# Seam Carving for Content-Aware Image Resizing

Shai Avidan, Ariel Shamir (2007)

Team 7 游凱雯、李婷穎、翁如萱

# Table of Contents

# 01
# Abstract

# Abstract

- Seam Carving is a content-aware image resizing method.
- Adjusting unnoticeable pixels to avoid distorting the content of an image.

Original image

Scaling

Seam carving

# Abstract

- The importance of each pixel is determined by its energy value.
- There are several ways to define an energy function, e.g. entropy, L1-norm of gradient.

- Seam is a 8-connected path from top to bottom, or left to right
- Enlarging an image by inserting seams.
- Shrinking an image by removing seams.

- Aside from image resizing, seam carving can also be used for content amplification and object removal.

# 02
# Motivation

# Motivation

- We found this paper when searching for related articles of image quilting (homework 3).
- Highly practical. Most people resize an image by cropping or scaling; however, it may sacrifice some important contents or lead to serious distortion problem.
- Many applications. It helps us complete some other interesting tasks, like object removal and content enhancement.

# 03

# Problem Definition

# Problem Definition

- Do “content-aware” image resizing for both expansion and reduction.
- Apply seam carving on tasks of content amplification and object removal.

# 04
# Algorithm

# List of Tasks

- Determine the importance of each pixel.
- Find out a seam to be carved out or inserted.

- Image reduction from one dimension.
- Image enlargement from one dimension.
- Image retargeting from two dimensions.

- Object removal.
- Content amplification.

# Energy Function

- Energy function is used to determine the importance of each pixel.
- A pixel with high energy value is regarded as an important pixel.
- We have tried $L_1$-norm of gradient and entropy.

$L_1$-norm of gradient

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

entropy

$$E\big(I(X)\big) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i)$$

# Find out a Seam - Background

- Example: reduce the width of an image.
- To preserve energy of an image, we would like to remove pixels with low energy values.

- [Remove_pix] Remove low energy pixels from each row? zig-zag effect
- [Remove_col] Remove columns with the lowest energy value ? not continuous
- [Crop] Cropping (a sub-window that contains the highest energy)? sacrifice the content

# Find out a Seam - Background

Original image

Remove_pix

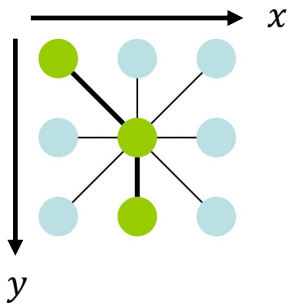
Remove_col

Crop

Seam carving

# Find out a Seam - Definition

- To preserve both energy and visual coherence, we need “seam”.
- Seam is a 8-connected path.




Vertical seam: $s^{\mathrm{X}} = \{(x(i), i)\}_{i=1}^{n}$, s.t. $\forall i, |x(i) - x(i-1)| \leq 1$

Horizontal seam: $s^{\mathrm{Y}} = \{j, y(j))\}_{i=1}^{m}$, s.t. $\forall j, |y(j) - y(j-1)| \leq 1$

- Minimize the cost of a seam by dynammic programming.

Vertical seam: $M(i,j) = e(i,j) + \min(M(i-1,j-1), M(i-1,j), M(i-1,j+1))$

Horizontal seam: $M(i,j) = e(i,j) + \min(M(i-1,j-1), M(i,j-1), M(i+1,j-1))$

# Image Reduction from One Dimension

- Image reduction from one dimension.

  Reduce the width: $n \times m \rightarrow n \times m', m > m'$

  Reduce the height: $n \times m \rightarrow n' \times m, n > n'$

- Repeat the following steps until achieving our target.

  Step 1: Obtain energy values of all pixels.

  Step 2: Find a seam with the lowest cost.

  (width: vertical, height: horizontal)

  Step 3: Remove the seam from an image.

# Image Insertion from One Dimension

- Image insertion from one dimension.

  Expand the width: $n \times m \rightarrow n \times m', m < m'$

  Expand the height: $n \times m \rightarrow n' \times m, n < n'$

- There are three methods to insert seams.

  Method 1: Insert one by one.

  Method 2: Insert all seams in one time.

  Method 3: Insert seams in batches.

# Image Insertion from One Dimension

- [Insert One by one] Repeat the following steps until achieving our target.

  Step 1: Find a seam to be removed from the image.

  (width: vertical, height: horizontal)

  Step 2: Insert the seam back to the image.

  Step 3: Duplicate the seam by averaging it with its left and right neighbors.

- Problem: stretching artifact by choosing the same seam.

