

Report: Seam Carving as HMM

CS5340 Uncertainty Modeling in AI

Xudong Shen (A0210662R) & Ruixi Lin (A0210662R)

September 25, 2019

Background

Effective image resizing should consider not only geometry but also image content, i.e., content-aware image resizing. Seam carving [1] is a simple method, which builds on Hidden Markov Model (HMM), to achieve this. It resizes an image by successively deleting or duplicating a vertical/horizontal seam with lowest energy. In implementation, the optimal seam can be tracked using Viterbi algorithm.

In this project we only consider the Viterbi algorithm part: given the energy map of a image, find the vertical seam with minimal energy.

Problem formulation

We construct the Hidden Markov Model as Figure 1. Observed variable $x_n, n = 1, \dots, H$ represents n^{th} row of the image. Latent variable $z_n, n = 1, \dots, H$ represent the seam's pixel's index in each row. Each latent variable z_n takes one discrete state from $\{1, 2, \dots, W\}$. H, W are the height and width of the image, respectively.

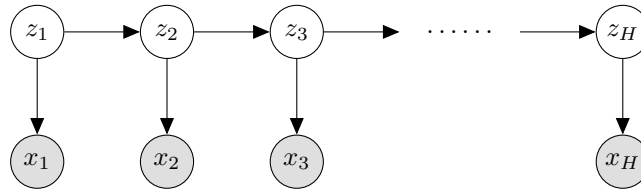


Figure 1: Graphical model representation of HMM

The joint probability can be expressed as:

$$p(X|Z) = \prod_{n=1}^H p(x_n|z_n)p(z_n|z_{n-1}) \quad (1)$$

The states of the latent variables (equivalent to the seam's pixels' indexes) can be found by maximum a posteriori estimation.

$$\operatorname{argmax}_{z_i, i=1, \dots, H} \prod_{n=1}^H p(x_n | z_n) p(z_n | z_{n-1}) \quad (2)$$

The state with lower energy corresponds to higher probability of being occupied. We reconstruct eq.(2) in terms of energy. We work on the log probability.

$$\operatorname{argmin}_{z_i, i=1, \dots, H} \sum_{i=1}^H [\ln(e(i, z_i)) + \ln(p(z_{i-1}, z_i))] \quad (3)$$

where $e(i, z_i)$ is the energy term of each pixel at (i, z_i) , and $p(z_{i-1}, z_i)$ is the transition probability that ensures continuity.

Viterbi algorithm

Optimal seam can be found using Viterbi algorithm. Similar to [1], first we traverse the image from the first row to the last row to calculate the cumulative log probability, according to eq. (4). Upon arriving at the last row, the minimum cumulative log probability gives z_H , the index of the last pixel of the seam.

$$M(i, z_i) = \ln(e(i, z_i)) + \min(M(i-1, z_i-1), M(i-1, z_i), M(i-1, z_i+1)) \quad (4)$$

Then, we backtrack to find the seam with the minimum energy.

Pseudo code

See algorithm 1.

Experiment

The input image and its original energy map (note the energy map changes after each seam operation) is shown in Figure 2, both of size 164*111 pixels. In the energy map, darker color indicates higher energy.

In order to better understand the optimal seam, we set the number of seams that need to be carved out as 100 and plot the seam energy. The seam energy is the sum of each pixel's energy. See Figure 3.

It is worth noting that, although each seam is of the lowest energy configuration in each image (image changes after preceding operations), it is not guaranteed that preceding seams will be of lower energy than latter ones.

We also set the number of seams to be carved out as 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, respectively. See Figure 4 for the outputs. It gives an informative visualization of how Seam Carving method carve out the seams successively.

Finally, in Figure 5 we set the number of seams to be carved out as 30 and label them in the input image.

Algorithm 1 Viterbi algorithm

Arguments: Energy map \mathbf{E} , Height H , Width W **Outputs:** seam indexes \mathbf{s}

```
1: function VITERBI( $\mathbf{E}$ )
2:   procedure FORWARDPASS
3:     for  $r$  in  $\text{range}(H)$  do ▷ enumerate rows
4:       if  $r = 0$  then ▷ initialize first row
5:          $\mathbf{m}(\cdot) \leftarrow \ln(\mathbf{E}(r, \cdot))$ 
6:          $\mathbf{m}_{old} \leftarrow \mathbf{m}$ 
7:       else
8:         for  $k$  in  $\text{range}(W)$  do ▷ enumerate all possible states
9:            $\phi(r, k) \leftarrow \underset{i=k-1, k, k+1}{\text{argmin}} \mathbf{m}_{old}(i)$ 
10:           $\mathbf{m}(k) \leftarrow \ln(\mathbf{E}(r, z_i)) + \mathbf{m}_{old}(\phi(r, k))$  ▷ accumulate
11:        end for
12:         $\mathbf{m}_{old} \leftarrow \mathbf{m}$  ▷ update to next row
13:      end if
14:    end for
15:  end procedure
16:  procedure BACKTRACK
17:    for  $r$  in  $\text{range}(H - 1, -1, -1)$  do ▷ enumerate rows reversely
18:      if  $r = H - 1$  then ▷ find the seam end to start with
19:         $\mathbf{s}(r) \leftarrow \underset{k}{\text{argmin}} \mathbf{m}(k)$ 
20:      else ▷ backtrack seam index
21:         $\mathbf{s}(r) \leftarrow \phi(r + 1, \mathbf{s}(r + 1))$ 
22:      end if
23:    end for
24:  end procedure
25:  return  $\mathbf{s}$ 
26: end function
```



Figure 2: Input image and its energy map

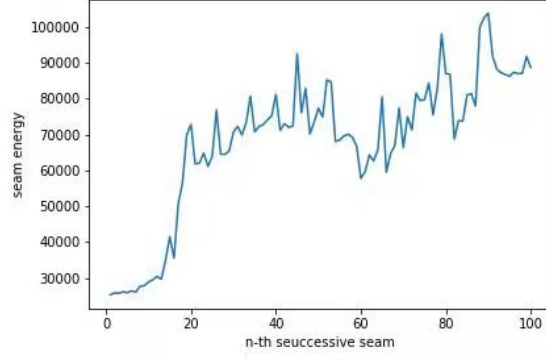


Figure 3: n-th Seam energy



Figure 4: upperleft is the input image. The rest are outputs after deleting different numbers of vertical seams. Images from left to right and from top down corresponds to seam number 10 to 100, respectively.

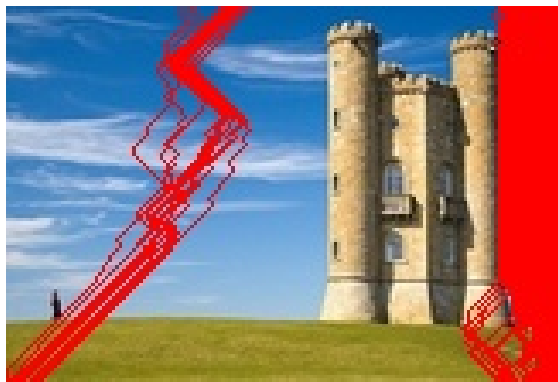


Figure 5: seams labeled in the input image

References

- [1] Shai Avidan and Ariel Shamir. Seam carving for content-aware image resizing. *ACM SIGGRAPH 2007 papers on - SIGGRAPH '07*, 2007.