Midterm Report
Content-Aware Seam Carving

HAN UL LEE (BRIAN)
CS6475 Fall 2020
hanbrianlee@gatech.edu

***Abstract*—In this report, results and analysis relating to attempts to replicate some of the results shown in "Seam carving for content-aware image resizing" by Avidian et al [1] and "Improved seam carving for video retargeting" by Rubinstein et al [3] will be shared. For specific details on which figures from the papers were replicated, refer to [4].**

I. Discussion of Algorithm

Here the overall implementation and discussion of some of the key algorithms not fully discussed in the authors' papers [1], [3] will be discussed. Although there are many different variations to energy functions and other approaches exist that solve the same problem, only the implementations that were done for this report will be shared; this mean only vertical seam removals will be discussed, but note that horizontal seam removals can simply be achieved by rotating the image by 90 degrees and using the same algorithm.

For seam removals, first the energy is computed as L1 magnitude of gradient (magnitude of x gradient and y gradient separately summed) of the image. Both full-image differential (computed by absolute of difference of image and itself shifted by 1 horizontally and vertically) and sobel [2] approximated gradients were tried, and I chose to go with sobel filter with kernel size 3 as it created seams closer to the authors' result [1], [3], specifically just enough seams crossing the dolphin. After this, the minimum cumulative energy M is computed following Algorithm 1 where with 8-point connection there are only 3 points (diagonal left, bottom, diagonal right) pixels to be considered at a time. With this M computed, finding a seam is simple as it is a process of starting on the bottom row of M where the energy is the minimum (call this index j_n), and finding all indices j going one row above each time where within the 3 pixels above (diagonal left above, above, diagonal right above) the M[row] is minimum. Collecting all the j indices and flipping the order gives one vertical seam $\{j_0,...,j_n\}$, and this process is enclosed in a function called "find_seam(M)". Collecting multiple sequentially removed seams is undertaken by repeating this function and caching the seams while removing a seam as it is found, until the width of the reduced image reaches the desired width. During this process, a retroactive seam indices compensation for removed seams happen which is described in more detail in the video presentation (link in Section V) in the interest of conserving space on the report. Seams found by this energy function and inserted are called backward energy approach as it only cares about the energy computed right before finding the seam.

For seam insertions, first the seams are found via going through the seam removal process for the number of pixels that are to be inserted. Then for each seam (in the order it was removed), each row gets expanded by 1 pixel and all the pixels on the right of the seam's index j gets pushed to the right by 1 pixel. Then the row[j] gets updated by the average of its left and right neighbors and duplicated, where I assumed (as it was ambiguous, discussed further in Section IV) this meant left and right pixels that will be there once pixel at j is inserted. I interpreted this as computing the average with expanded_row[j-1], expanded_row[j], and expanded_row[j+1], where the first two terms are actually equal. This made sense as much details as possible would ideally be preserved while merging with the right pixel for smoothness. This computed average filled the first two terms. Here also seam indices adjustment get made while inserting each seam, which will be further explained in the video presentation.

For forward energy based seam insertions, the forward energy is computed in a different way where the gradients that would result after a seam is inserted are computed (further details in the video presentation) while accumulating the minimum energy. As this already creates the accumulated energy, the "find_seam(M)" method is invoked thereafter to find the seam for removals, and the rest of the process is identical to backward energy approach.

Algorithm 1: Minimum Cumulative Energy Function

```
function cum_energy (e);
n,m = height, width of e; M = (n,m) filled with zeros
  for i in range(n) do
    for j in range(m) do
      left = max(0, j-1), right = min(m-1, j+1)
      M[i,j] = e[i,j] + min(M[i-1, left], M[i-1,j],
        M[i-1, right])
    end
end
return M
```

II. Comparison Metrics

For comparison metrics, I chose mainly two kinds of metrics. One that can compare pixel values as well as relative location and scale based on spatial pixels, and another kind that is less sensitive to location and scale based on histograms. For

the former, I used normalized Mean Squared Error (MSE), and for the latter I used histogram intersection and Chi-Squared distance.