# Image Insertion from One Dimension

- ‘’Insert one by one’’ may create a stretching artifact by choosing the same seam.

# Image Insertion from One Dimension

- [Insert all seams in one time]

  Step 1: Find all the seams to be removed.

  (Goal: Insert $n$ seams, #seams to be removed: $n$)

  Step 2: Insert seams back to the image in backward, and duplicate it afterwards.

- Problem: Duplicates all the seams in an image is equivalent to standard scaling.

  50% enlargement

# Image Insertion from One Dimension

- “Insert all seams in one time” may choose all the seams in an image, which is equivalent to standard scaling.

# Image Insertion from One Dimension

- [Insert seams in batches]

  Break the process of “Insert all seams in one time” into several steps.

# Image Retargeting from Two Dimensions

- Image retargeting from two dimensions.

  $n \times m \rightarrow n' \times m'$

- We propose the new method.

  Paper: Horizontal seams + vertical seams → order is the matter. [Not yet]

  Ours: Proportional scaling + seam removal.

# Image Retargeting from Two Dimensions - Author

- Optimal order.

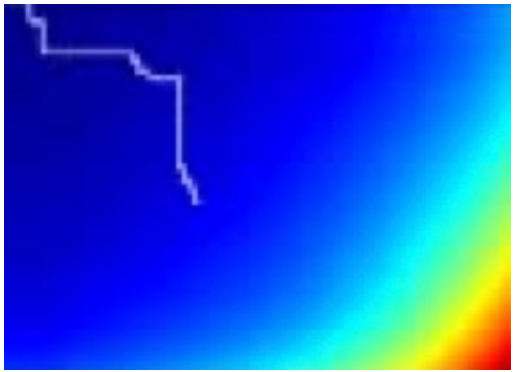
  Vertical first? Horizontal first? Alternate between the two?

- Image reduction from two dimension.

  Limitation: $n' < n, m' < m$

$$\min_{s^x, s^y, \alpha} \sum_{i=1}^{k} E(\alpha_i s_i^x + (1-\alpha_i) s_i^y)$$

$$T(r,c) = \min \begin{cases} T(r-1,c) + E\left(s^x\left(I_{(n-r-1)\times(m-c)}\right)\right) \\ T(r,c-1) + E\left(s^y\left(I_{(n-r)\times(m-c-1)}\right)\right) \end{cases}$$



# Image Retargeting from Two Dimensions - Ours

- Image retargeting from two dimensions.

  Step 1: Set the original size as $(h, w)$ and the target size as $(h', w')$

  Step 2: Calculate the ratio $r_h = \frac{h'}{h}$ and $r_w = \frac{w'}{w}$.

  Step 3: Scale the image by the factor of $r_h$ if $r_h > r_w$, and vice versa.

  Step 4: Do seam removal to achieve the target size.

- Advantages.

  Easier ☺.

  No limitation: Apart from image reduction, we can also do enlargement.

# Content Amplification

- Content amplification.

  Step 1: Do proportional scaling to enlarge an image.

  Step 2: Apply seam removal on the image to carve the image back to its original size.

# Object Removal

- Object removal

  Step 1: Mark the target object to be removed.

  Step 2: Give the marked pixels low enough energy values, so that they tend to be removed.

  This step should be done in every cycle of seam removal.

  Step 3: Do seam removal until all marked pixels are gone.

  The orientation is set by user.

  Step 4: Do seam insertion until the image back to its original size.

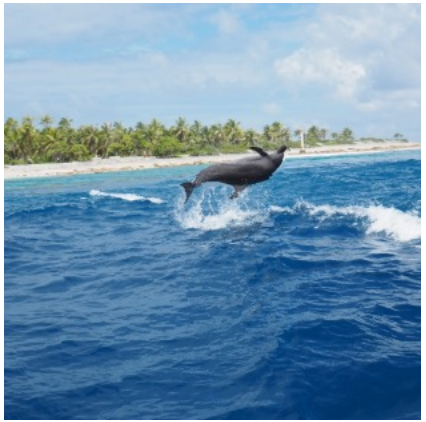
# 05
# Results

# Image Reduction from One Dimension

# Image Reduction from One Dimension

- The dolphin looks good under both energy functions. However, they show different effect!

Original image



$L_1$-norm of gradient

Entropy

# Image Reduction from One Dimension

- Noticeable artifacts: the unnatural variation of light and shadow.

