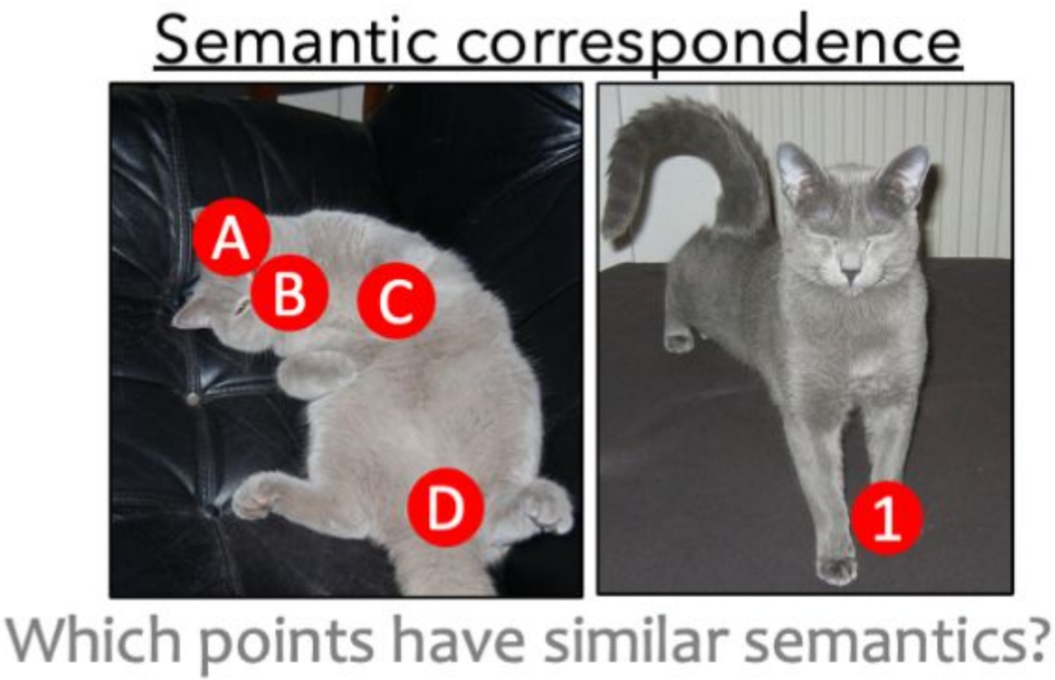
Lecture 9

Saliency and Retargeting

New publication from my group

New publication from my group



New publication from my group

Task	Vis.Corr.	Depth	Multi-view	Sem.Corr.	Forensic	Reflect.
Random	25.00	50.00	50.00	25.00	25.00	33.33
Human						
Gemini Pro						
GPT-4V						
Specialist						

New publication from my group

Task	Vis.Corr.	Depth	Multi-view	Sem.Corr.	Forensic	Reflect.
Random	25.00	50.00	50.00	25.00	25.00	33.33
Human	99.56	99.59	92.10	94.60	100.00	99.63
Gemini Pro						
GPT-4V						
Specialist						

New publication from my group

Task	Vis.Corr.	Depth	Multi-view	Sem.Corr.	Forensic	Reflect.
Random	25.00	50.00	50.00	25.00	25.00	33.33
Human	99.56	99.59	92.10	94.60	100.00	99.63
Gemini Pro						
GPT-4V						
Specialist	DIFT [69] 96.51	DepthAnything [80] 97.58	LoFTR [67] 90.22	DIFT [69] 71.22	DIRE [78] 68.94	Ordinal Shading [13] 77.61

These are models that utilize ideas we are learning in this class

New publication from my group

Task	Vis.Corr.	Depth	Multi-view	Sem.Corr.	Forensic	Reflect.
Random	25.00	50.00	50.00	25.00	25.00	33.33
Human	99.56	99.59	92.10	94.60	100.00	99.63
Gemini Pro	42.44	40.32	44.36	26.62	50.76	45.52
GPT-4V	33.72	59.68	55.64	28.78	34.09	38.81
Specialist	DIFT [69]	DepthAnything [80]	LoFTR [67]	DIFT [69]	DIRE [78]	Ordinal Shading [13]
	96.51	97.58	90.22	71.22	68.94	77.61

Administrative

A2 is out

- Due April 28th
- Date moved back

A3 is out

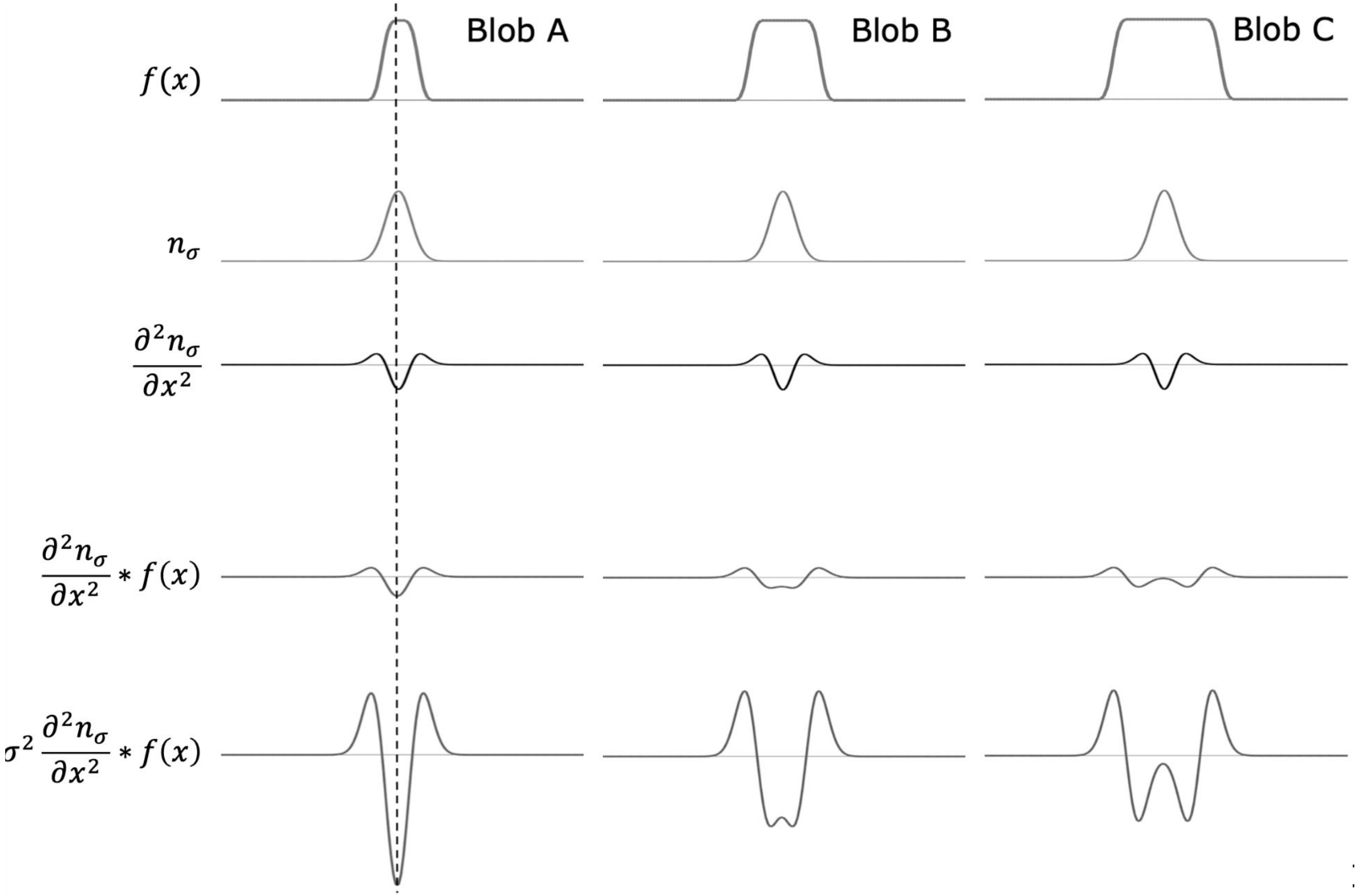
- Due May 9th

Administrative

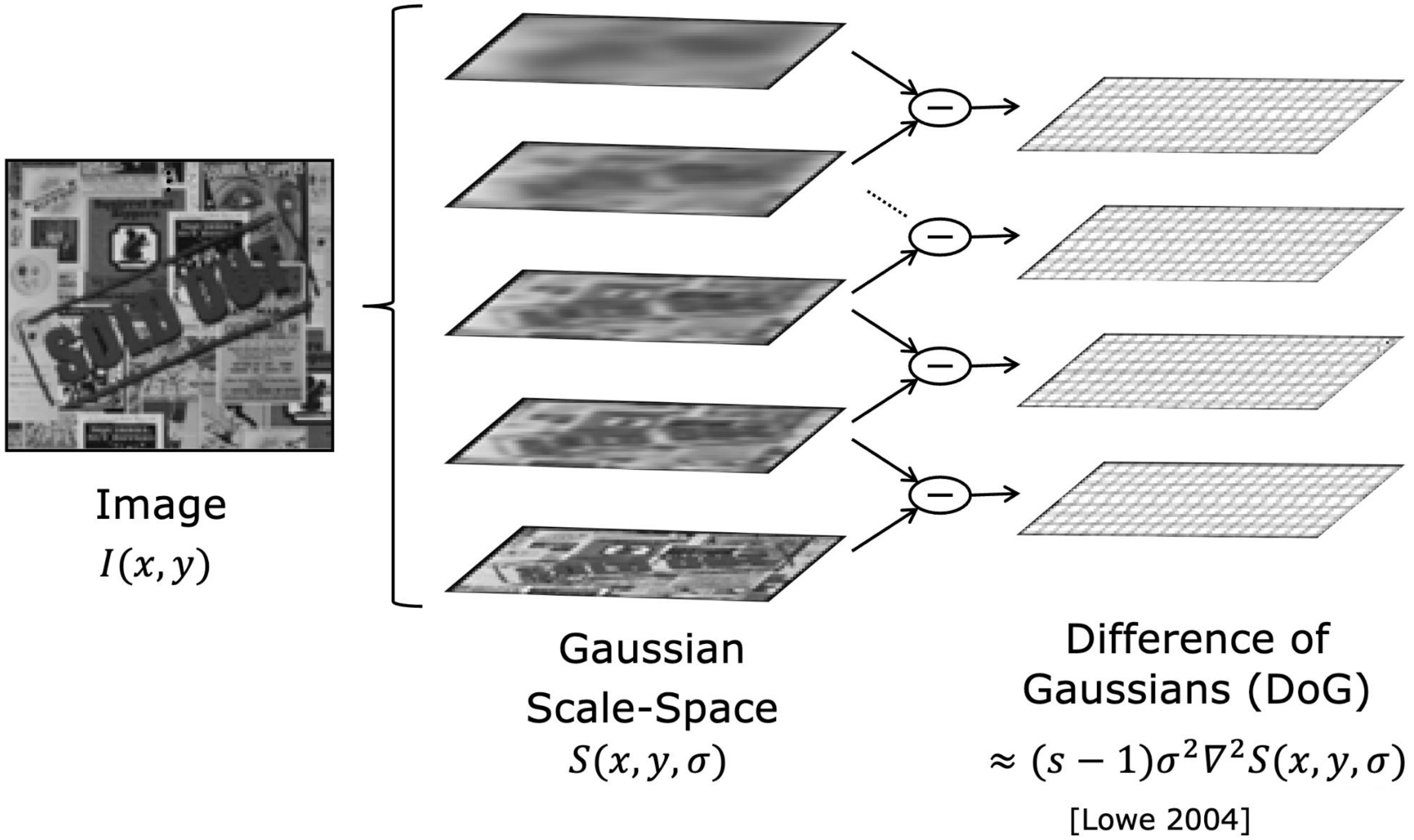
Recitation

- Xiaojuan Wang
- Panorama (part of your A2)
- detector, descriptor, RANSAC recap

So far: 1D example of how blobs are detected with LoG

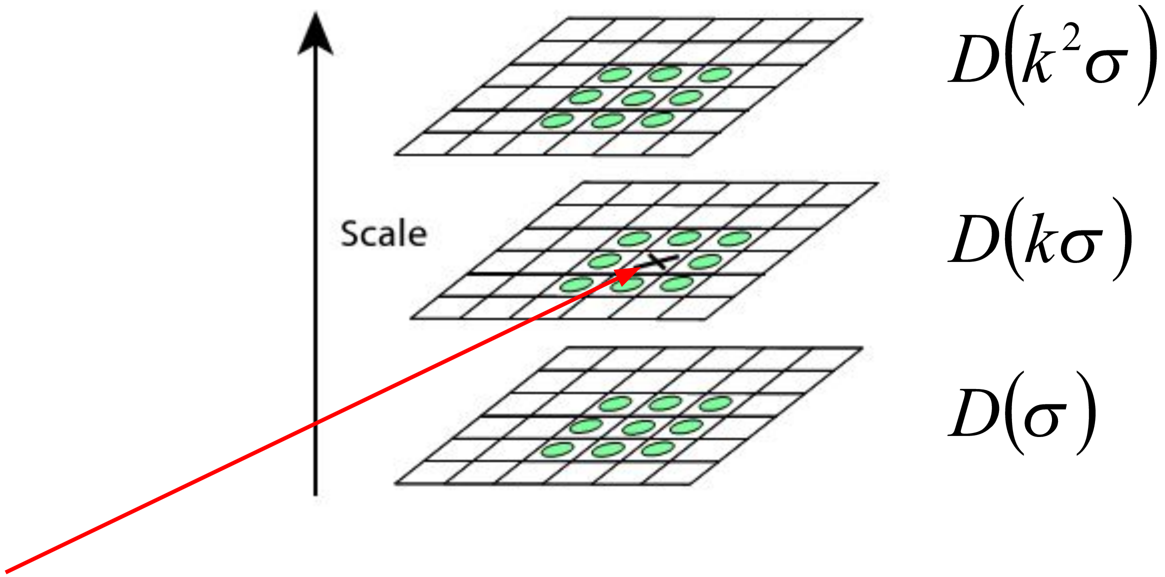


So far: SIFT detector algorithm



So far: Extracting SIFT keypoints and scales

- Choose the maxima within 3x3x3 neighborhood.



X is selected if it is larger or smaller than all 26 neighbors

Finishing up last lecture -> slide 64

Today's agenda

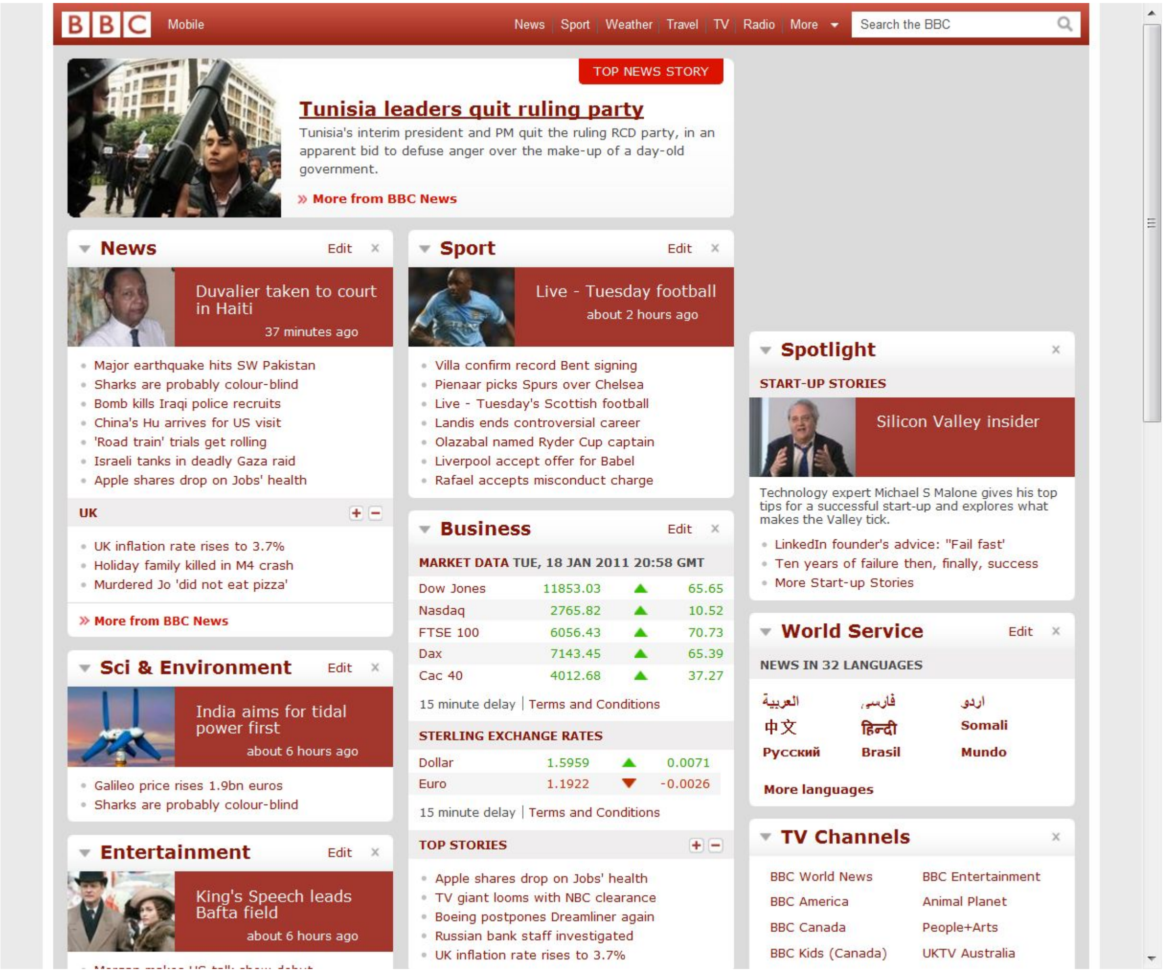
- Image retargeting
- Seam carving
- Dynamic programming
- Applications
- Forward algorithm

Today's agenda

- Image retargeting
- Seam carving
- Dynamic programming
- Applications
- Forward algorithm

