Image and Video Retargeting Based on Seam Carving

-- aMMAI Final Project

Wei-Lun Chao (趙偉崙)
Hsiao-Hang Su (蘇小杭)
Advisor: Prof. Wiston Hsu
Date: 2010/06/28
Outline

- Introduction to Retargeting
- Image and Video Retargeting
  - Seam Carving
  - Forward Warping
- Our Modification on Video Retargeting
  - Seam Prediction Based on Motion Estimation
    - Forward, Backward
- Comparison of All Methods
- Conclusion and Future Work
- Reference
Outline

- Introduction to Retargeting
- Image and Video Retargeting
  - Seam Carving
  - Forward Warping
- Our Modification on Video Retargeting
  - Seam Prediction Based on Motion Estimation
    - Forward, Backward
- Comparison of All Methods
- Conclusion and Future Work
- Reference
Introduction to Retargeting

- There are lots kinds of photo and video display units around our daily life
  - ex. notebooks, PDAs, cell phones, etc

- Different kinds of display units may have different aspect ratios and sizes

- If we want to display it, we need to change its aspect ratio and size!
Previous and Related Works (1/2)

- Scaling, Cropping, Black Box (content-independent)
Warping (content-dependent)

Seam Carving (content-dependent)
Outline

- Introduction to Retargeting
- Image and Video Retargeting
  - Seam Carving
  - Forward Warping
- Our Modification on Video Retargeting
  - Seam Prediction Based on Motion Estimation
    - Forward, Backward
- Comparison of All Methods
- Conclusion and Future Work
- Reference
Introduction to Seam Carving

- A seam is defined as a path in image
  - Defining energy function based on importance of pixels
  - Seam cost: \( E(s) = E(I_s) = \sum_{i=1}^{n} e(I(s_i)) \)

- An optimal seam
  - \( s^* = \min_s E(s) = \min_s \sum_{i=1}^{n} e(I(s_i)) \)
  - Minimizing the seam cost
  - Can be found using dynamic programming
    \[
    M(i, j) = e(i, j) + \\
    \min(M(i-1, j-1), M(i-1, j), M(i-1, j+1))
    \]

Seam Carving for Image Resizing

- Using seam carving for content-aware image resizing of reduction and expansion
- Important parts of image have high energy
- Resizing the image by inserting or removing seams continuously

Original image

Content-aware resizing

Standard resizing

Seam Carving – Video Retargeting (1/2)

- If resizing each frame independently …
  - lack of temporal coherency
  - a global approach is required
- There are 2 methods that protect salient spatial and temporal content proposed in this paper
- Method 1: Static Seams Method
  - taking the maximum energy value at each pixel location
    \[
    E_{\text{spatial}}(i, j) = \max_{t=1}^{N} \{|\frac{\partial}{\partial x} I_t(i, j)| + |\frac{\partial}{\partial y} I_t(i, j)|\}
    \]
    \[
    E_{\text{temporal}}(i, j) = \max_{t=1}^{N} \{|\frac{\partial}{\partial t} I_t(i, j)|\}
    \]
    \[
    E_{\text{global}}(i, j) = \alpha \cdot E_{\text{spatial}} + (1 - \alpha)E_{\text{temporal}}
    \]

Then retargeting each frame independently!

Method 2: Graph Cut Method

- combining the frames of the video to form a 3D cube
- and finds 2D monotonic and connected manifold seams using graph cut

Disadvantage:
- Time consuming and has no motion information

“Warping” is a non-linear and non-homogeneous scaling technique!!

Regions with high importance will remain nearly unchanged, while regions with low importance will shrink!!

Forward warping is composed of three steps: Importance counting, Coordinate estimation, and Forward mapping!!