MSE is useful in the way that it is sensitive to translation and scale. To illustrate this, consider for example the seam removal process removed only the right side of the scene in the dolphin image (see Figure 2), the overall histogram statistics of pixel values may be similar to the one where both sides of the dolphin are equally removed, due to the overall similarity between the left and right side of the dolphin (largely blue-ocean and sky). However, clearly the dolphin with only right side removed will be sitting closer to the right side of the reduced image frame, whilst the one where both sides are equally removed, the dolphin would be in the middle. This shift in dolphin position will be easily discovered by pixel-wise mean squared error, as there will be mostly skies in the balanced-removed image (if this were the target) where there will be a dolphin in the right-side-removed image (if this were the replication attempt). Similar argument holds for scale changes where lots of seam insertion enlarges certain parts of objects, compared to the target image where perhaps that object did not get enlarged. This metric is simply calculated as the mean of squared difference between pixel values across 3 channels, divided by the image area and additionally dividing by 255 in order to obtain a normalized metric ranging from 0 (perfect match) to 1 (maximum difference of 255, where one image is all white and the other image is all black for example). So the lower would be better. This metric is called MSE-normalized from here on.

Histogram based distance methods (1 0 255 bin histogram per channel for the 3 channels Blue, Green, Red; the 3 channels distances get averaged) are beneficial in a way that it cares mostly about the match between the statistics of colors globally, and thus even if the dolphin in the replicated image was slightly shifted to a direction by a few pixels, as long as the overall distribution of colors are roughly the same, the computed distances will not penalize them as much as they would in MSE-like spatial-sensitive distances. The intersection approach is commonly used in classification tasks (reference) where it tries to measure the area under two histograms (original and replication in this paper's case, or ground-truth and to-be-classified image in classification task) where both histograms intersect (minimum of the two for each bin). The resulting intersecting count for bins get summed up and get divided by the original's histogram for normalization (produces value from 0 1, where 1 means perfect match, and 0 means no match). While the intersection metric focuses on the "match", the Chi-Squared distance focuses on the "mis-match" part. By taking square of the differences between the two histograms and dividing by their sum and image area (height x width), it can measure on average per pixel location, what the relative difference is taking into account the magnitude of each bin (division by their sum). This metric isn't bounded by 1 as in the intersection metric, although 0 does mean that the two histograms, and therefore images have completely the same color distribution. Higher numbers indicate large mis-match. The two histogram based methods here can be thought of as inversely related and are referred to as hist-intersection-metric and hist-chi-squared-metric, respectively.

III. Replicated Images and Comparison of Results

Suggested length for this entire section is 1 to 2 pages.

Here, the replicated images are provided along with additional visualizations that show the pixel-wise average magnitude (over BGR channels) showing from 0 (no difference, black) to 255 (maximum difference, white). For the original images from the author, refer to [1] and [3]. The histogram plot is only shown for the very first replication attempt of waterfall in Figure 1, because it is hard to discern the difference between the histograms and is already redundantly visualized by the black-white pixel-wise average magnitude. Whenever "authors" are mentioned, they are referred to the authors from the two papers [1] and [3].

As for deeper discussion on why the differences may have occurred, it will be discussed later in Section IV. Therefore, this section will discuss more on qualitative side (visual) and quantitative side (metrics).

A. Fig. 5 Waterfall (2007) Backward Energy Seam Removal

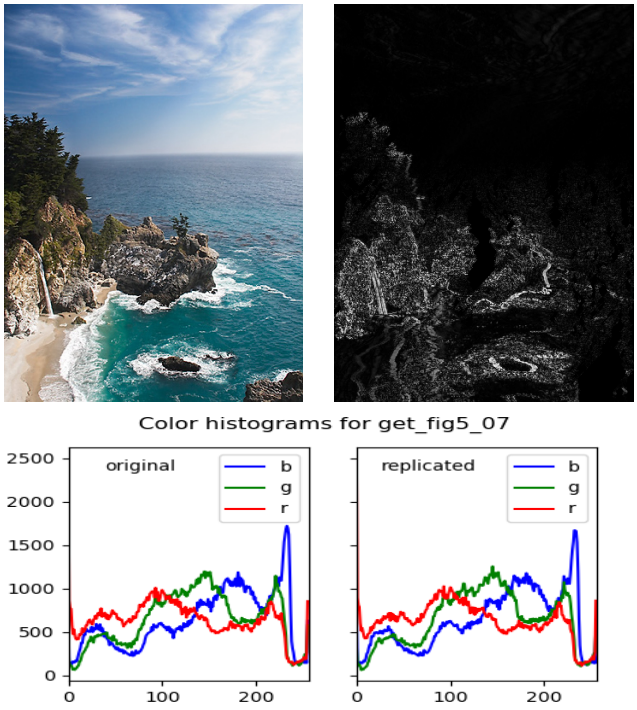

Fig. 1. Left-Top: 50% width reduced waterfall image. Right-Top: Magnitude difference visualization. Bottom: BGR 3 channel histogram comparison between original and replication attempt.