Display Devices

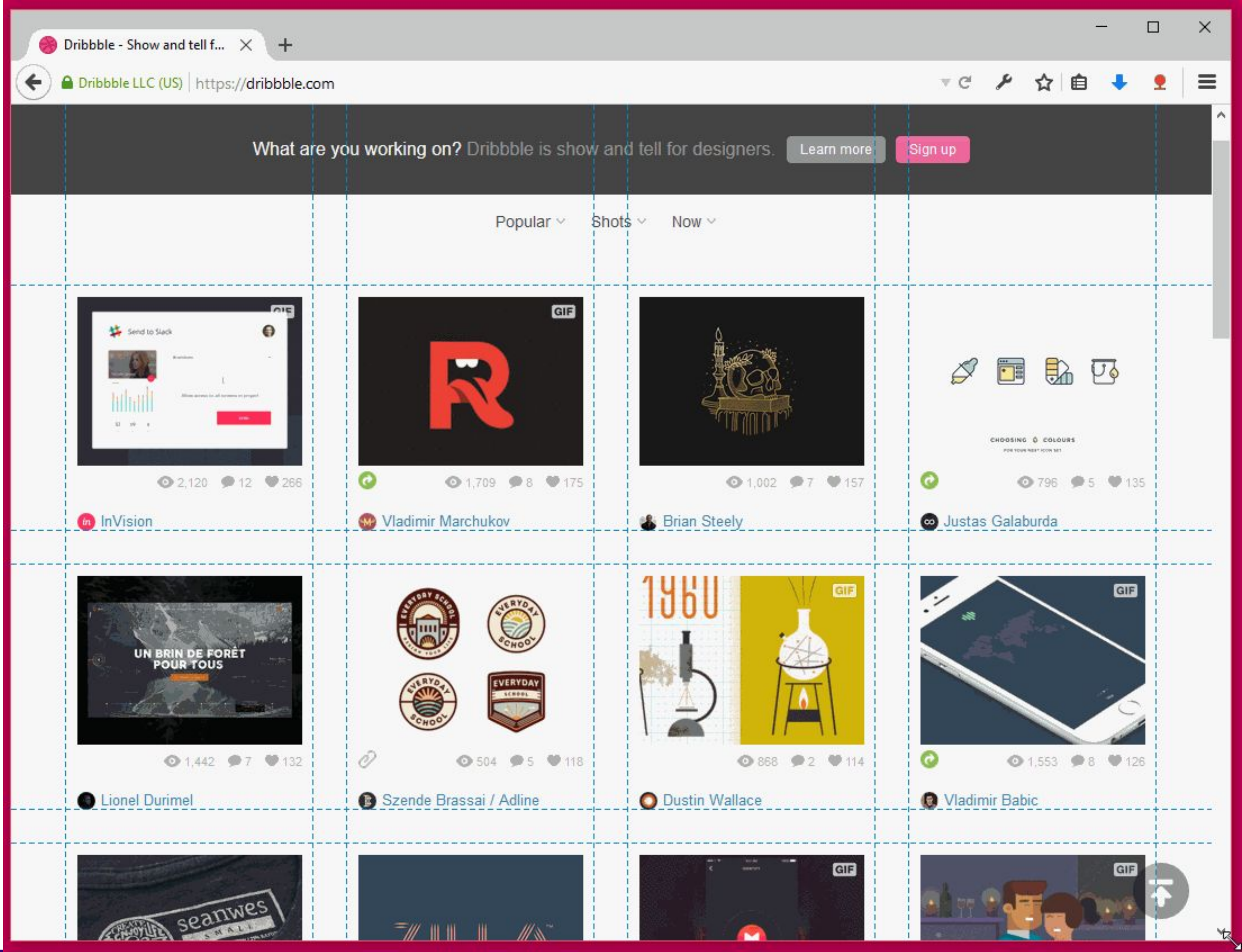
Content Retargeting



PC

iPhone

Page Layout



Dribbble - Show and tell f...
Dribbble LLC (US)
https://dribbble.com
What are you working on? Dribbble is show and tell for designers.
Learn more
Sign up
Popular
Shots
Now
InVision
Vladimir Marchukov
Brian Steely
Justas Galaburda
Lionel Durimel
Szende Brassai / Adline
Dustin Wallace
Vladimir Babic

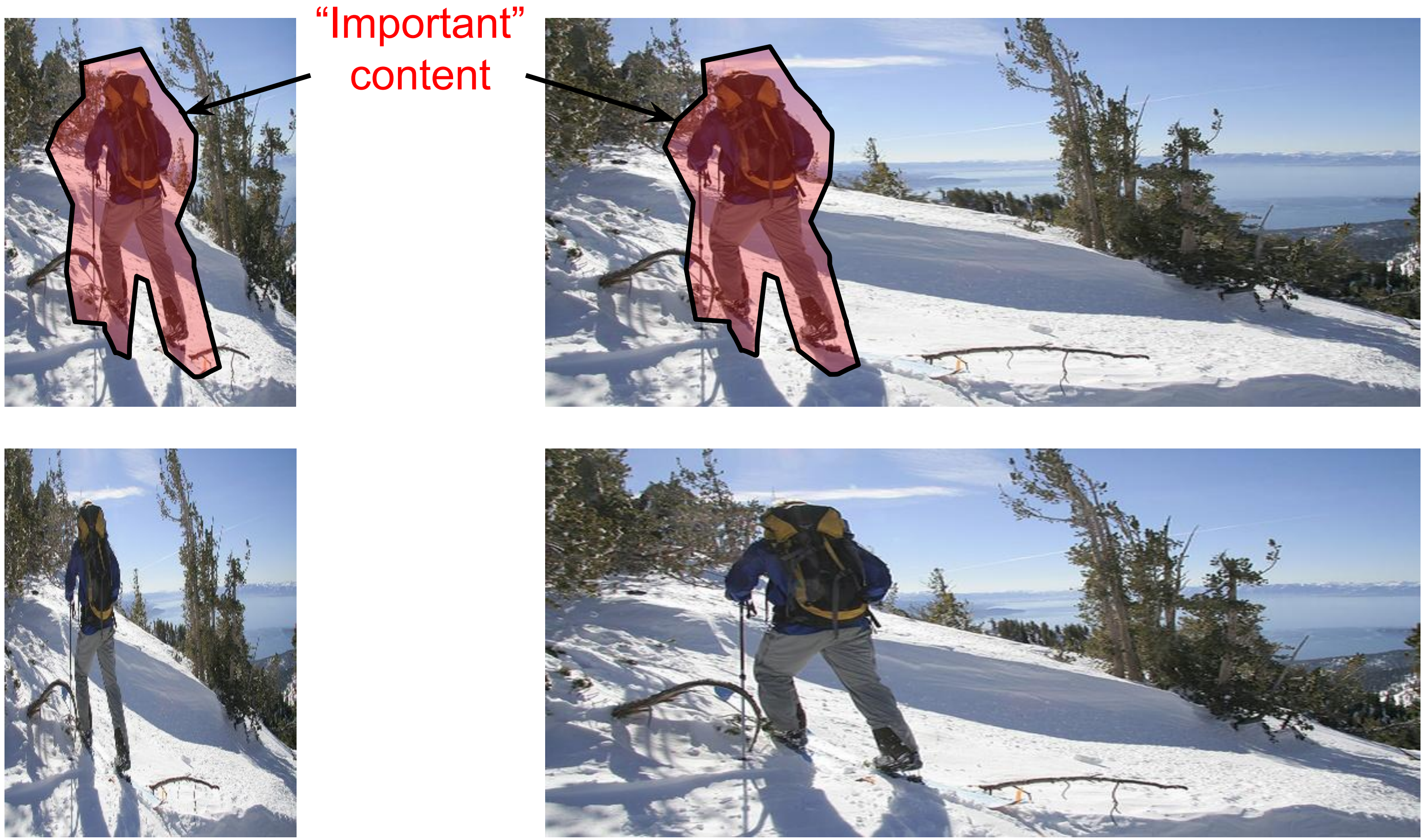
Simple Media Retargeting Operators

Letterboxing
Scaling

Content-aware Retargeting Operators

Content-*aware*

Content-*oblivious*



Content-aware Retargeting

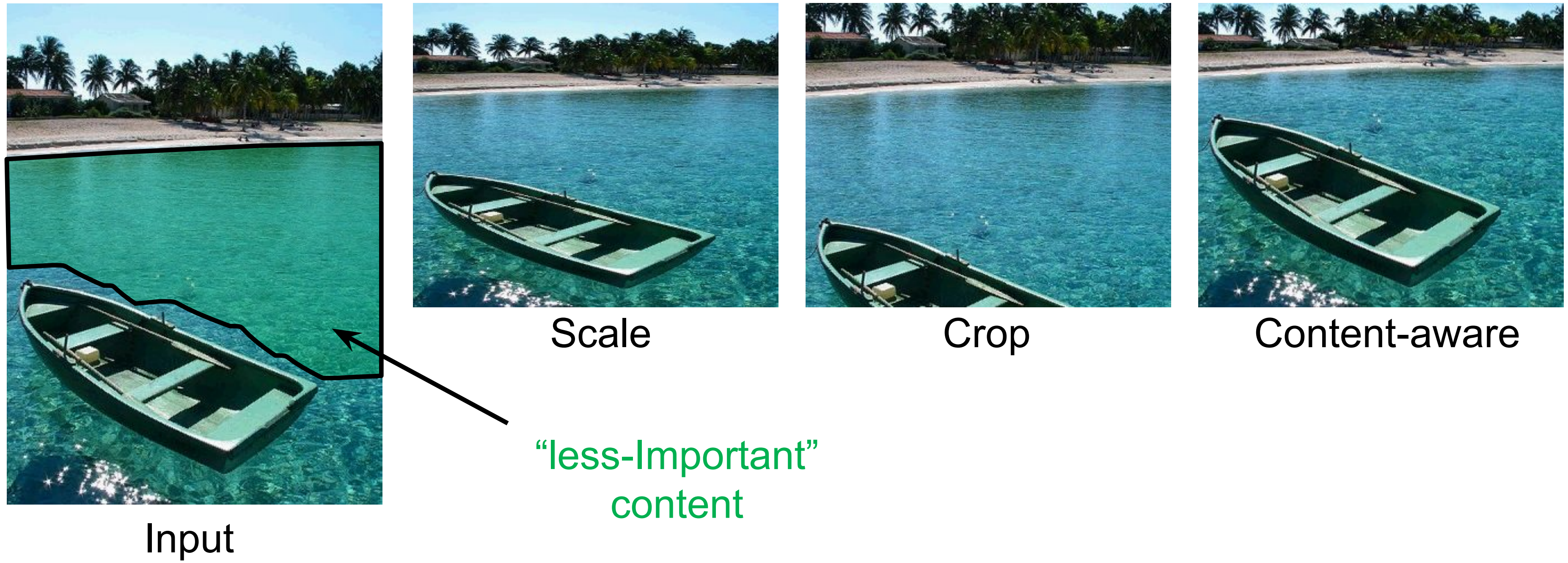


Image Retargeting

Problem statement

- Input image I of size n x m
- Output image I' of size n' x m'

Output image should be geometrically and semantically consistent with input image

- Till date, there is no formal definition of what constitutes as a "consistent" view.

How can we define consistency?

In large, we would expect retargeting to:

1. Adhere to the geometric constraints (display/aspect ratio)
2. Preserve the important content and structures
3. Limit artifacts

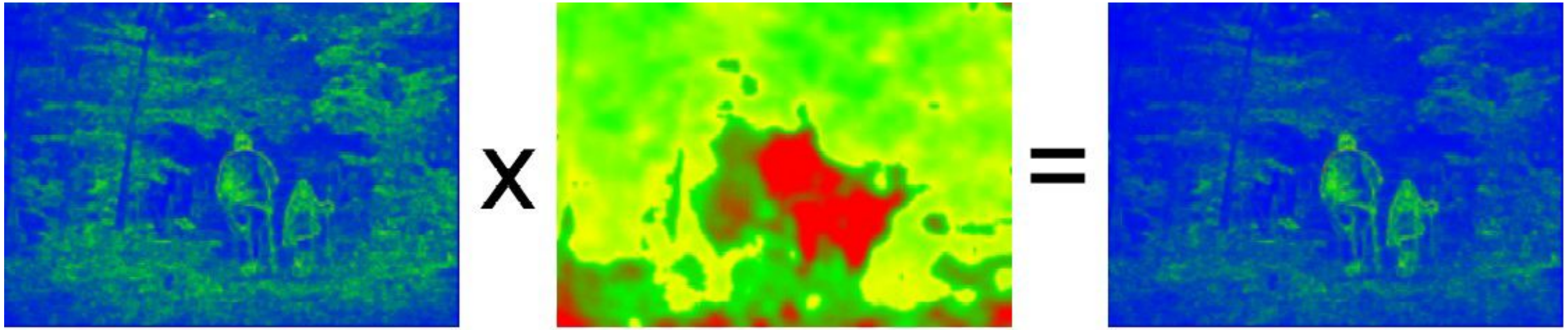
Very Ill-posed!

- How do we define what is important?
  - Is there a universal important vs unimportant?
- Would different people find different image regions more or less important?
- What about artistic impression in the original content?

Importance (Saliency) Measures

- A function $\mathcal{S} : n \times m \rightarrow [0, 1]$

- Ideas from human perception



First stage: coarse scan over entire image

Second stage: more focused attention on specific region

Wang et al. A Two-stage approach to saliency detection in images 2008

Importance (Saliency) Measures

- A function $\mathcal{S} : n \times m \rightarrow [0, 1]$

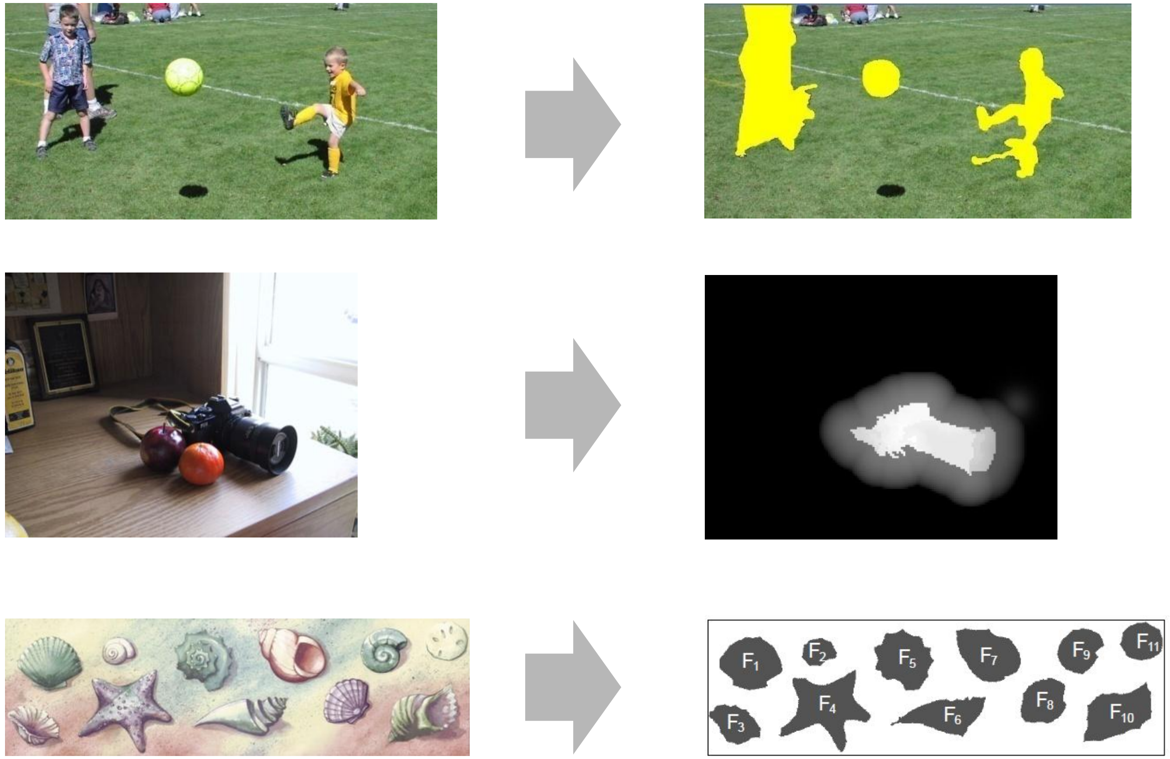
- More sophisticated: attention models, eye tracking (gazing studies), face detectors, …

Judd et al. *Learning to predict where people look* ICCV 2009

General Retargeting Framework

General Retargeting Framework

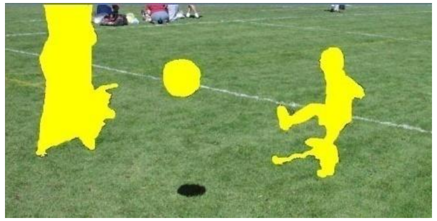
Step 1. Define an energy function
E(I) (interest, importance, saliency)



General Retargeting Framework

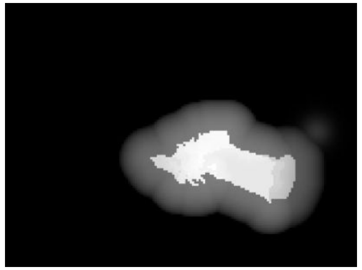
Step 1. Define an energy function $\mathbf{E(I)}$ (interest, importance, saliency)

Step 2. Use some operator(s) to change the image $\mathbf{I}$

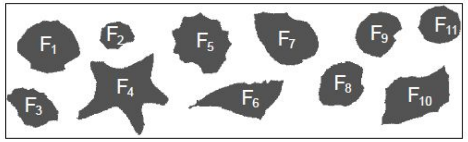
Recompose

Setlur et al. [2005]

Crop

Santella et al. [2005]

