection

Table (1): First image with six edge detection Threshold value (70).

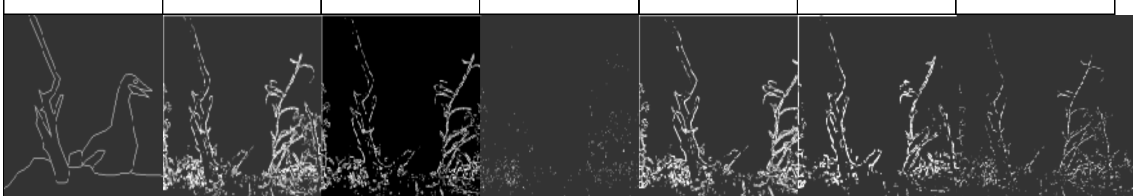
Edge Detection Filter	Threshold value	TP	TN	FP	FN	Accuracy%
Sobel	70	565	141987	7360	4489	92.32
Prewitt	70	411	142434	6913	4643	92.51
Discrete Laplacian	70	10	149128	219	5044	96.59
Frei - Chens	70	345	142123	7224	4709	92.27
Pyramid	70	459	140923	8424	4595	91.56
First order derivative of Gaussian	70	88	147505	1842	4966	95.59

Table (2): First image with six edge detection and Threshold value (90).

Edge Detection Filter	Threshold value	TP	TN	FP	FN	Accuracy %
Sobel	90	276	145669	3678	4778	94.52
Prewitt	90	192	146057	3290	4862	94.72
Discrete Laplacian	90	1	149308	39	5053	96.70
Frei - Chens	90	170	145812	3535	4884	94.54
Pyramid	90	163	145879	3468	4891	94.58
First order derivative of Gaussian	90	30	148781	566	5024	96.37

- **Second image:**

Ground Truth	Sobel	Prewitt	Discrete Laplacian	Frei - Chens	Pyramid	First order derivative of Gaussian



Figure(11): Second image with six edge detection

Table (3): Second image with six edge detection and Threshold value (80).

Edge Detection Filter	Threshold value	TP	TN	FP	FN	Accuracy %
Sobel	80	198	146853	5357	1993	95.23
Prewitt	80	149	147436	4774	2042	95.58
Discrete Laplacian	80	0	152062	148	2191	98.48
Frei - Chens	80	100	147079	5131	2091	95.32
Pyramid	80	64	149124	3086	2127	96.62
First order derivative of Gaussian	80	23	151048	1162	2168	97.84

Table (4): Second image with six edge detection and Threshold value (90).

Edge Detection Filter	Threshold value	TP	TN	FP	FN	Accuracy %
Sobel	90	175	143231	8890	2105	92.87
Prewitt	90	165	144319	7802	2115	93.57
Discrete Laplacian	90	7	151768	353	2273	98.29
Frei - Chens	90	151	143946	8175	2129	93.32
Pyramid	90	133	143963	8158	2147	93.32
First order derivative of Gaussian	90	44	150207	1914	2236	97.31

From the tables shown, it is obvious that discrete laplacian edge detection filter gives the highest matching with the three ground truth images, which proves that its work is the best in detecting the energy of an image. Also, it can be inferred from the tables that first order derivative of Gaussian comes in the second best edge detection filter among the six edge detection filter types.

7.Conclusion and Future Work

In this paper, seam carving operator was implemented in order to retarget an image while considering its content. The main advantage of seam carving resizing is to protect the feature in the image and only remove or insert the un important pixels in the image instead of shrink or enlarge uniformly over the whole image. Due to the effects of noise in retargeting process, pre-processing is necessary to remove noise so seam operator can choose path through parts of the image that had noise. Also, because of numerical stability, when cumulative minimum energy is computed from last row up to first row (compute vertical seam costs) gives the same optimal seam path when starting the computation from first row until reaching last row. To compute the horizontal seam costs, the cumulative minimum energy from left to right gives the same optimal seam path as if it calculated from right to left.

By matching the obtained energy (from the six edge detection filters) with a ground truth image, discrete laplacian gives the best edge responses.

The forward energy enhancement is implemented by taking into account the energy created from the new neighbors after the seam is removed. The resulted images show that forward energy that is used in computing seam costs gives better results than backward energy calculation.

As future work, Seam carving was applied automatically for all the tested images, but in order to make this paper give better results, face detection should be provided to prevent seams from passing through faces, and the proposed algorithm can be extend to work on videos. Also, find a perfect way to combine vertical and horizontal seams in multi-size images.

8.References

[1] S. Avidan and A. Shamir,"Seam Carving for Content-Aware Image Resizing," Journal ACM Transactions on Graphics (TOG) - Proceedings of ACM, Volume 26 Issue 3, July 2007.
[2] Viola P., and JONES M., “Rapid object detection using a boosted cascade of simple features”, In Coference on Computer Vision and Pattern Recognition (CVPR),2001.
[3] Itti L., Koch C., and Neibur E. ,” A model of saliency based visual attention for rapid scene analysis”, PAMI 20, 11, pp:1254–1259,1999.
[4] F. Shafieyan, N. Karimi, Ebrahim N.Esfahani, S. Samavi ,“Image Seam Carving Based on Content Importance and Depth Maps” Isfahan University of Technology, May- 2014.
[5] Chen J., Paris S., Durand F.,” Real-Time Edge-Aware Image Processing with the Bilateral Grid”, ACM Trans. Graph. 26, 3, Article 103 (July 2007), 9 pages. DOI = 10.1145/1239451.1239554 http://doi.acm.org/10.1145/1239451.1239554,2007.
[6] Sylvain P., Pierre K., Jack T. and Fr´edo D., “Bilateral Filtering: Theory and Applications”, Foundations and Trends in Computer Graphics and Vision Vol. 4, (2008).
[7] Gonzalez R. C. and Woods R. E.,”Digital Image Processing”, Pearson Education Asia Pie Ltd, 2000.
[8] Umbaugh S. E,”Computer Vision and Image Processing”, A Practical Approach using CVIP tools, Prentice Hall PTR, 1998.
[9] C. H. Chuang and Z. Y. Shi, ‘‘Target-preserving content-aware image resizing,’’ *IEEE Conf. Genetic and Evoluntionary Computing (ICGEC),* pp. 19-22, 2012.
[10] Z. He, M. Gao, H. Yu, X. Ye , L. Zhang, and G. Ma, “A new improved seam carving content aware image resizing method,” IEEE Conf. Industrial Electronics and Applications (ICIEA), pp. 738-741, 2013.

صورة نحت التماس على أساس تحجيم المحتوى عن طريق أسلوب التدرج

ا.م.د مثيل عماد الدين* رنا محمد حسن*

المستخلص

في هذا البحث يتم تنفيذ المرشح الثنائي وتقنية نحت التماس للحصول على صورة يتم اعادة توجيهها الى حجم جديد ذات مظهر واضح. يستخدم المرشح الثنائي لتحسين الصورة عن طريق تجانس النسيج والحفاظ على الحواف. ان تنفيذ نحت التماس يدرك تغير لمحتوى حجم الصورة ويعطي نتائج افضل في تغير حجم الصورة من تحجيم الطرق القياسية وتقنيات الاقتصاص ,استخدمت ستة انواع من فلاتر كشف الحواف للحفاظ على طاقة الاجسام في الصورة وجرى تقييم لهذه المرشحات للعثور على افضل طاقة بين هذه الانواع الستة.

الكلمات المفتاحية: المرشح الثنائي، تقنية نحت التماس ، كشف الحواف

*الجامعة التكنولوجية