Figure 1 shows the replication attempt and comparison visualizations. It visually looks very similar to the results that the authors had [1], with some difference especially on the left-bottom part (waterfall area) where my result shows waterfall and the surrounding mountain and nearby beachfront a bit

shifted compared to the author's result, as visualized by the Top-Right magnitude difference visualization. The sky and the sea portions had almost the same result indicated by almost completely black areas in the visualization.

Metrics: 'MSE-normalized': 0.067, 'hist-intersection-metric': 0.981, 'hist-chi-squared-metric': 1.612. With low MSE-normalized hist-chi-squared and high intersection-metric, the replication seemed to have succeeded in closely replicating the authors result.

B. Fig. 8 Dolphin (2007) Backward Energy Seam Insertion

Fig. 2. Left: Enlarged dolphin image (approx 50% increased in width) with inserted seams shown in red. Middle: Without red seams. Right: Pixel-wise difference compared to the authors' result.

Figure 2 shows the approximately 50% enlarged dolphin image with and without seams, as well as the pixel-wise magnitude difference compared to the authors' result. As can be seen from the difference visualization, mostly the replication succeeded, except for some pixel locations on the ocean where the difference magnitude doesn't seem to be too high (not completely white, more grayish indicating values ¡¡ 255) and some of the contour lines of the dolphin, where my replication attempt did not remove the exactly same seams as did in the authors' result. The sky and clouds seem to have been more ideally seam-removed and replicated.

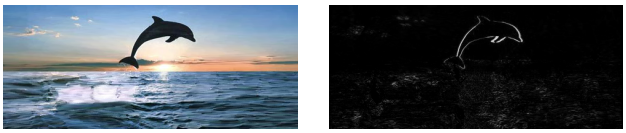

Fig. 3. Left: 2-step consecutive 50% enlargement resulting in 75% enlargement. Right: Pixel-wise difference.

Figure 3 shows approximately 75% enlarged image via two consecutive 50% enlargements. There doesn't seem to be any obvious visual artifacts, and in the difference visualization, it is shown that similar to 2, the ocean has slight dissimilarities here and there, but the dolphin's contour seems to have thicker pixels of differences around the contour. This is most likely attributed to the fact that, since the one-step 50% enlargement had different seams for dolphin removed, doing the same step twice would create more differences on the second pass, compounding the dissimilarities. As will be discussed in Section IV, this is most likely due to the slightly different energy computation or edge case handling.

Metrics: "enlarged 50% in 1-step": 'MSE': 624.59, 'MSE-normalized': 0.022, 'hist-intersection-metric': 0.977, 'hist-chi-squared-metric': 0.817, "enlarged 75% in 2-steps": 'MSE': 711.028, 'MSE-normalized': 0.025, 'hist-intersection-metric': 0.974, 'hist-chi-squared-metric': 1.543. The low MSE-normalized value and close to 1 hist-intersection-metric reveal that the replication is good.

C. Fig. 8 Bench (2008) Seam Removal by Two Methods

Fig. 4. Top: Width-reduced (approx 50%) bench image with backward energy seam removal on the left and difference with author's result. Bottom: Width-reduced (approx 50%) bench image with forward energy seam removal on the left and difference with author's result.

Figure 4 shows the backward energy results on the top and the forward energy results on the bottom. As the difference magnitude visualization shows, both results seem to have replicated the authors' results except for differences near the sides of the bench, along the contours of some of the back-support-bars, and a little on the grass area under the bench. The differences on the sides can stem from the seams finding either not enough or more seams dissecting the bench, resulting in longer/shorter benches in terms of width. As for the grass, if the many seams that are removed are not equal in the order and locations, then they can compound and manifest as very noisy, yet small differences (as grass colors are not too different from pixels around it even if differently seam-removed). In backward energy, most of the seams do not dissect the bench and therefore it is expected that the difference is small in the middle area of the bench, but what's impressive is that even though many seams dissect the bench for the forward-energy method at slightly different locations compared to the authors' result, the difference still remains small; this means that the forward energy really did work in that it was able to find the seams which introduces least new energy (created by new edges or abrupt changes in pixels by removal of a seam). Visually, the forward energy result looks more natural (backward energy bench looks unnaturally wide and also the thin vertical beams are too narrow and broken in places) and much better blends in the contrast difference on the extreme

sides of the lake and the middle part of the lake. The results really accentuate the effectiveness of forward energy compared to the backward energy.