Warp

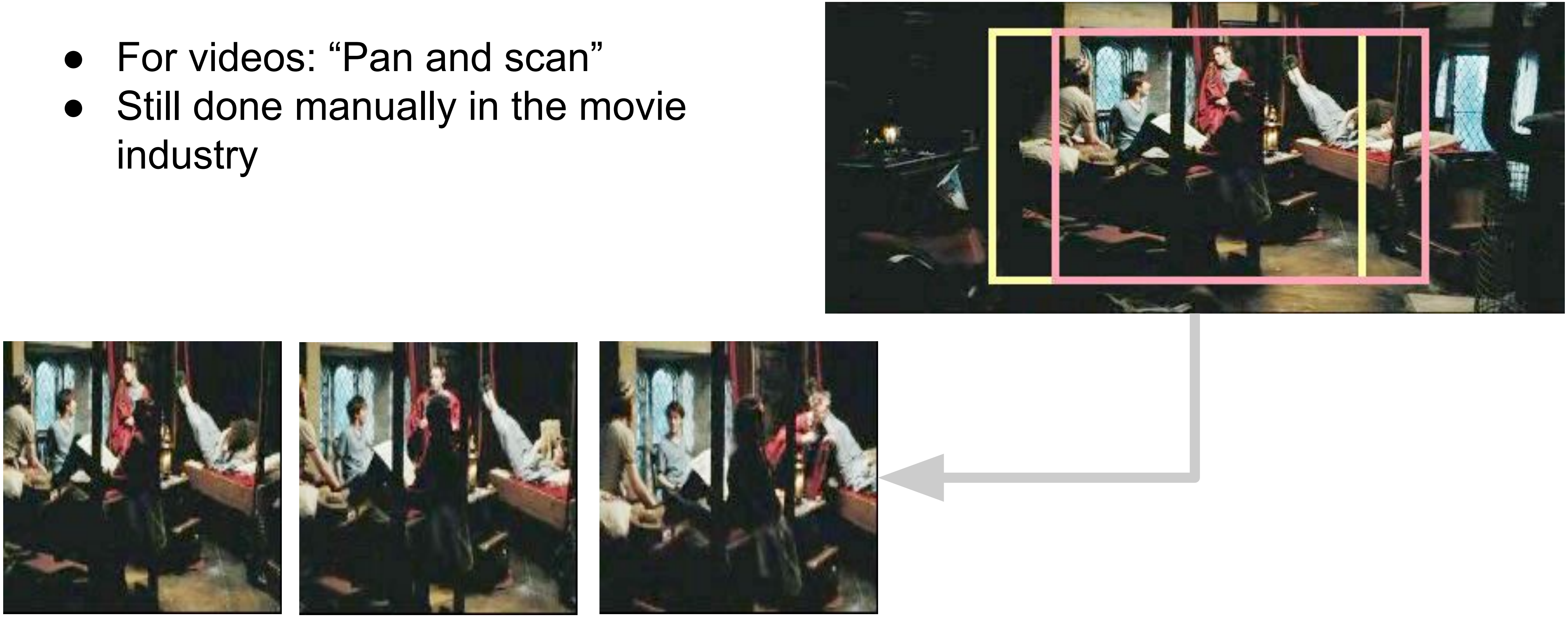
Gal et al. [2006]

Potential Retargeting Approaches

- Optimal Cropping Window

Potential Retargeting Approaches

- For videos: “Pan and scan”
- Still done manually in the movie industry



Liu and Gleicher, Video Retargeting: Automating Pan and Scan (2006)

Cropping

Today's agenda

- Image retargeting
- Seam carving
- Dynamic programming
- Applications
- Forward algorithm

Seam Carving

- Assume input I is size m x n
- Output I is m x n',
  - where n'<n

- Basic Idea: remove unimportant pixels from the image
  - Unimportant = pixels with less "energy"

$$E(I) = |\frac{\partial I}{\partial x}| + |\frac{\partial I}{\partial y}|$$

$$E(I) = \sqrt{(\frac{\partial I}{\partial x})^2 + (\frac{\partial I}{\partial y})^2}$$

- Intuition for gradient-based energy:
  - Preserve edges
  - Human vision more sensitive to edges – so try remove content from smoother areas
  - Simple enough for producing some nice results

Let’s do a though experiment

We calculate the energy for this image.

Q1. Can we just remove the K **pixels** with the lowest energy?

Let’s do a though experiment

We calculate the energy for this image.

Q1. Can we just remove the K **pixels** with the lowest energy?

Q2. Can we remove the K **rows** with the lowest energies?

Let’s do a though experiment

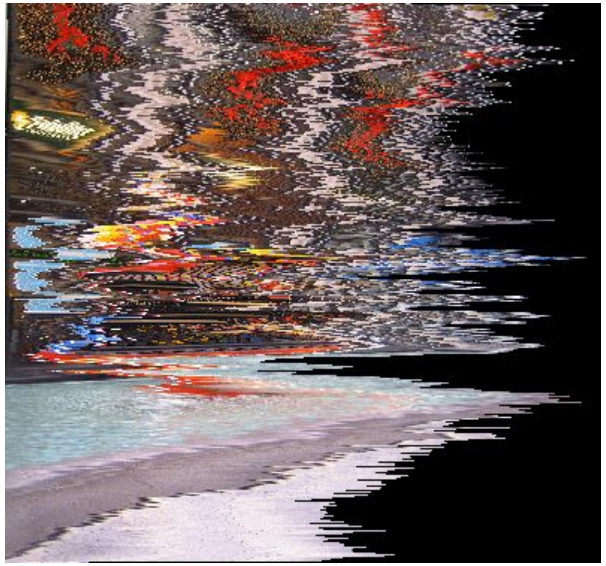
We calculate the energy for this image.

Q1. Can we just remove the K **pixels** with the lowest energy?

Q2. Can we remove the K **rows** with the lowest energies?

Q3. Can we remove the K **columns** with the lowest energies?

Pixel Removal



Optimal

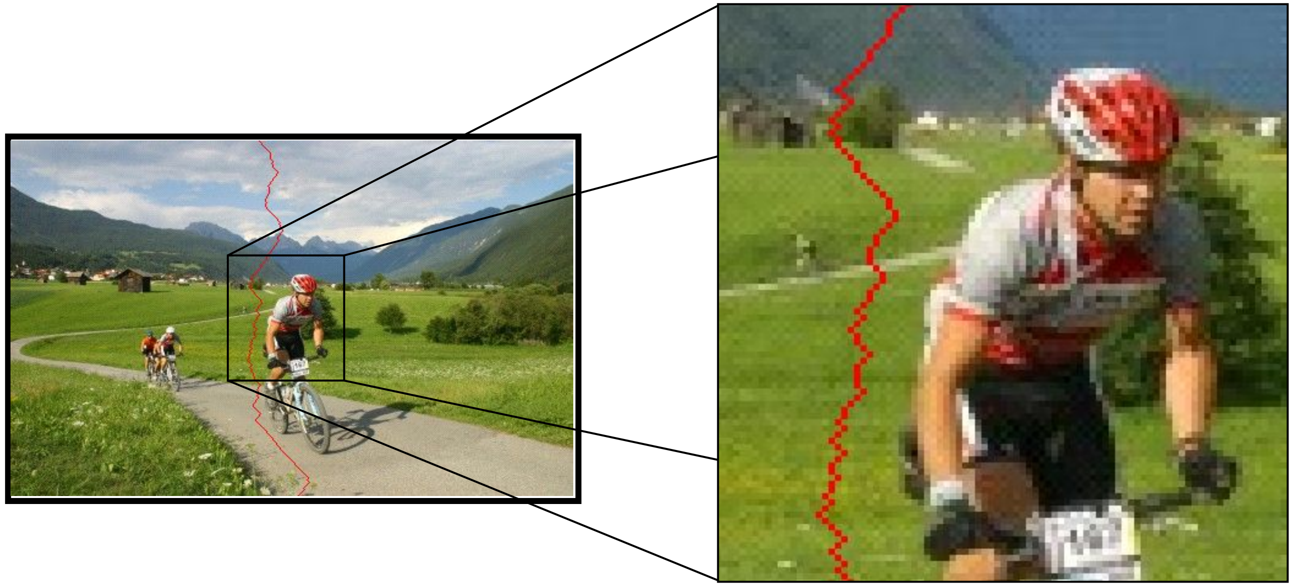
Least-energy pixels (per row)

Least-energy columns

Solution: A Seam

- A seam is a connected path of pixels from top to bottom (or left to right). Exactly one in each row (or column)

$$s^x = \{s_i^x\}_{i=1}^{n}$$



Solution: A Seam

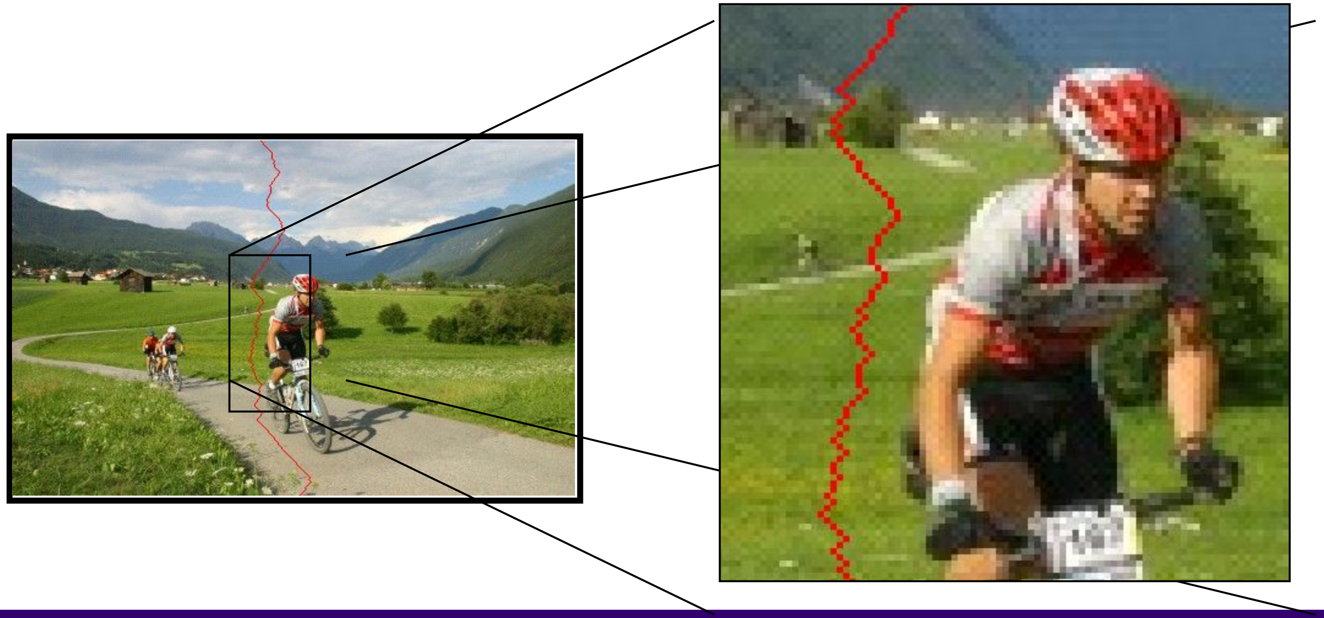
- A seam is a connected path of pixels from top to bottom (or left to right). Exactly one in each row (or column)

$$s^x = \{s_i^x\}_{i=1}^n$$

$$s^x = \{x(i), i\}_{i=1}^n$$

for every row i

find the column with the lowest energy



Solution: A Seam

- A seam is a connected path of pixels from top to bottom (or left to right). Exactly one in each row (or column)

$$s^x = \{s_i^x\}_{i=1}^n$$

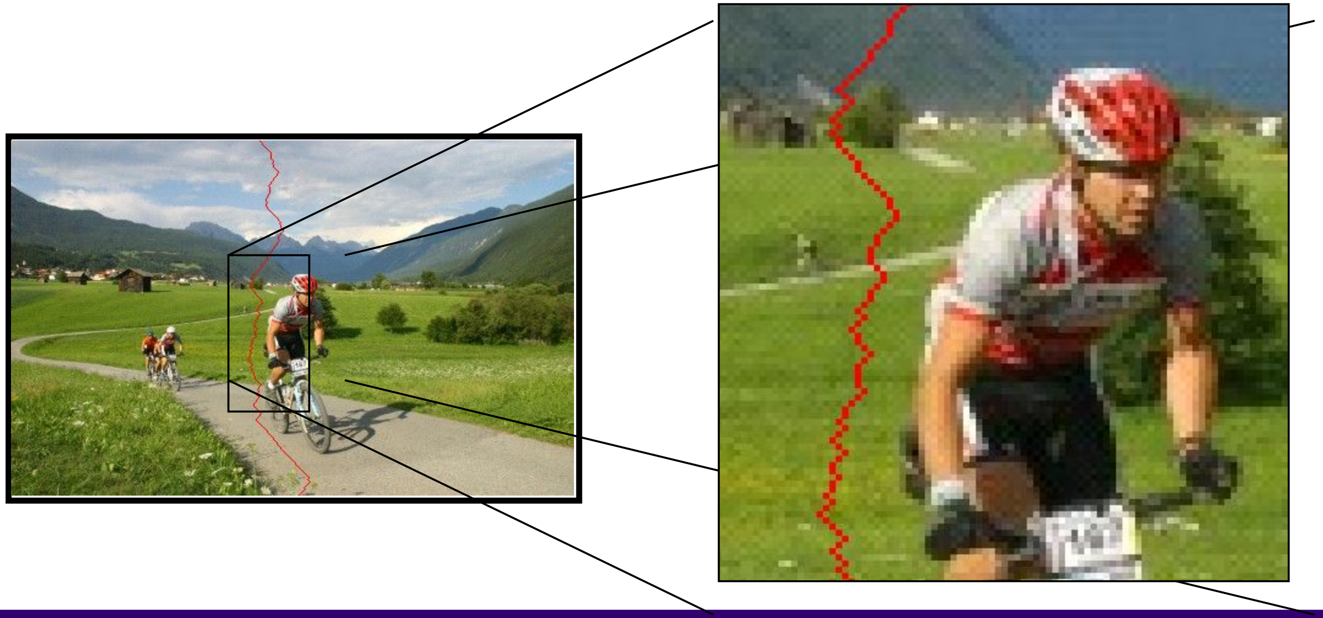
$$s^x = \{x(i), i\}_{i=1}^n$$

for every row i

find the column with the lowest energy

$$\text{s.t. } \forall i, \ |x(i) - x(i-1)| \leq 1$$

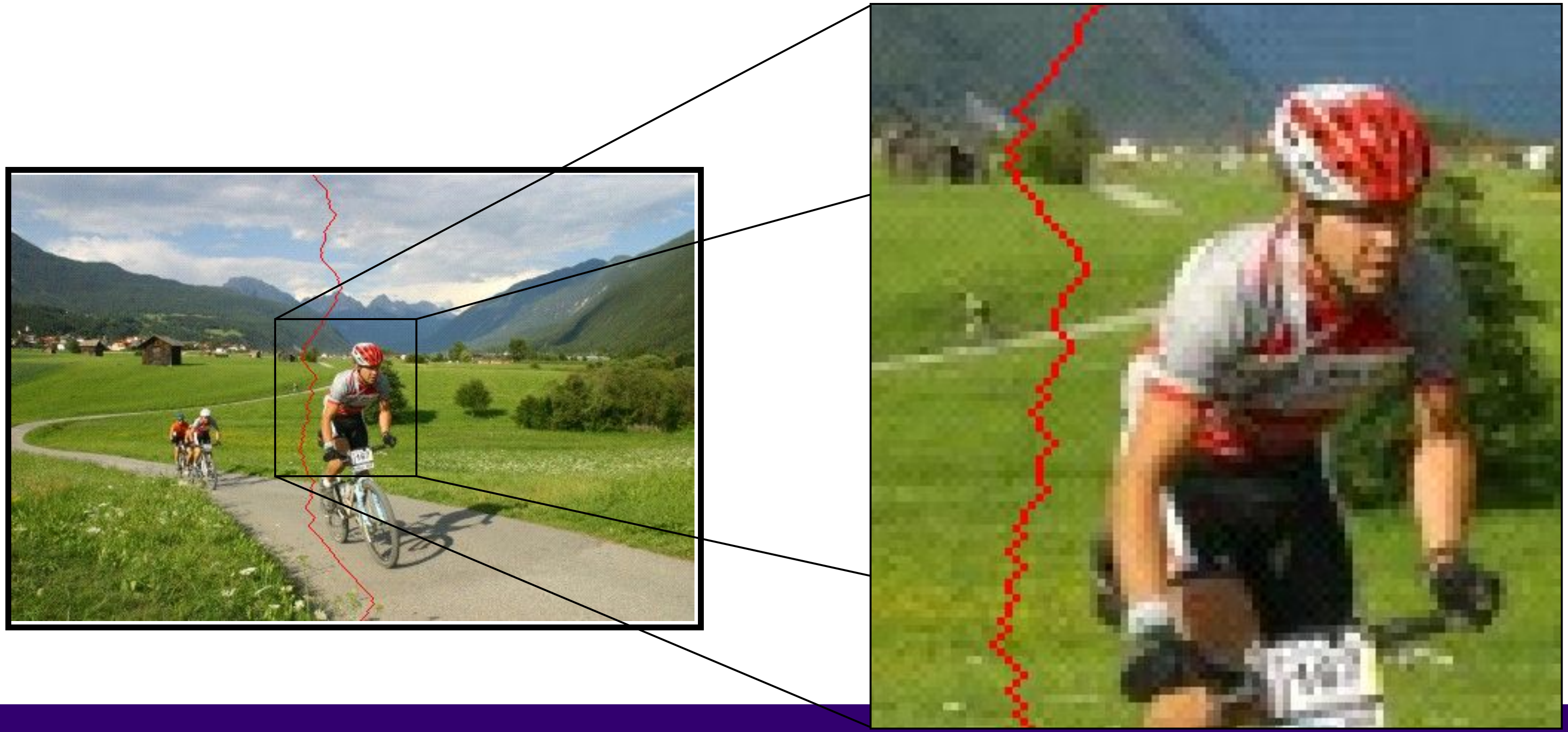
Ensure that seam is "connected". Columns can only change by a maximum of 1 column