Original image

$L_1$-norm of gradient

Entropy

# Image Reduction from One Dimension

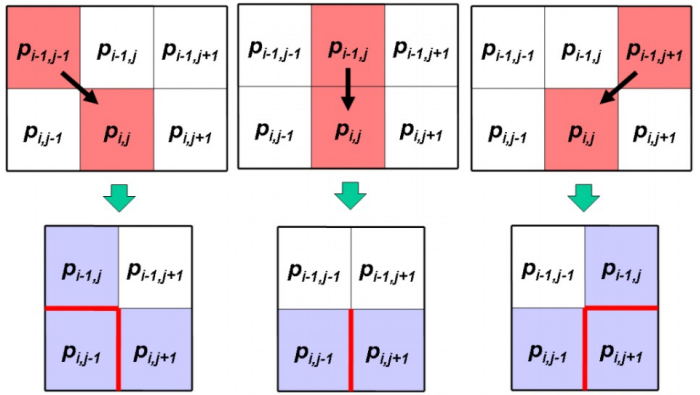
- Forward enregy.

$$M(i,j) = \min \begin{cases} M(i-1,j-1) + C_L(i,j) \\ M(i-1,j) + C_U(i,j) \\ M(i-1,j+1) + C_R(i,j) \end{cases}$$

$$C_L = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j-1)|$$

$$C_U = |I(i,j+1) - I(i,j-1)|$$

$$C_R = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j+1)|$$



# Image Reduction from One Dimension

- “Forward energy” works better!

$L_1$-norm of gradient

Entropy

Forward energy

# Image Reduction from One Dimension

- The mouse is diminishing.

Original Image

$L_1$-norm of gradient

Forward energy

# Image Reduction from One Dimension

- Using a “protective mask” to protect important content of the image.

  Give the red region “very high” energy value.

Protective mask

With mask

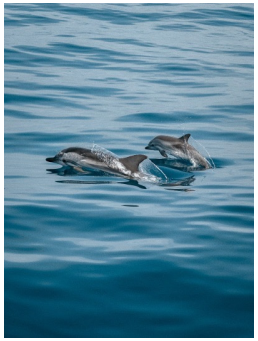
Without mask

# Image Insertion from One Dimension

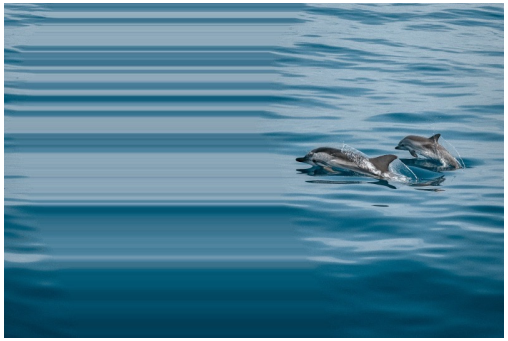
- Insert one by one vs. Insert all seams at one time vs. Insert seams in 5 batches.

stretching artifact

standard scaling



Original image



Insert one by one

Insert all seams at one time

Insert seams in 5 batches

# Image Retargeting from Two Dimensions

- Our method supports us to do enlarge or shrink an image from two dimensions!

Original image (360, 640)

Reduction (250, 550)

Expansion (400, 800)

# Content Amplification

- They are different in the cloudy sky.

Original Image

$L_1$-norm of gradient

Forward energy

# Object Removal

- Define a “mask”.

  Give the blue region “very low” energy value.

Original image

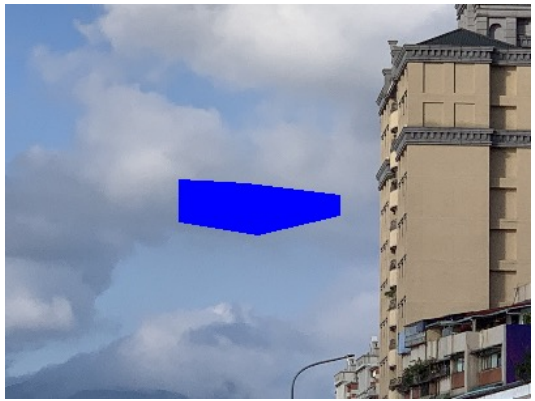


Image with a mask

# Object Removal

- Orientation does great impact on the result.
- There are some noticeable artifacts in the image.

  Improvement: insert the seams around masks. [Not yet]

Remove horizontally

Remove vertically

# 06
# References

# References


- Avidan, Shai, and Ariel Shamir. "Seam carving for content-aware image resizing." *ACM SIGGRAPH 2007 papers*. 2007. 10-es.
- Rubinstein, Michael, Ariel Shamir, and Shai Avidan. "Improved seam carving for video retargeting." *ACM transactions on graphics (TOG)* 27.3 (2008): 1-9.

# Thank You