Metrics: "forward energy": 'MSE': 1183.54, 'MSE-normalized': 0.038, 'hist-intersection-metric': 0.947, 'hist-chi-squared-metric': 15.629, "backward energy": 'MSE': 1049.788, 'MSE-normalized': 0.037, 'hist-intersection-metric': 0.957, 'hist-chi-squared-metric': 3.172. The low MSE-normalized value and close to 1 hist-intersection-metric indicate that the replication is good.

D. Fig. 9 Car (2008) Seam Insertion by Two Methods

Fig. 5. Top: Width-enlarged (approx 50%) car scene image with backward energy on the left and difference with author's result on the right. Bottom: Width-enlarged (approx 50%) car scene image with forward energy on the left and difference with author's result on the right.

Figure 5 shows the car scene enlarged approximately 50% with backward and forward energy. As shown by the difference magnitude visualization, the backward energy version replicated better on the regions around the tire and worse on the front bumper region compared to the forward energy. For other background, the replication result seems not too different between backward and forward energy. Qualitatively assessing, the forward energy version, just as shown by the authors' result, certainly amplifies the effectiveness of the forward energy. The car itself may pass as "good" with both backward and forward energy, if one doesn't know what the original car looked like (the boxy shape of windshield area is better captured by forward energy). What is very noticeable however is the hotel entrance and the guardrails in the second floor. The entrance is very distorted and no one would think that the entrance is designed and built in such shape. Likewise, the vertical orange-ish pink vertical section on the guardrails is curved and while artistic buildings may have such a shape, it is unlikely for a hotel to have such shape. Therefore, forward energy in this case seems to certainly produce more realistic enlargement of the original image. Looking into the replication a little further, one can notice how the white parking line vertical to the sidewalk in front of the hotel has lots of salt and pepper like discolorations. These types of features in the original image make it hard for any two sets of seam insertion results to look completely alike; if the seams are not exactly the same and ordered then the difference will easily manifest.

Metrics: "forward energy": 'MSE': 335.698, 'MSE-normalized': 0.015, 'hist-intersection-metric': 0.986, 'hist-chi-squared-metric': 0.718, "backward energy": 'MSE': 435.789, 'MSE-normalized': 0.017, 'hist-intersection-metric': 0.982, 'hist-chi-squared-metric': 0.855. The low MSE-normalized value and close to 1 hist-intersection-metric show that the replication is good.

IV. Ambiguities and Issues

There were a number of parts in the authors papers that could be interpreted one way or another which caused issues.

In the 2007 paper [1], "duplicate the pixels of s by averaging them with their left and right neighbors" is mentioned in section 4.3. Imagine a row composed of [a b c d e] where for this row the seam index is 2 pointing to c, so after insertion it becomes [a b c x d e] where x is the newly inserted pixel. It was unclear whether c x = (b + c + d) / 3 or only x is updated, or whether c x = (c + d)/2. I followed the last approach as the author used the word duplicated, and averaging involving the value b for example seemed to create less smoother result.

Also, how e1 (equation (1) in [1]) was exactly computed wasn't mentioned. There are multiple ways to compute the gradients and details on for example border padding was not mentioned. Changing border padding type from reflection to replication and so on changed the seams immediately and so did the kernel sizes for sobel filter, if sobel were used. These ambiguities presented challenges in replicating exactly the same results authors got.

Another ambiguity was edge case handling for forward energy computation. "After removing the seam, new neighbors (in gray) and new pixel edges (in red) are created." is mentioned in 2008 paper [3] figure 7 caption. However, on the very left and right edge or the very top row, there is no explanation on what should happen there. I assumed the top row gets left as 0's and the extreme ends compute the accumulated minimum energy only based on the edges created with the pixels above, which generated satisfactory results.

V. Video Presentation

Here you can see my presentation where I share more explanations to regarding detail implementations and visualizations.: https://youtu.be/ZnhMaA0nBBw

References

[1] Shai Avidan and Ariel Shamir. Seam carving for content-aware image resizing. *ACM Trans. Graph.*, 26(3):10, 2007.

[2] Nick Kanopoulos, Nagesh Vasanthavada, and Robert L Baker. Design of an image edge detection filter using the sobel operator. *IEEE Journal of solid-state circuits*, 23(2):358–367, 1988.

[3] Michael Rubinstein, Ariel Shamir, and Shai Avidan. Improved seam carving for video retargeting. *ACM Transactions on Graphics (SIGGRAPH)*, 27(3):1–9, 2008.

[4] Kimberly Piyawan Sirichoke. Midterm project. https://github.gatech.edu/omscs6475/assignments/tree/master/MT-Research_Project, 2020.