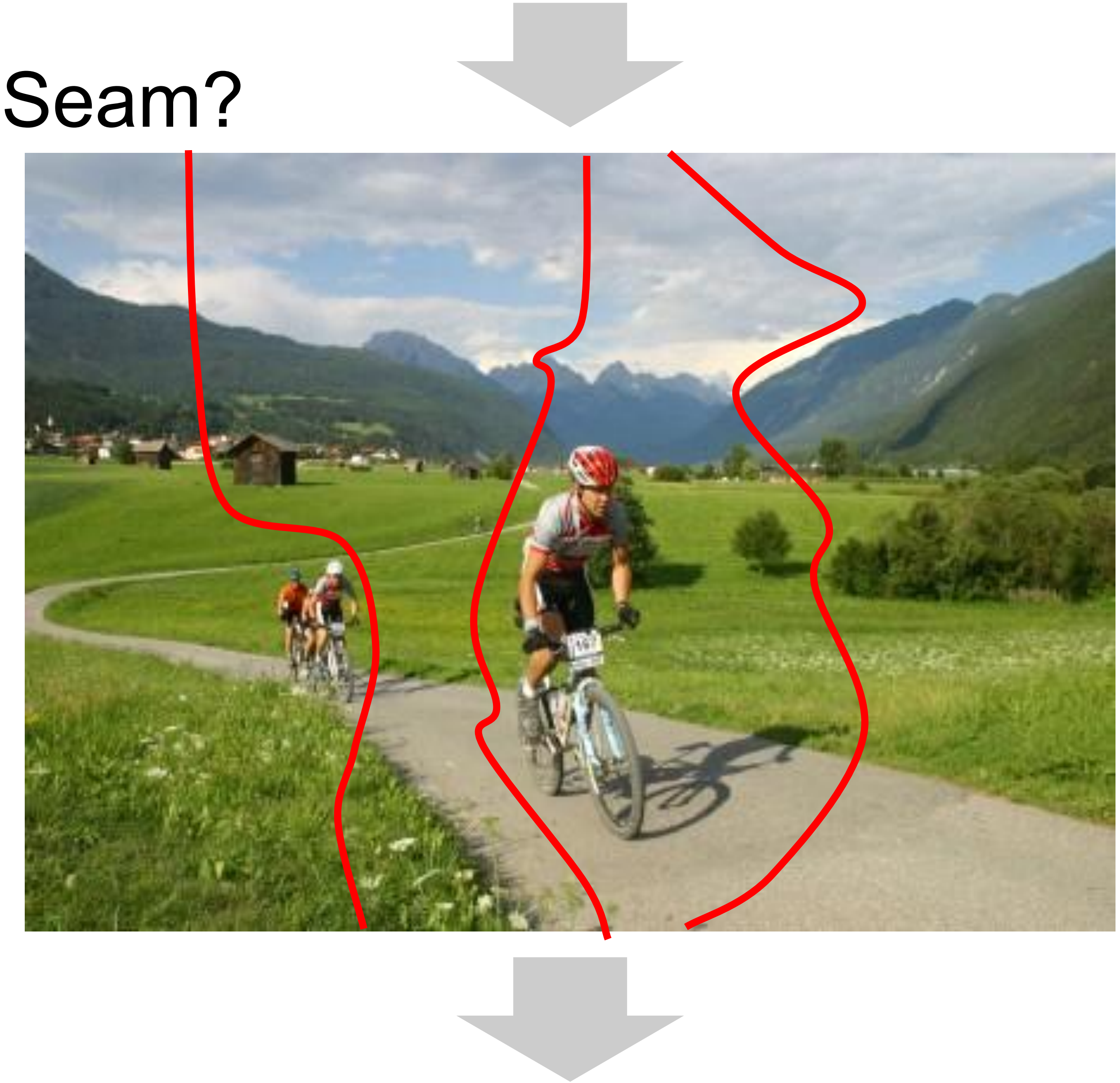
A Seam

- A connected path of pixels from top to bottom (or left to right). Exactly one in each row

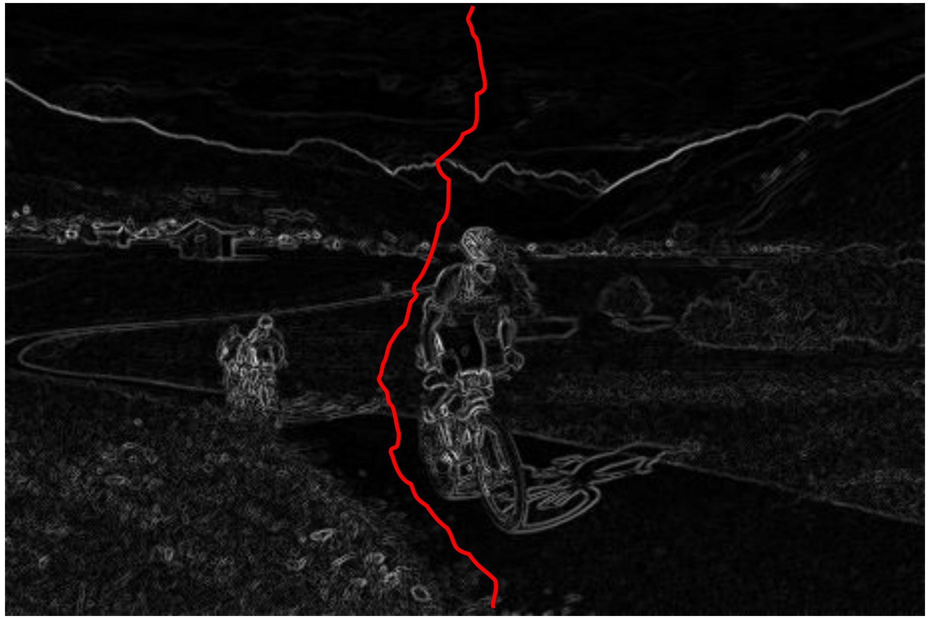
$$\mathbf{s^x} = \{s_i^x\}_{i=1}^n = \{(x(i), i)\}_{i=1}^n, \text{ s.t. } \forall i, |x(i) - x(i-1)| \leq 1$$
$$\mathbf{s^y} = \{s_j^y\}_{j=1}^m = \{(j, y(j))\}_{j=1}^m, \text{ s.t. } \forall j |y(j) - y(j-1)| \leq 1$$



How do we find the optimal Seam?



The Optimal Seam



$$E(\mathbf{I}) = |\frac{\partial}{\partial x}\mathbf{I}| + |\frac{\partial}{\partial y}\mathbf{I}| \Rightarrow s^* = \arg\min_s E(s)$$

Today’s agenda

- Image retargeting
- Seam carving
- Dynamic programming
- Applications
- Forward algorithm

Dynamic Programming

Input: Given an energy E(i, j)

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

- Create a cost matrix M with the following property:
  - M(i, j) = minimal cost of a seam going through pixel (i, j)
    - starting from j=0

M(i, j)

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

M(i, 0) = E(i, 0) of a seam going through pixel (i, j)

M(i, j)

5	8	12	3

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

Q. What do you think should be this value?

M(i, j)

5	8	12	3
	?		

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

M(i, j) = total energy of seam going through pixel (i, j) from j=0

5	8	12	3
	2+5		

M(i, j)

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

The recurrence formula

$$\mathbf{M}(i,j) = E(i,j) + \min\left(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\right)$$

5	8	12	3
	2+5		

M(i, j)

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

5	8	12	3
	7		

M(i, j)

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

$$\mathbf{M}(i,j) = E(i,j) + \min\big(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\big)$$

5	8	12	3
	7	?	

M(i, j)

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

$$\mathbf{M}(i,j) = E(i,j) + \min\left(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\right)$$

M(i, j)

5	8	12	3
	7	3+3	

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Dynamic Programming

$$\mathbf{M}(i,j) = E(i,j) + \min\big(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\big)$$

M(i, j)

5	8	12	3
9	7	6	12
14	9	10	8
14	14	15	8+8

5	8	12	3
4	2	3	9
7	3	4	2
5	5	7	8

Energy - E(i, j)

Searching for minimum seam

Backtrack: Find the minimum M(i, j=m)

5	8	12	3
9	7	6	12
14	9	10	8
14	14	15	16

M(i, j)

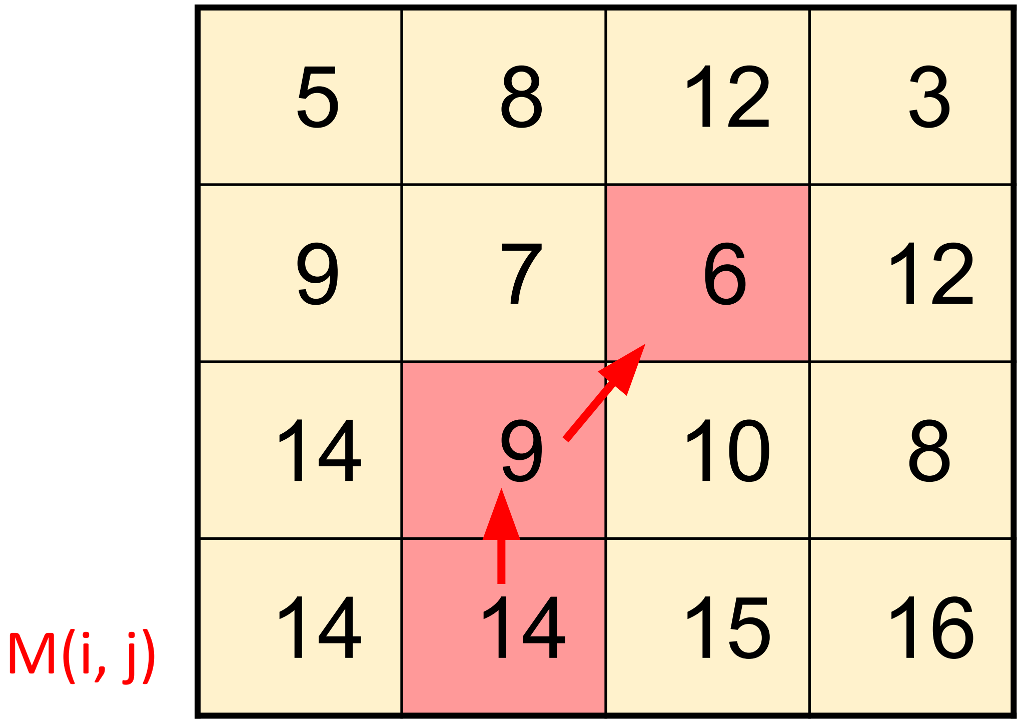
This is the minimum in the last row

Backtrack

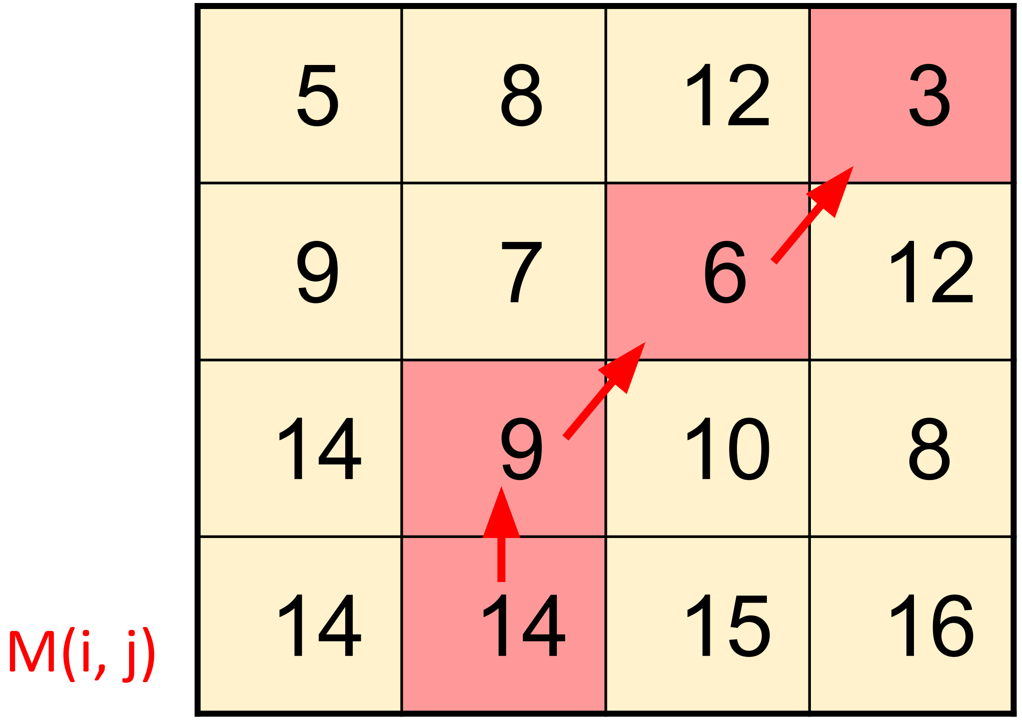
After finding minimum M(i, j) at row j,
find minimum M(i, j-1) but only be looking at neighboring locations: i-1, i, i+1

	5	8	12	3
	9	7	6	12
	14	9	10	8
M(i, j)	14	14	15	16

Searching for Minimum



Searching for Minimum



The Optimal Seam - dynamic programming

- The recursion relation

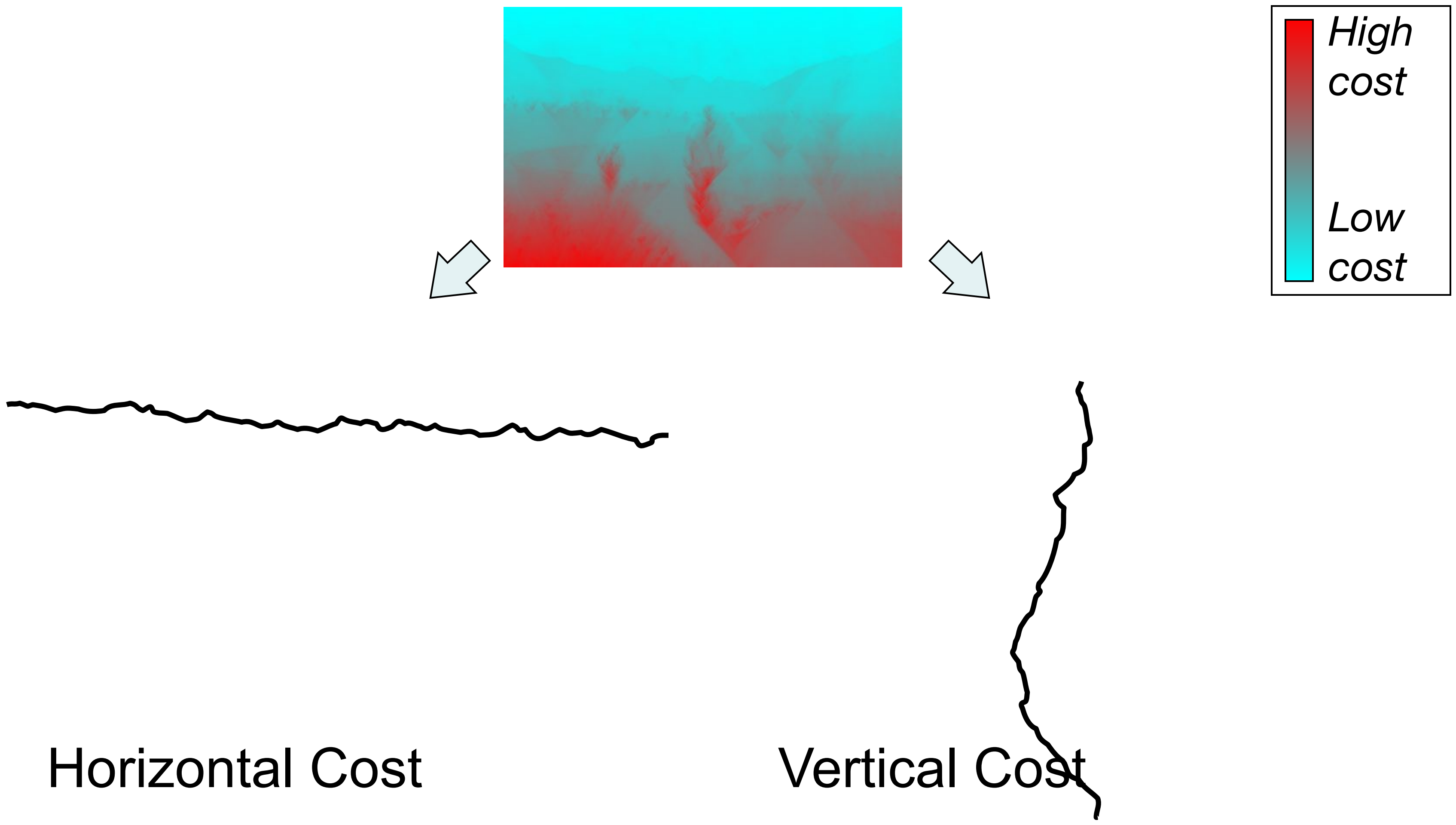
$$\mathbf{M}(i,j) = E(i,j) + \min\left(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\right)$$

- Can be solved efficiently using dynamic programming in

$$O(s \cdot n \cdot m)$$

(s=3 in the original algorithm)

Horizontal and vertical cost maps



Seam Carving

The Seam-Carving Algorithm

Algorithm: Seam carving
Input: Image I of size m x n
Output: Image I' of size m x n' where n' < n

I' = I

Do d=(n-n') times

    Compute energy map on I'

    Find optimal seam in E

    Remove s from im

Return I'

For vertical resize: transpose the image

Running time: O(dmn)

Changing Aspect Ratio

Another example

Example seam carving

Example seam carving

Changing Aspect Ratio

Original

Retargeting

Scaling

Changing Aspect ratio

Cropping

Retarget

Scaling

Changing Aspect Ratio

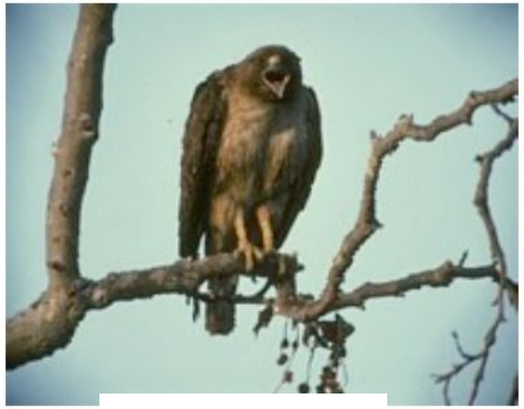
Original

Retarget

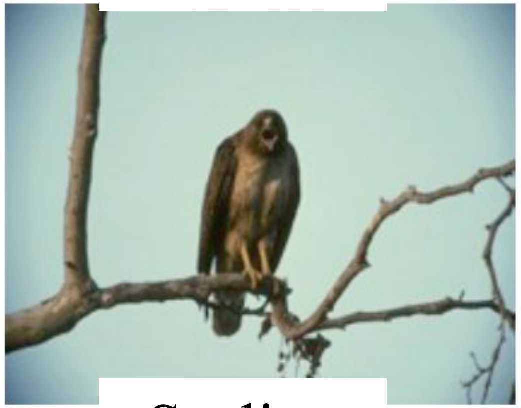
Scaling

Changing Aspect Ratio

Original



Retarget



Scaling

Questions

- Q: Will the result be the same if the image is flipped upside down?