Forward Warping (2/4)

- **Importance counting**: Saliency map + Face detection
- **Coordinate estimation**: Least square error solution

Forward Warping (3/4)

- Shrink less at important regions:
  \[ x_{i,j} - x_{i-1,j} = S_{i,j} \]
  \[ y_{i,j} - y_{i,j-1} = S_{i,j} \]

- Spatial constraints:
  \[
  \begin{bmatrix}
  1 & -1 & 0 & 0 & \ldots & \\
  1 & \ldots & -1 & \ldots & \\
  \end{bmatrix}
  \begin{bmatrix}
  x_{i,j} \\
  y_{i,j} \\
  \end{bmatrix}
  = 
  \begin{bmatrix}
  \end{bmatrix}
  \]

Forward Warping (4/4)

Forward mapping:
For each pixel \((x, y)\) in the retargeted frame, find four pixels \((i, j)\) which have the nearest \((x_{i,j}, y_{i,j})\) and use there pixel colors to interpolate \(I_{\text{ret}}(x, y)\)

Results:

For videos: Adding temporal constraints

\[
x_{i,j,t} - x_{i,j,t-1} = 0
\]
\[
y_{i,j,t} - y_{i,j,t-1} = 0
\]
Issues of Energy Function (1/2)

- **Backward Energy in Dynamic Programming**
  \[ M(i, j) = e(i, j) + \text{energy function (e.g. gradient)} \]
  \[ \min(M(i - 1, j - 1), M(i - 1, j), M(i - 1, j + 1)) \]

- **Forward Energy in Dynamic Programming**
  \[ M(i, j) = P(i, j) + \]
  \[ \min \left\{ \begin{array}{l}
  M(i - 1, j - 1) + C_L(i, j) \\
  M(i - 1, j) + C_U(i, j) \\
  M(i - 1, j + 1) + C_R(i, j)
  \end{array} \right\} \]


[Diagram showing energy function calculation with formulas for \( C_L \), \( C_U \), and \( C_R \).]
Experiment Result – Energy Function

1. Backward energy
2. Forward energy
3. Forward & Backward

Original image

removed seams

half-size image
Issues of Energy Function (2/2)

- Image edge energy: gradient map

- Saliency map:

- Face detection: Haar feature + Adaboost
Outline

- Introduction to Retargeting
- Image and Video Retargeting
  - Seam Carving
  - Forward Warping
- Our Modification on Video Retargeting
  - Seam Prediction Based on Motion Estimation
    - Forward, Backward
- Comparison of All Methods
- Conclusion and Future Work
- Reference
Motion Estimation-Forward Seam Prediction

- Motion vectors:

$$MV_i = \arg \min_{\vec{V}} \left\{ DFD_i(\vec{V}) + \lambda \sum_{B_j \in N(B_i)} |MV_j - \vec{V}| \right\}$$

- We want to predict the motion of seams by the recorded motion vectors in video files.
Motion Estimation-Forward Seam Prediction

- Motion map estimation:
  - Use motion vector to predict seam
  - Using Gaussian Kernel to diffuse importance of pixels

Previous frame

Current frame

Seam
Motion Estimation-Forward Seam Prediction

○ **Problem:** Motion vectors are recorded in original size

○ **Solution:**

1. **Original size**
   - Recording reduced seams
   - \( K \) seams reduced

2. **Frame: t-1**
   - Find the seam location in the original size image

3. **Frame: t**
   - Reduce the motion map by recorded seam locations at frame “t”
   - Modified dynamic programming range

4. **Building the motion map**
Minimizing energy and motion by dynamic programming

Just like estimating optical flow

- Assuming image intensity is constant

Brightness constancy Equation

\[ I(x, y, t) = I(x + dx, y + dy, t + dt) \]

\[ I_x dx + I_y dy + I_t dt = 0 \]

Considering gradient and motion as energy function

Advantage:

- Much faster than using graph cuts
- Only performing full search at the beginning of one shot
Motion Estimation-Backward Seam Prediction

Illustration:

$$M(i, j) = \left| \frac{\partial}{\partial x} I \right| + \left| \frac{\partial}{\partial y} I \right| + \left| \frac{\partial}{\partial t} I \right|$$

$$\min(M(i - 1, j - 1), M(i - 1, j), M(i - 1, j + 1))$$
Outline

- Introduction to Retargeting
- Image and Video Retargeting
  - Seam Carving
  - Forward Warping
- Our Modification on Video Retargeting
  - Seam Prediction Based on Motion Estimation
    - Forward, Backward
- Comparison of All Methods
- Conclusion and Future Work
- Reference
Comparison of Methods - Reducing Width

Original Image
size: 213*284

Resized Image size: 213*142

Seam Carving

Warping

Cropping

Scaling
Evaluating Retargeting Result – BDW(1/3)