Q. What if we simultaneously want to reduce both width and height?

m x n -> m’ x n’

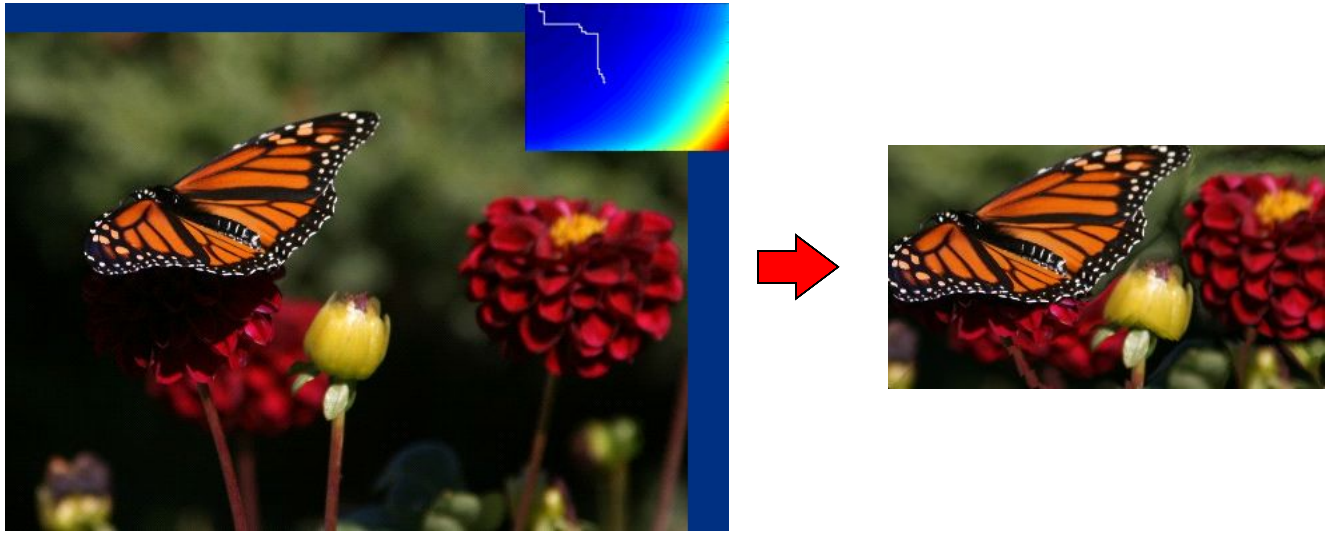
1. Should we remove horizontal seam first?
2. Should we remove vertical seams first?
3. Alternate between the two?
4. Any other ideas?

What if we simultaneously want to reduce both width and height?

m x n -> m' x n'

1. Should we remove horizontal seam first?
2. Should we remove vertical seams first?
3. Alternate between the two?
4. Any other ideas?

The optimal order can be found! Dynamic Prog (again)



Retargeting in Both Dimensions

- Let T(r,c) denote a new cost matrix of obtaining an image of size (n-r)x(m-c).

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

Retargeting in Both Dimensions

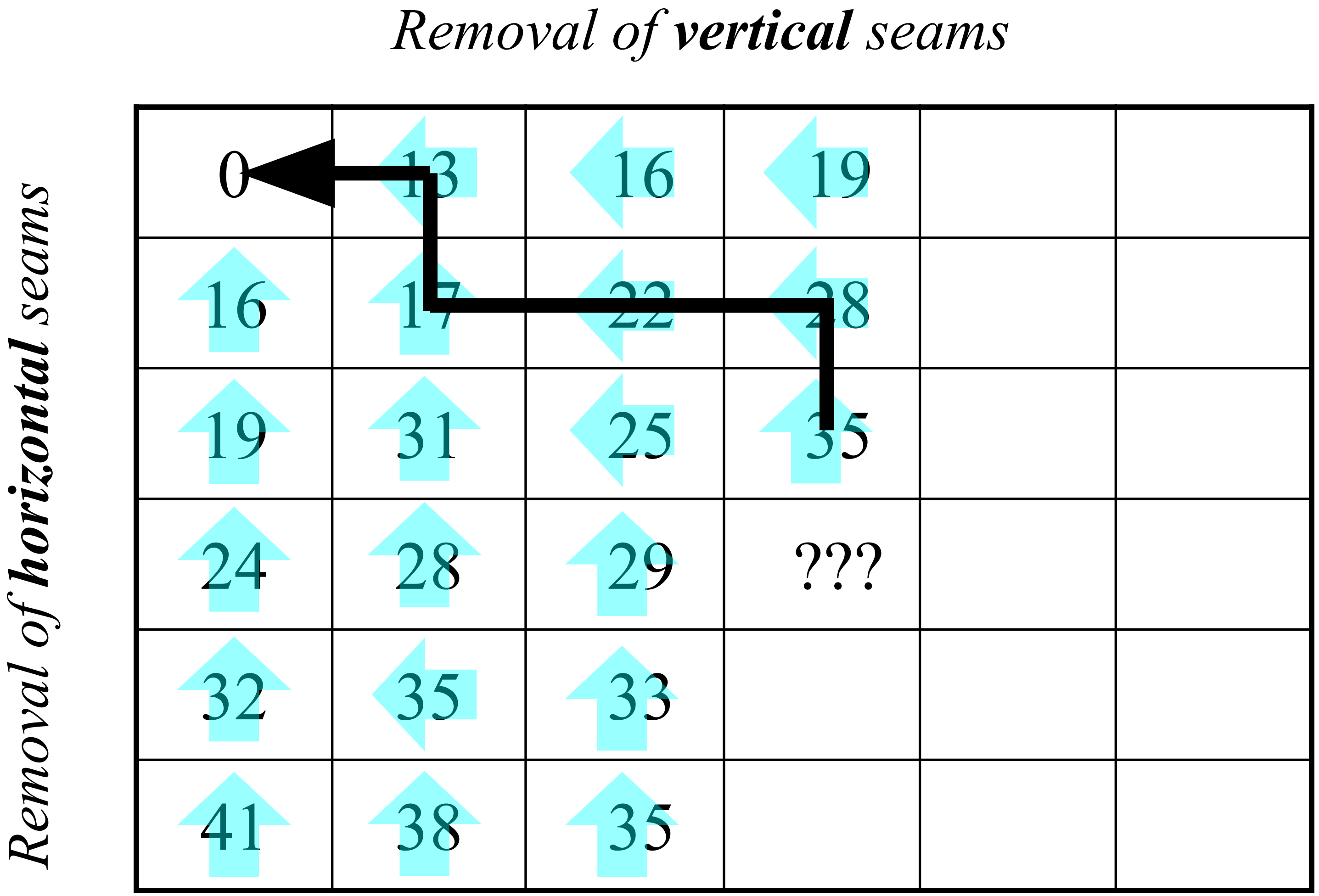
- Let T(r,c) denote a new cost matrix of obtaining an image of size (n-r)x(m-c).

$$\begin{aligned}\mathbf{T}(r,c) &= min(\mathbf{T}(r-1,c) + E(\mathbf{s^x}(\mathbf{I_{n-r-1\times m-c}})), \\ &\qquad \mathbf{T}(r,c-1) + E(\mathbf{s^y}(\mathbf{I_{n-r\times m-c-1}})))\end{aligned}$$

where $E(\mathbf{s^x}(\mathbf{I_{n-r-1\times m-c}}))$ is the cost of removing a horizontal

seam from the image $\mathbf{I_{n-r-1\times m-c}}$

Optimal Order Map



Is it optimal…

- … for removing ONE seam?

- … for removing multiple seams?

Is it optimal…

- … for removing ONE seam?

- … for removing multiple seams?

    - Consider HVV (how many possible orderings?)
    - Cost(V) on HV not necessarily equal Cost(V) on VH
    - But we keep track of only one: min(HV,VH)…

Today's agenda

- Image retargeting
- Seam carving
- Dynamic programming
- Applications
- Forward algorithm

Image expansion - Repeat the lowest energy seam?

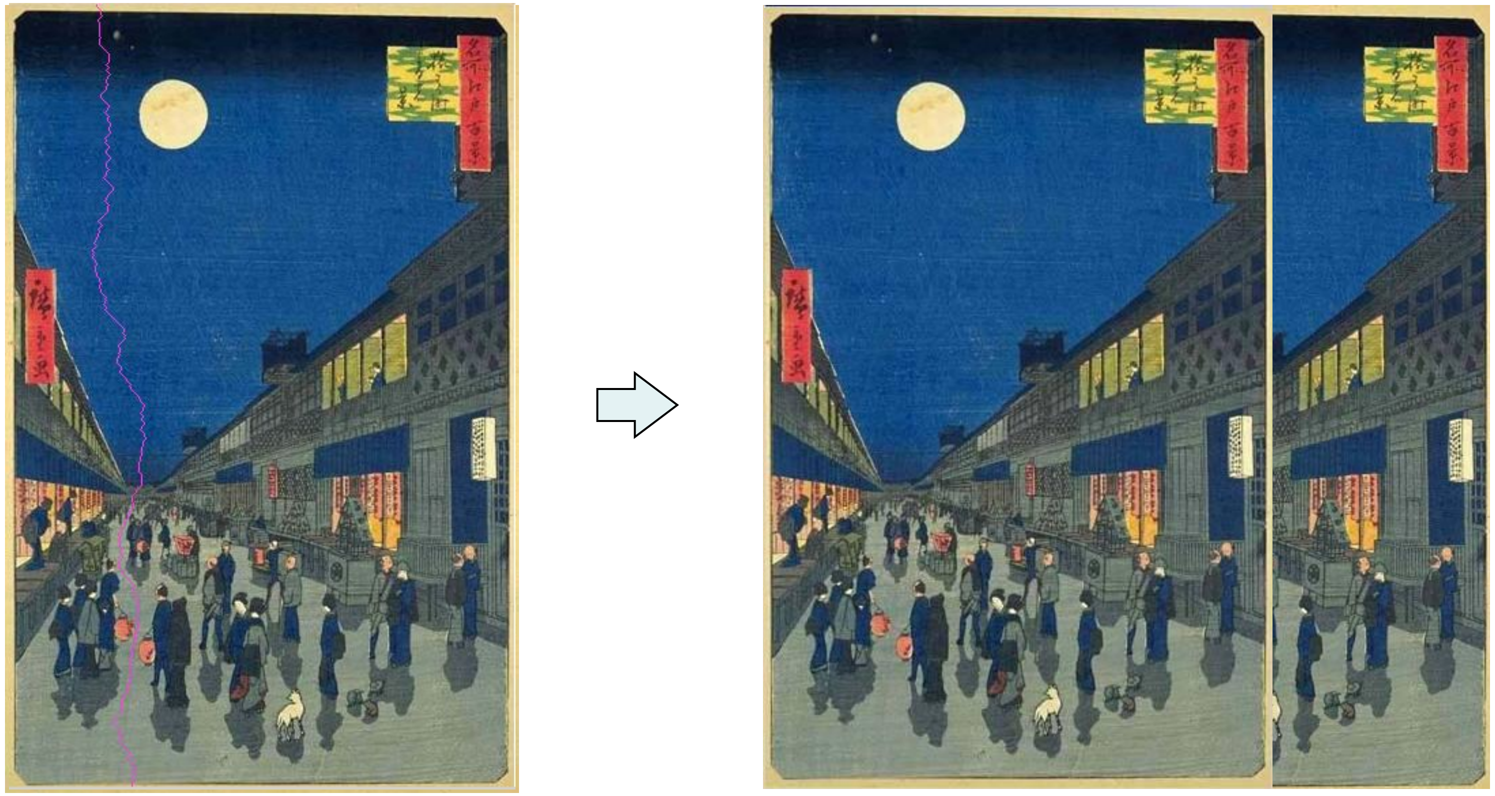
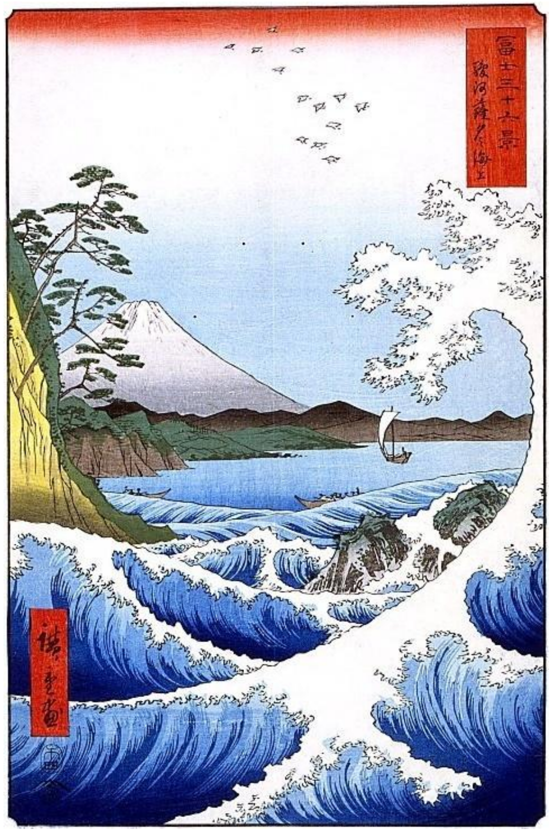


Image Expansion – Repeat the K lowest energy seams

Scaling

Can you tell if this image has been enlarged or reduced?



Combined Insert and Remove

Insert & remove seams

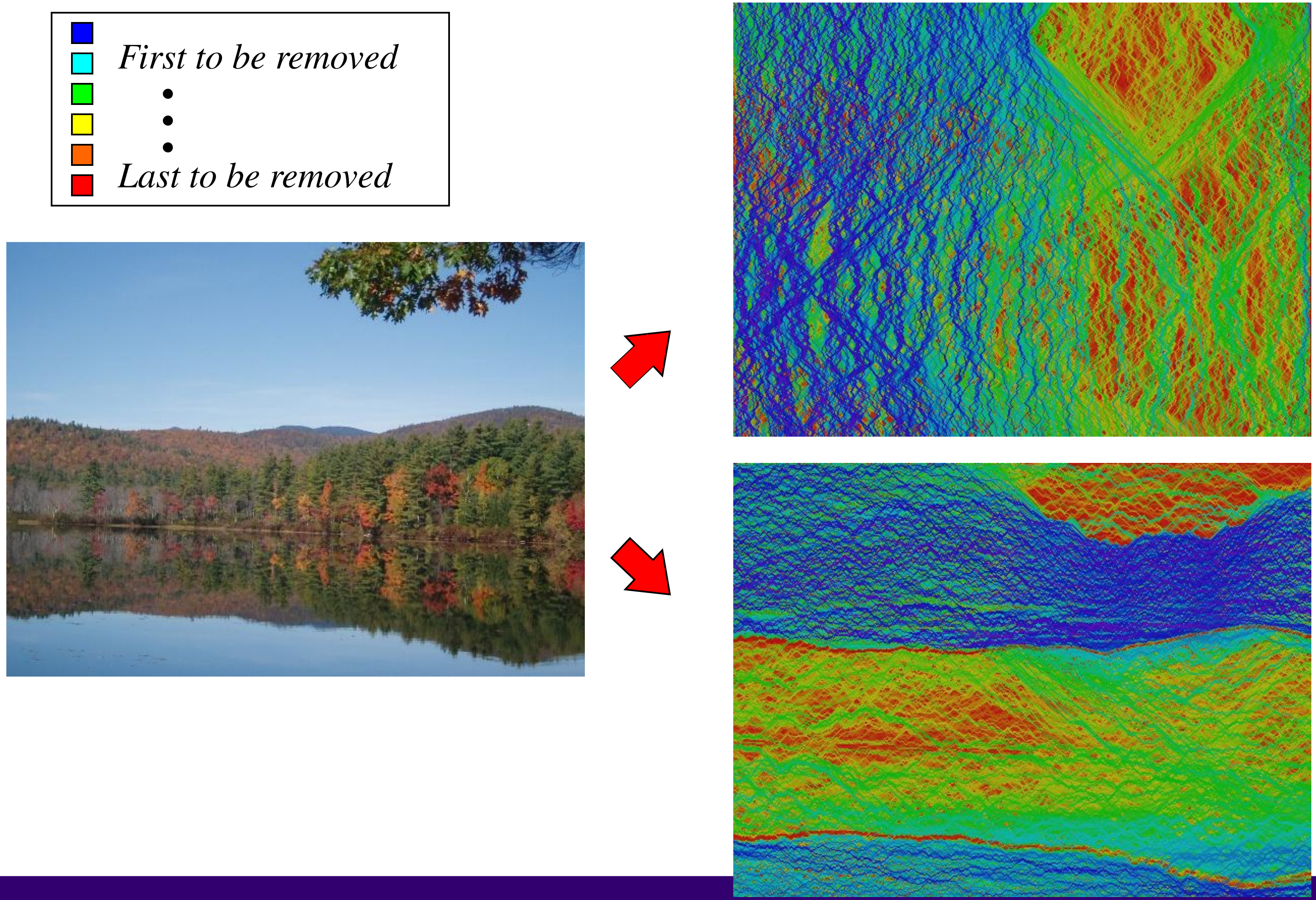
Scaling

Multi-Size Images

- We can create a new *representation* of an image that will allow adapting it to different sizes!