- **Similarity Measurement : Bi-Directional Warping (BDW)**
  - Calculate distance between source(S) and target(T) images

\[
BDW(S, T) = \frac{1}{N_S} \sum_{i=1}^{h} A-\text{DTW}(S_i, T_i) + \frac{1}{N_T} \sum_{i=1}^{h} A-\text{DTW}(T_i, S_i)
\]

- **Based on Dynamic Time Warping (DTW)**
  - finding the optimal matching between S and T images
  - by non-linearly warping one to another (S→T and T→S)

Evaluating Retargeting Result – BDW(2/3)

Algorithm of Dynamic Time Warping

\[ \text{BDW}(S, T) = \frac{1}{N_S} \sum_{i=1}^{h} A-\text{DTW}(S_i, T_i) + \frac{1}{N_T} \sum_{i=1}^{h} A-\text{DTW}(T_i, S_i) \]

Algorithm 1: Asymmetric-DTW(s[1..|s|], t[1..|t|])

1: allocate \( M[|s| + 1][|t| + 1] \)
2: \( M[0, 0] := 0 \)
3: for \( i = 1 \) to \(|s|\) do
4: \( M[i, 0] := \infty \)
5: for \( j := 1 \) to \(|t|\) do
6: \( M[0, j] := 0 \)
7: for \( i := 1 \) to \(|s|\) do
8: \( \quad \) for \( j := 1 \) to \(|t|\) do
9: \( \quad \) \( M[i, j] := \min(M[i - 1, j - 1] + d(s[i], t[j]), M[i, j - 1], M[i - 1, j] + d(s[i], t[j])) \)
10: return \( M[|s|, |t|] \)

Dynamic programming and backtracking for optimal matching between source and target images.

Evaluating Retargeting Result – BDW(3/3)

BDW(S, T) = \frac{1}{N_S} \sum_{i=1}^{h} A-DTW(S_i, T_i) + \frac{1}{N_T} \sum_{i=1}^{h} A-DTW(T_i, S_i)

Testing frame:
Each patch should be matched to one patch in the reference frame

BDW Scoring – Seam Carving & Warping

Seam Carving  
A-DTW(S,T)  
A-DTW(T,S)  

Warping  
A-DTW(S,T)  
A-DTW(T,S)  

BDW Score:  
2724.2  
2328.9
BDW Scoring – Cropping & Scaling

Cropping

A-DTW(S,T)

A-DTW(T,S)

BDW Score: 2712.3

Scaling

A-DTW(S,T)

A-DTW(T,S)

BDW Score: 2277.8
BDW Scoring – Reducing Width

Original Image
size:213*284

Reduced Image size:213*142

Seam Carving:2724.2
Cropping:2712.3

Warping:2328.9
Scaling:2277.8
BDW Scoring - Reducing Height

Original Image size: 213*284

Reduced Image size: 150*284

Seam Carving: 796.2
Cropping: 2620.0
Warping: 1335.3
Scaling: 1424.1
Multi-Operator – Scaling & Seam Carving

original size: 213*284  desired size: 150*284

Step 1: Reducing size (< desired size)
Scaling down to 150*200

Step 2: Enlarging size (to desired size)
Performing Seam Carving to 150*284

BDW Score : 2647.2

Manual cropping to desired size

BDW Score : 2777.8
Comparison of Methods—More Results

Seam Carving  Warping  Cropping  Scaling
Comparison of Methods--Videos

- Demo time!!!
Outline

- Introduction to Retargeting
- Image and Video Retargeting
  - Seam Carving
  - Forward Warping
- Our Modification on Video Retargeting
  - Seam Prediction Based on Motion Estimation
    - Forward, Backward
- Comparison of All Methods
- Conclusion and Future Work
- Reference
Different kinds of methods have their own features and suitable cases!!!

- Content
- Structure

The motion estimation for seam movement still results in artifacts!!!
Future Works (1/3)

- How to improve the seam motion prediction
  - To reduce jittery artifacts due to lack of temporal coherency in video
- How to combine different types of retargeting methods to perform better results
- How to build a optimization criterion among important regions, the original frame size, and the desired frame size

How about building a bounding box?
Future Works (2/3)

- Bounding box of Saliency maps and face:
Future Works (3/3)

- Bounding box of Saliency maps and face:
Reference


Thank you for your attention!