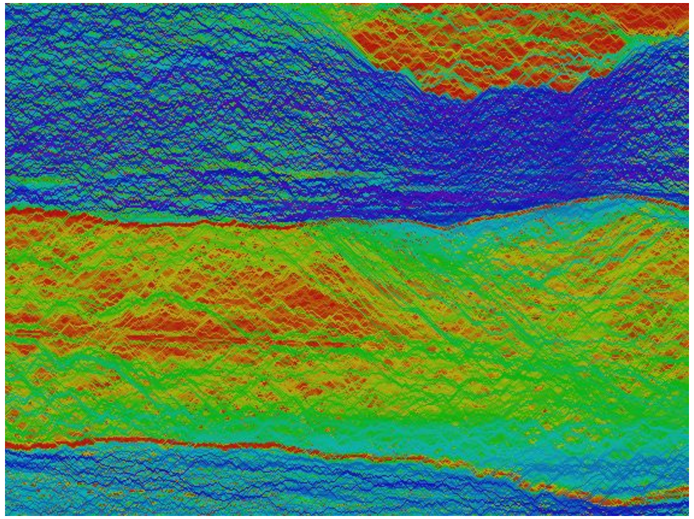
1. Precompute all seams once
2. Realtime resizing / transmit with content

Multi-Size Images

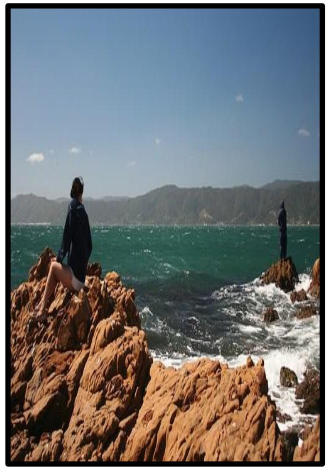


Multi-Size Image Representation

+



Multi-Size Image Representation



Content Enhancement

Q. How would you use seam carving to do this?

Replace E(i, j) with user defined energies

Recall our seam equation

$$\mathbf{M}(i,j) = E(i,j) + \min\left(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\right)$$

Set E(i, j) to be infinity is a user wants to keep this pixel
Set E(i, j) to be negative number if a user wants to get rid of it.

Object Removal

Object Removal

Input Retargeted Pigeon Removed Girl Removed

Find the Missing Shoe!

Solution

Use face detector to set energies of faces high

With User Constraints

Scaling Retargeting Retargeting

Today's agenda

- Image retargeting
- Seam carving
- Dynamic programming
- Applications
- Forward algorithm

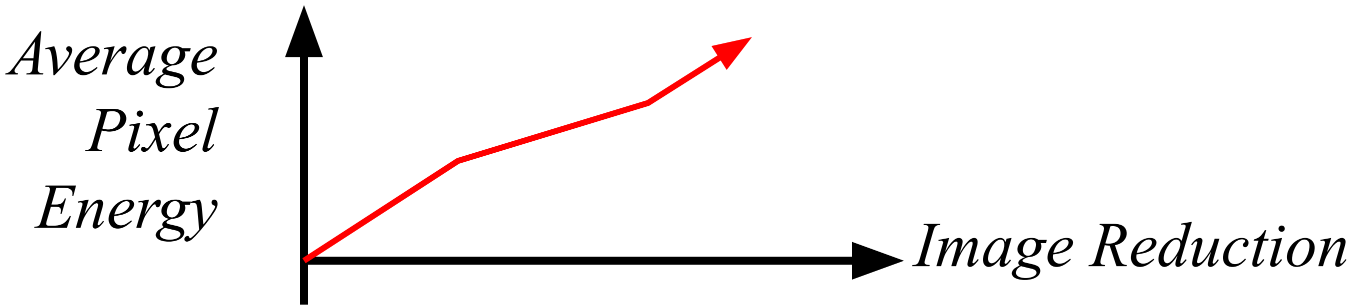
Questions

Q: What happens to the overall energy in the image during seam carving?

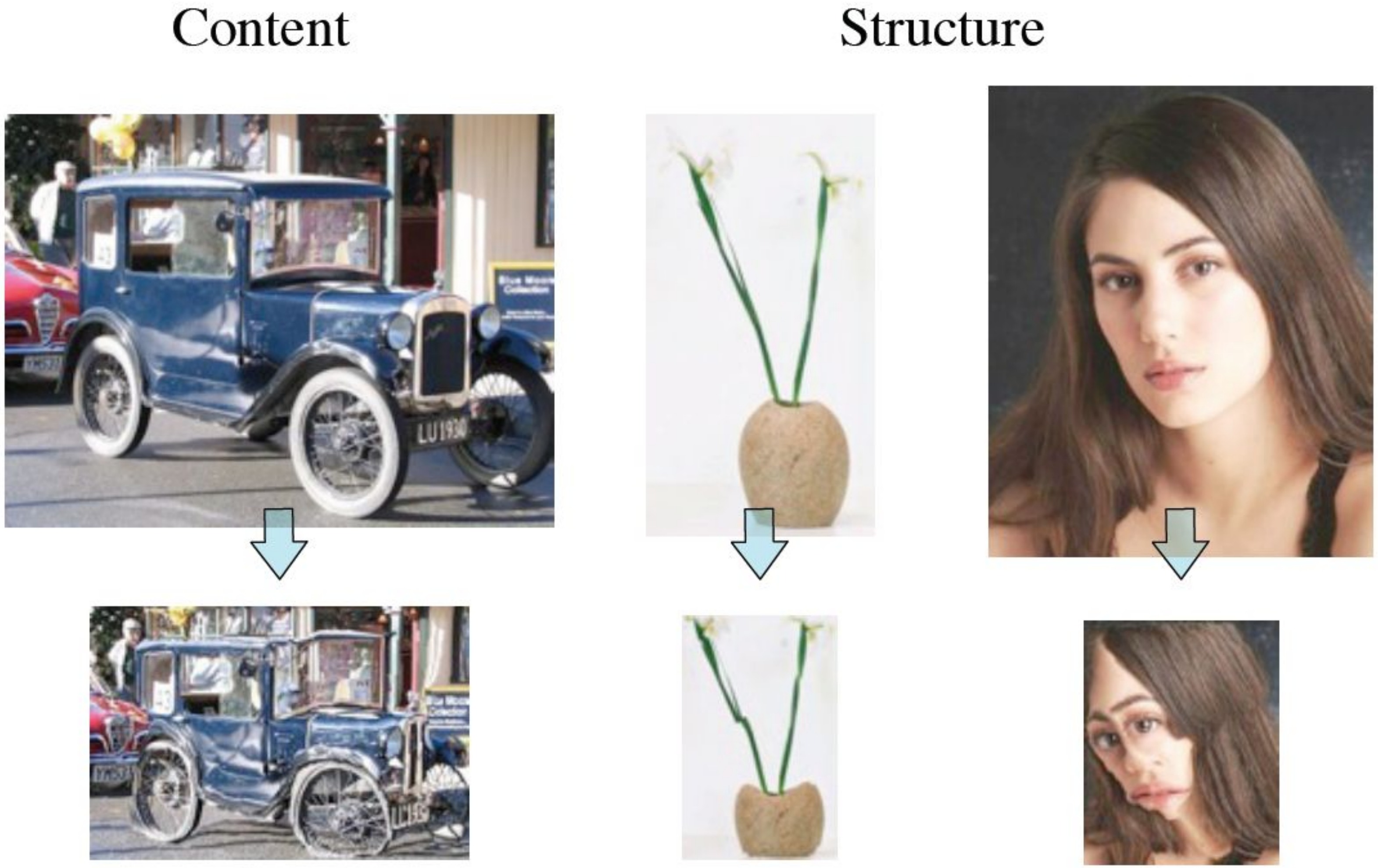
Preserved Energy

If we measure the average energy of pixels in the image after applying a resizing operator…

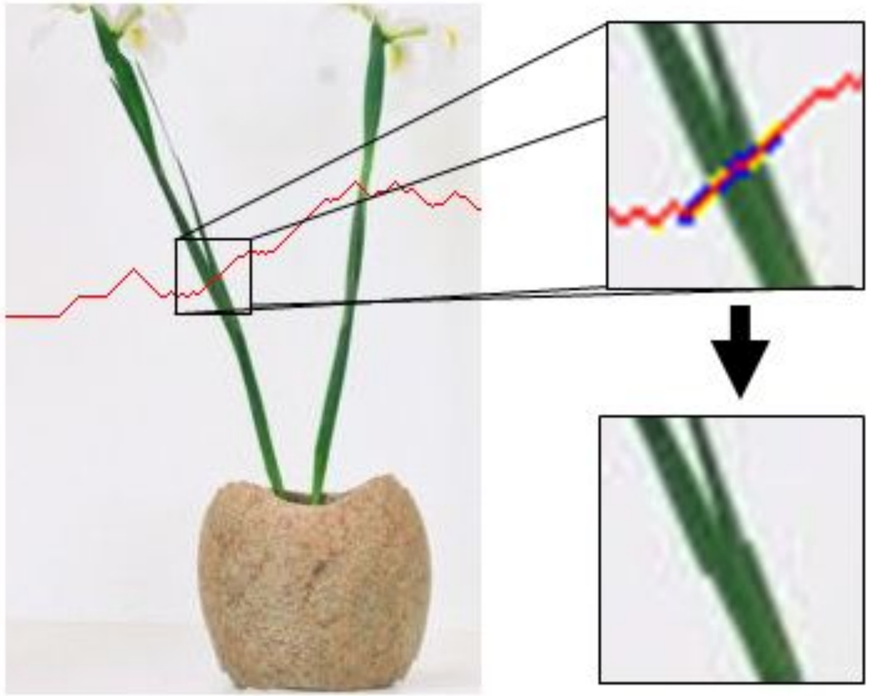
…the average should increase!

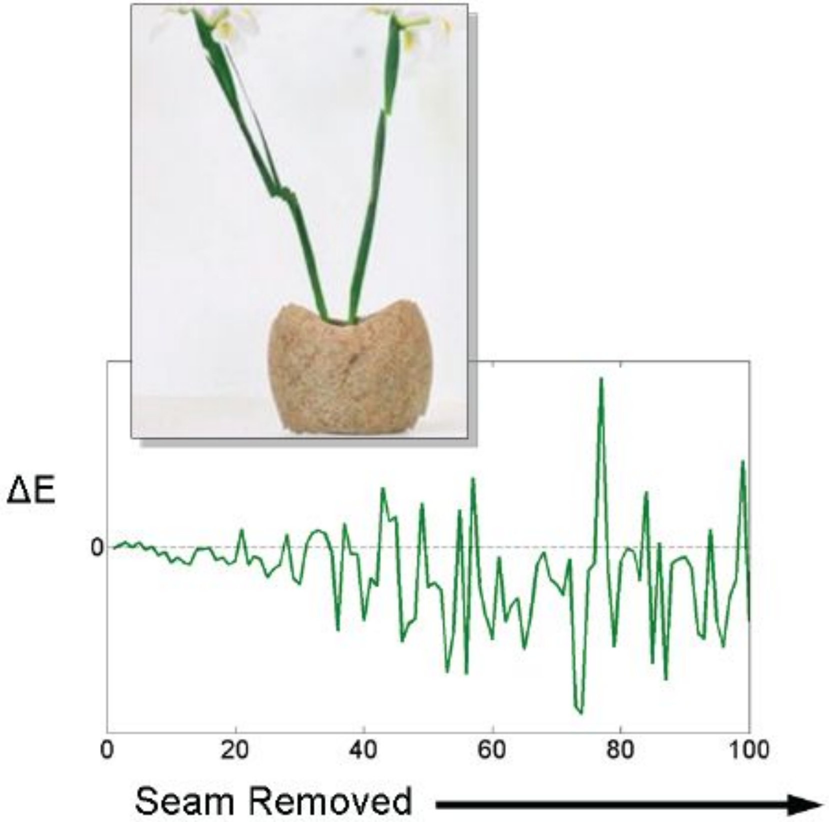


Limitations



Inserted Energy



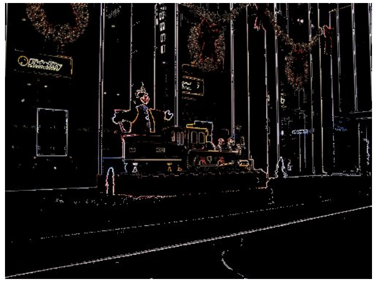


Seam carving creates artifacts breaks edges

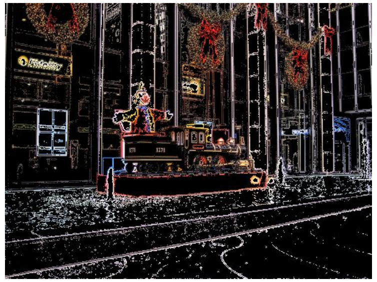
Preserved Energy

Energy

E(i, j)



15%



30%

40%

75%

M(i, j)

Energy increases after every seam removal

While resizing: remove ***as many*** low energy pixels and ***as few*** high energy pixels!

Preserved Energy

Average Pixel Energy

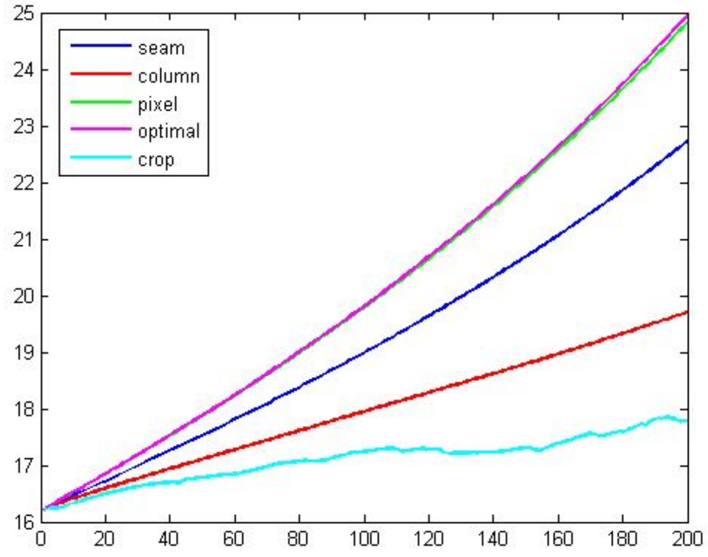


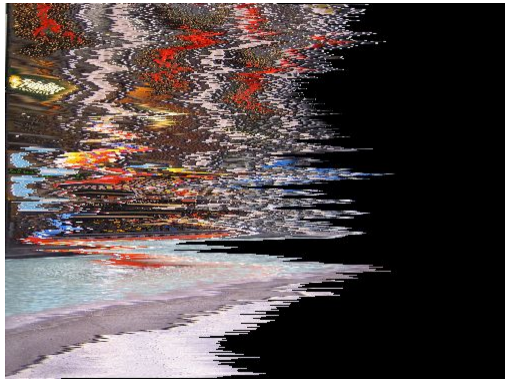
Image Reduction

crop

column

seam

pixel

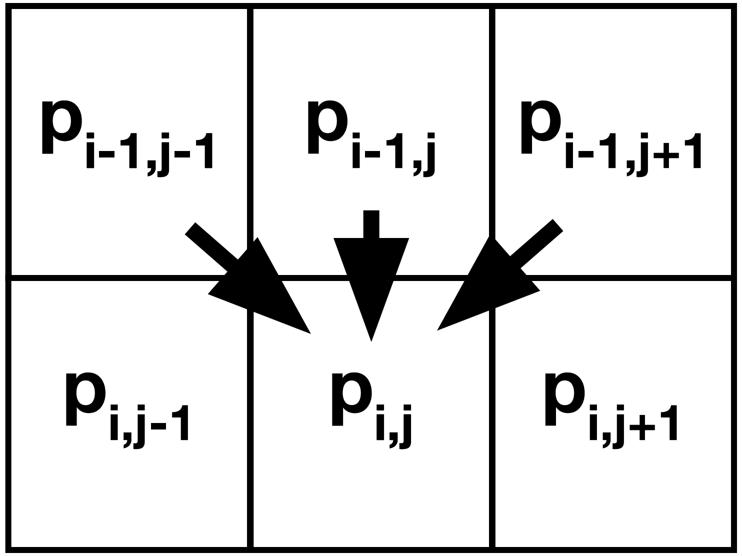


optimal

Minimize Inserted Energy

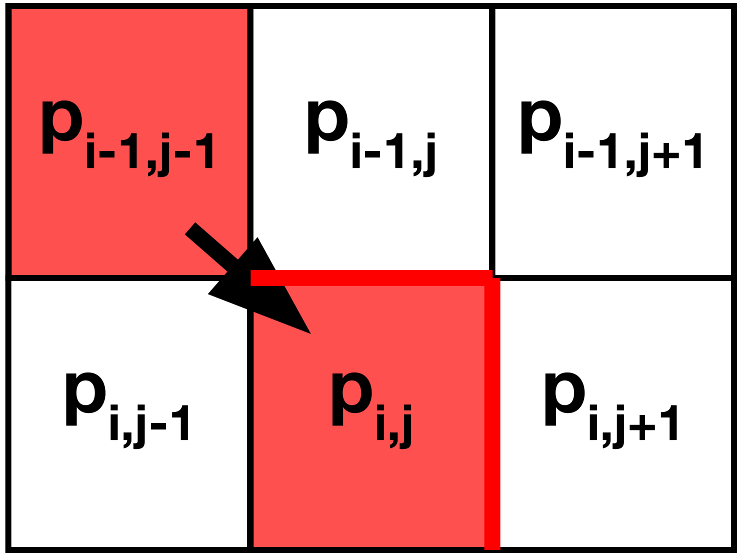
- Instead of removing the seam of least energy, remove the seam that *inserts the least energy* to the image !

Tracking Inserted Energy



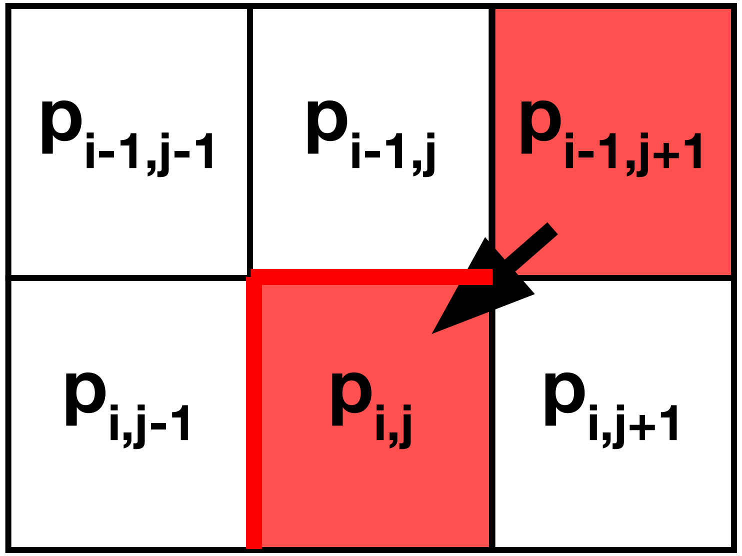
- Three possibilities when removing pixel $P_{i,j}$

Pixel $P_{i,j}$: Left Seam



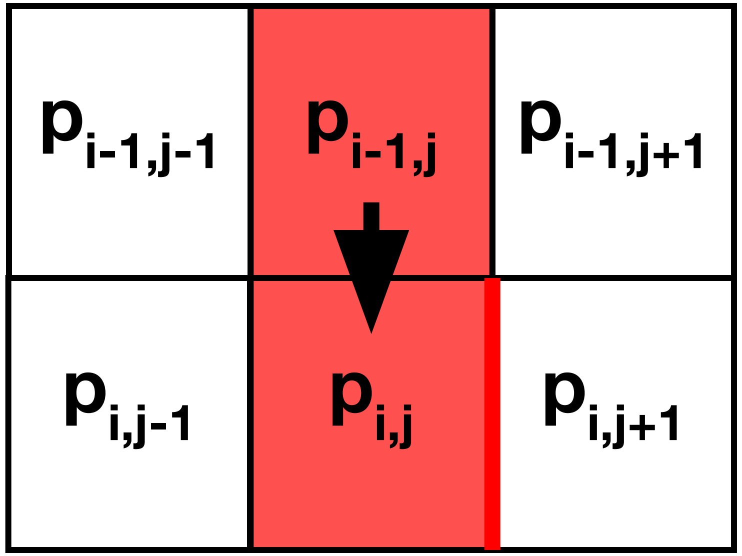
$$C_L(i,j) = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j-1)|$$

Pixel $P_{i,j}$: Right Seam



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

Pixel $P_{i,j}$: Vertical Seam



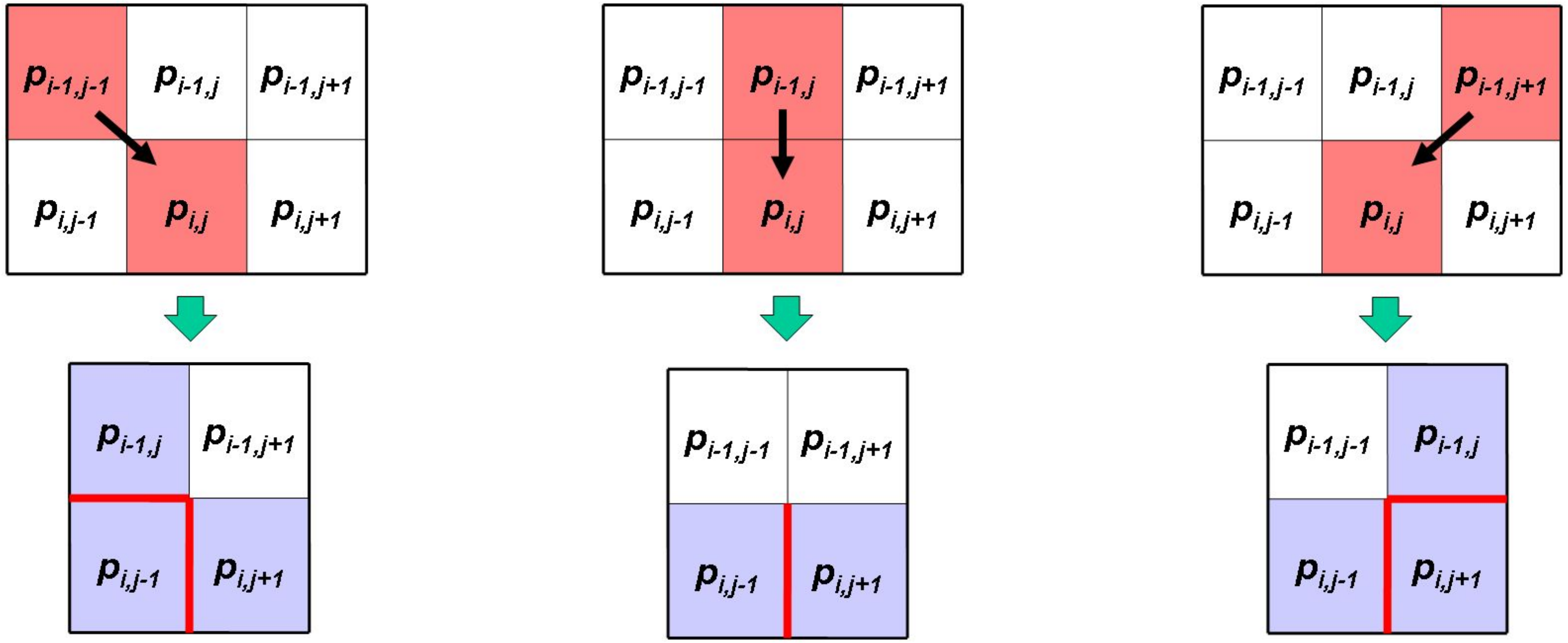
$$C_V(i,j) = |I(i,j+1) - I(i,j-1)|$$

Old Backward Cost Matrix

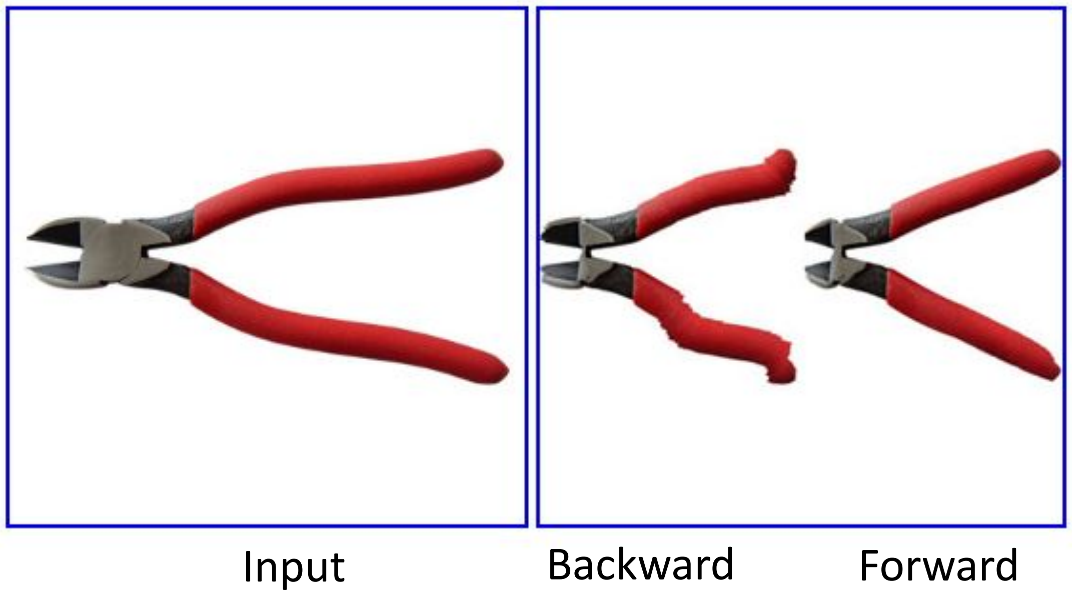
$$M(i,j) = E \qquad j) + \min \begin{cases} M(i-1, j-1) \\ M(i-1, j) \\ M(i-1, j+1) \end{cases}$$

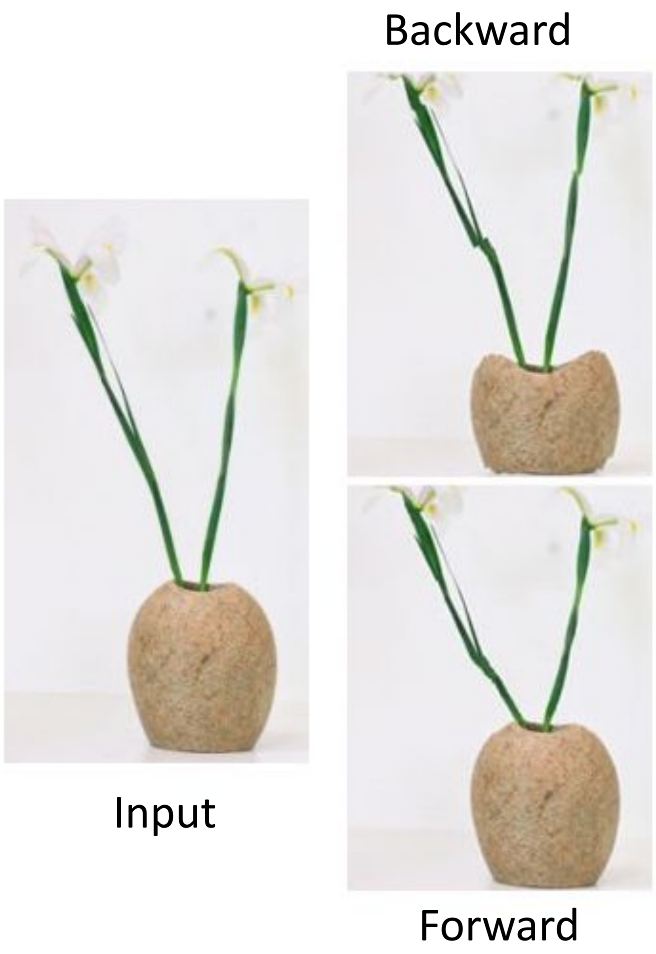
New Forward Looking Cost Matrix

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

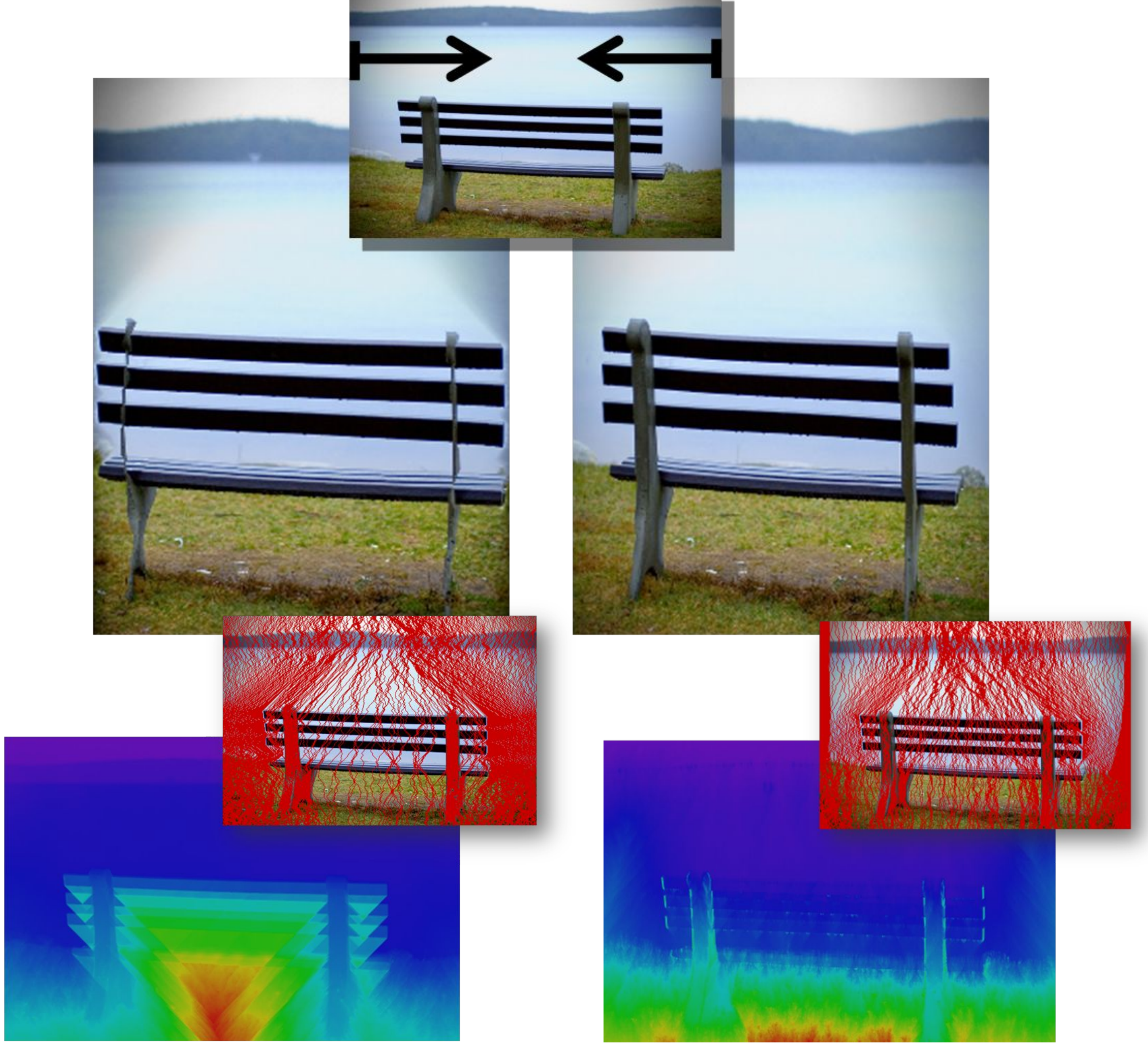


Results





Results

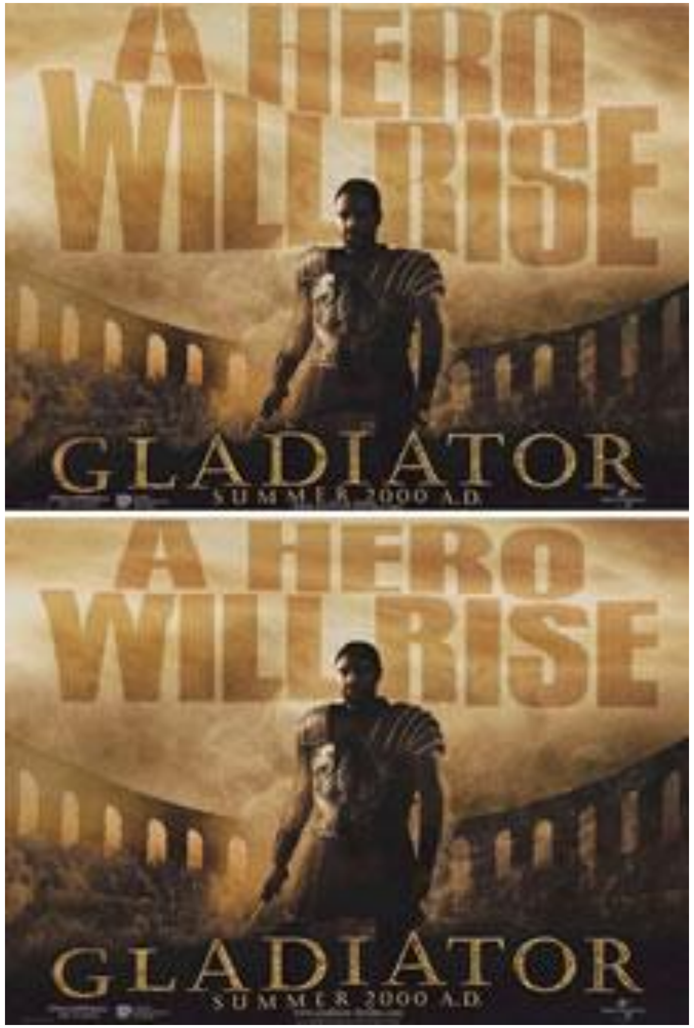


Backward vs. Forward

Backwar
d

Forward

Results


A HERO WILL RISE
GLADIATOR
SUMMER 2000 A.D.
A HERO WILL RISE
GLADIATOR
SUMMER 2000 A.D.

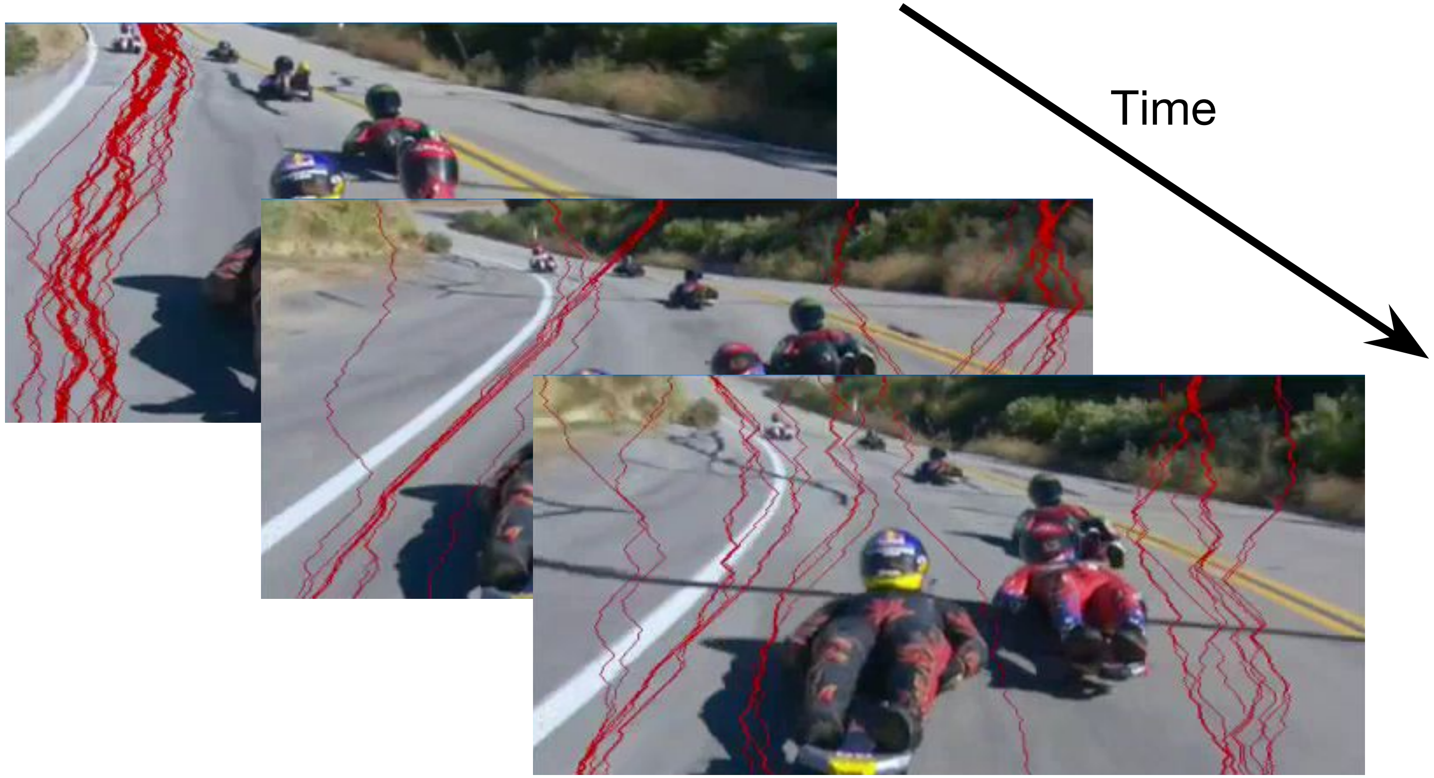
From Images to Videos

In general, video processing is a much (much!) harder problem

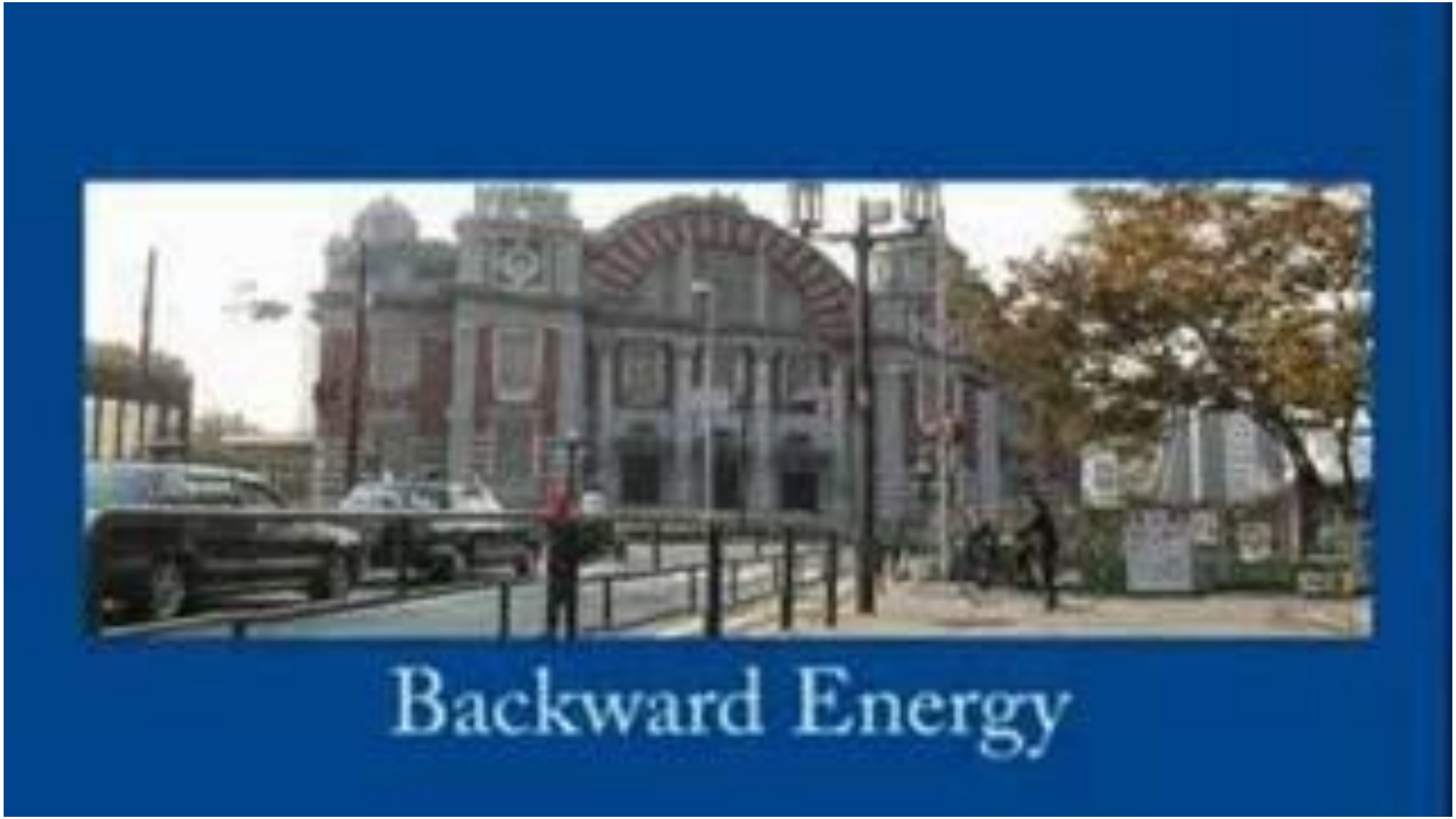
1. Cardinality
   - Suppose 1min of video x 30 fps = 1800 frames
   - Say your algorithm processes an image in 1 minute
     - 1 video would take **30 hours !!**
2. Dimensionality/algorithmic
   - Temporal coherency: human visual system is highly sensitive to motion!

Seam-Carving Video?

- Naive… frame by frame independently

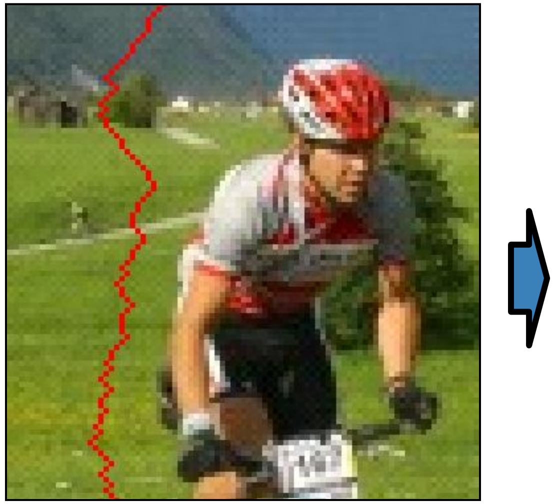


Frame-by-frame Seam-Carving

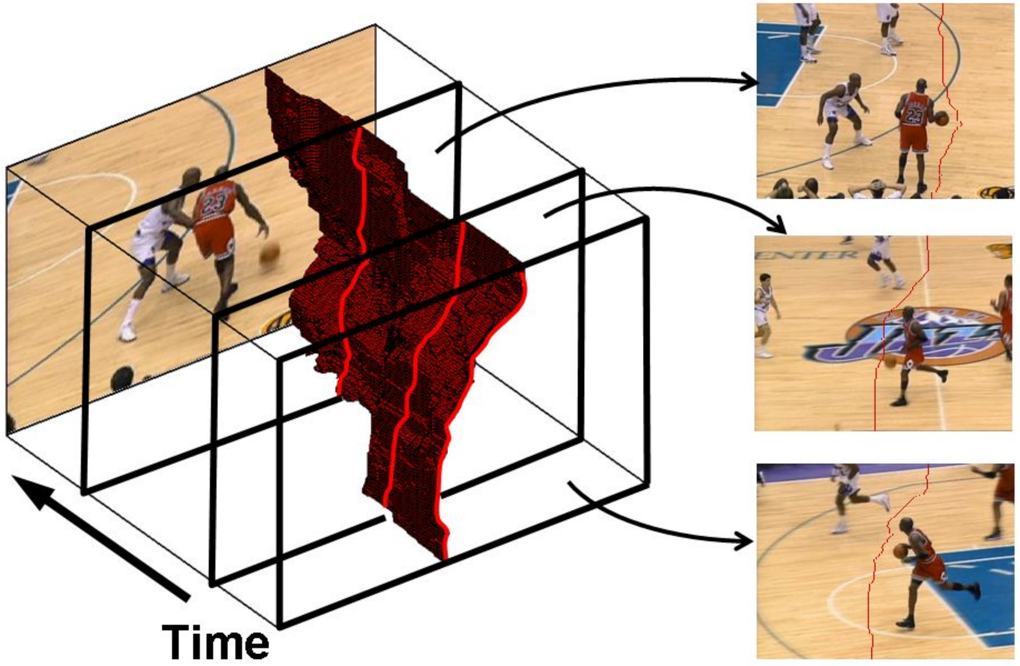


Let's check out this video

From 2D to 3D

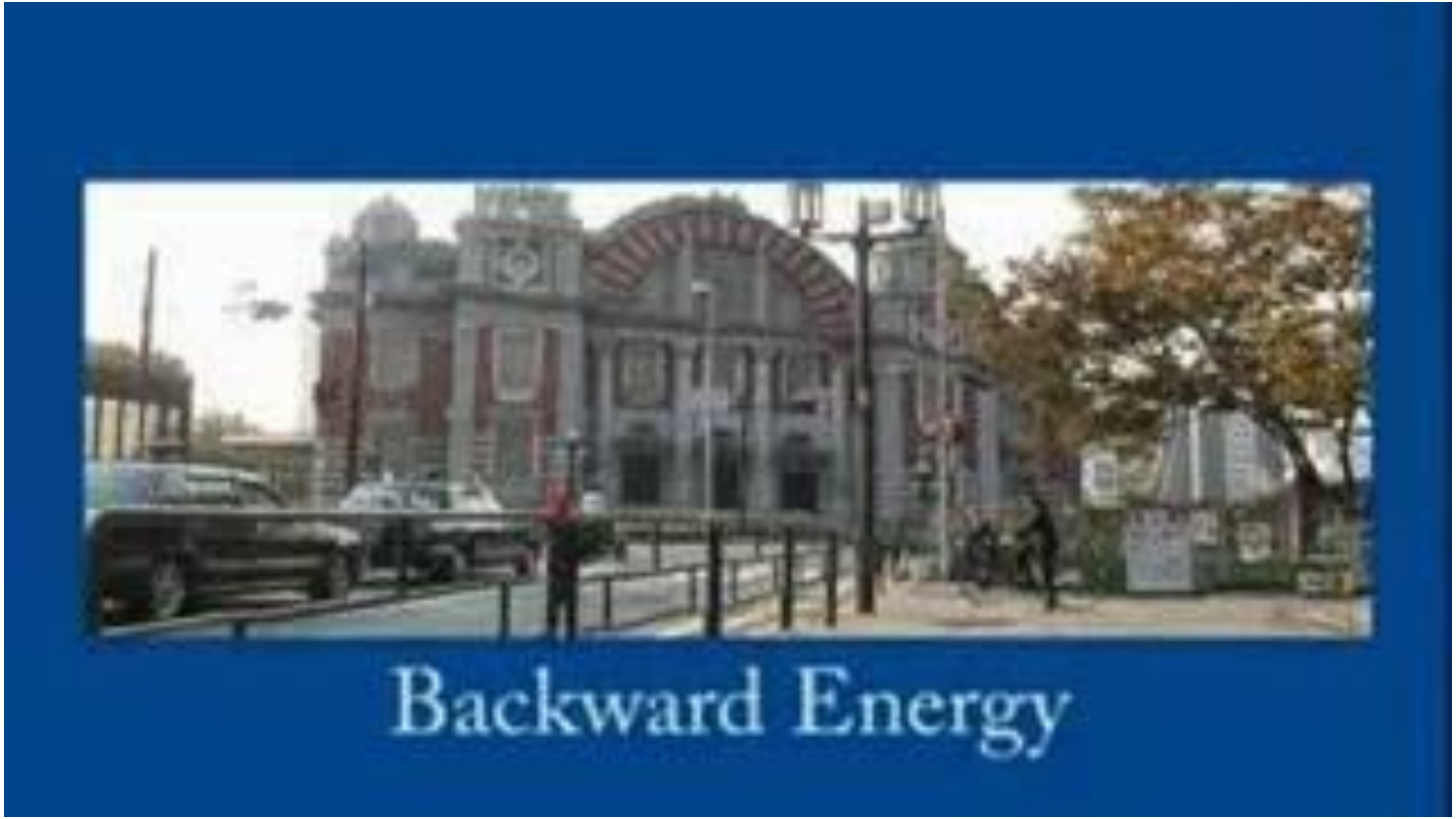


1D paths in images

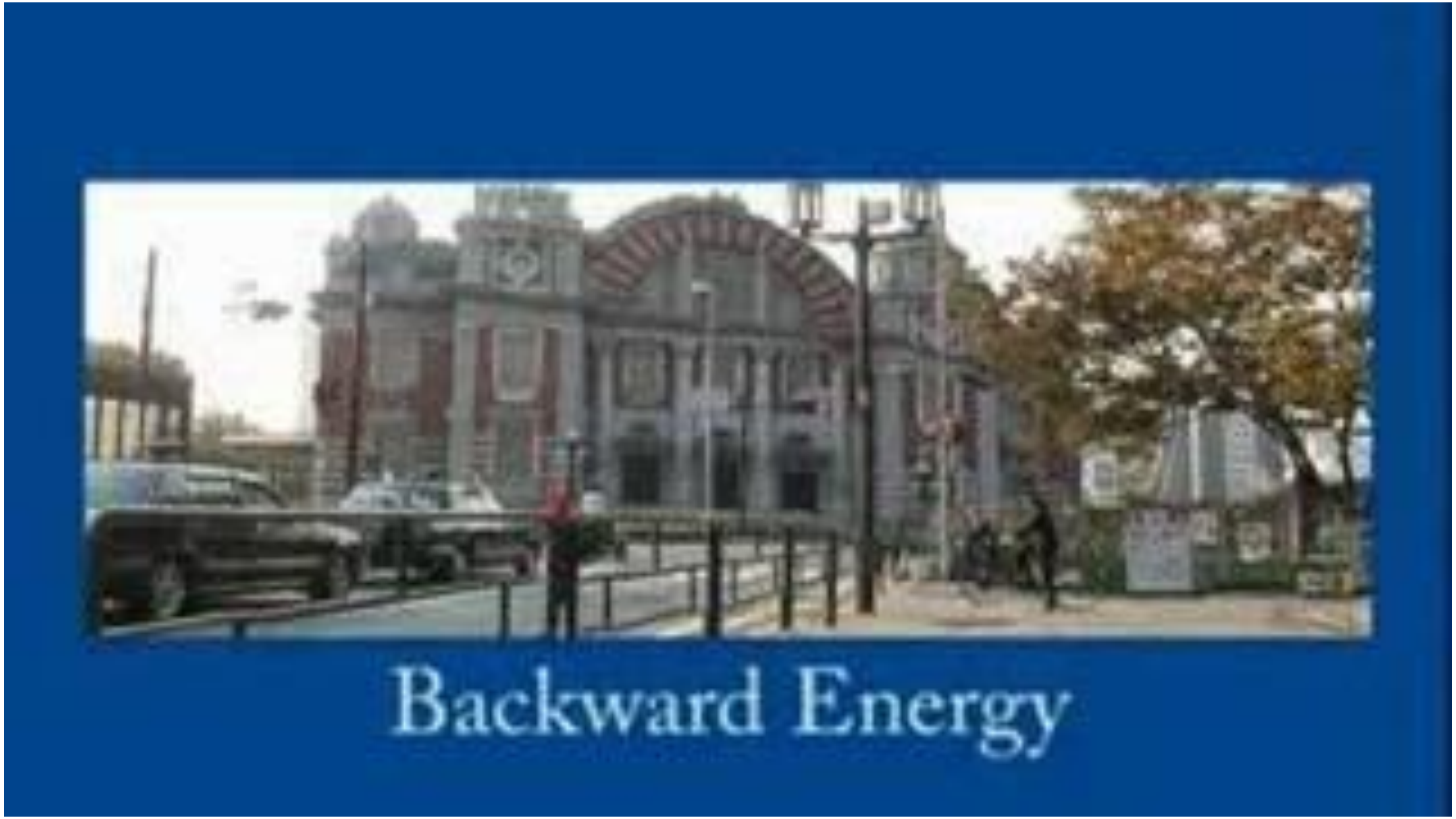


2D manifolds in video cubes

Example video retargeting



Object detection + seam carving



Today’s agenda

- Image retargeting
- Seam carving
- Dynamic programming
- Applications
- Forward algorithm

Next lecture

Segmentation and grouping

References

- Seam Carving for Content-Aware Image Resizing – Avidan and Shamir 2007
- Content-driven Video Retargeting – Wolf et al. 2007
- Improved Seam Carving for Video Retargeting – Rubinstein et al. 2008
- *Optimized Scale*-and-*Stretch* for Image Resizing – Wang et al. 2008
- Summarizing Visual Data Using Bidirectional Similarity – Simakov et al. 2008
- Multi-operator Media Retargeting – Rubinstein et al. 2009
- Shift-Map Image Editing – Pritch et al. 2009
- Energy-Based Image Deformation – Karni et al. 2009

- Seam carving in Photoshop CS4: http://help.adobe.com/en_US/Photoshop/11.0/WS6F81C45F-2AC0-4685-8FFD-DBA374BF21CD.html

A Local Operator