Reminder - Image Transformations

image filtering: change ***range*** of image

$$g(x) = T(f(x))$$

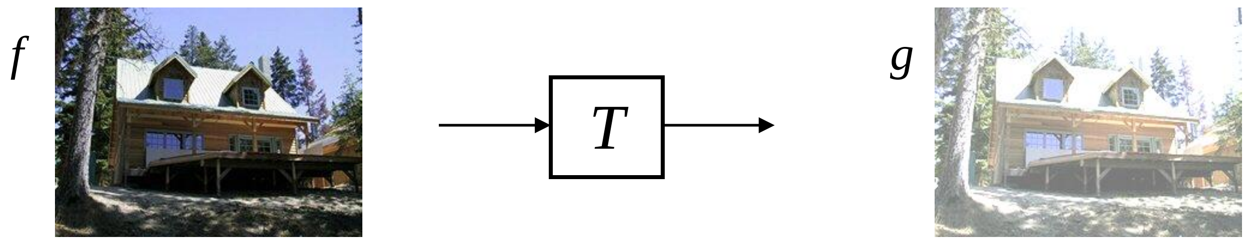
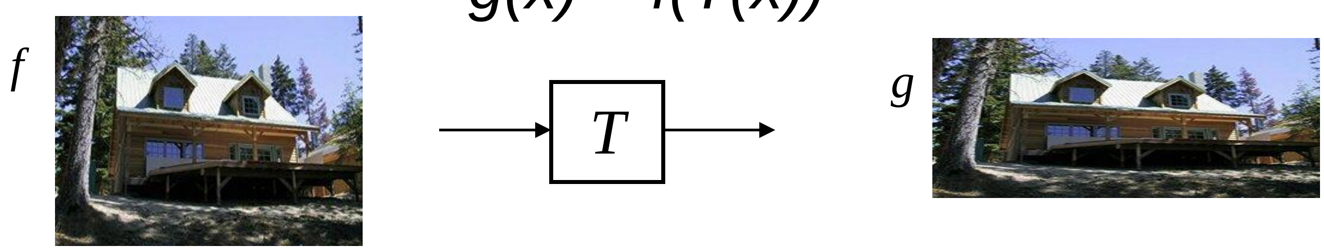


image warping: change ***domain*** of image

$$g(x) = f(T(x))$$



Reminder - Parametric (global) warping

Examples of parametric warps:

translation

rotation

aspect

affine

perspective

cylindrical

Image Retargeting

cs129: Computational Photography
James Hays, Brown, Fall 2012

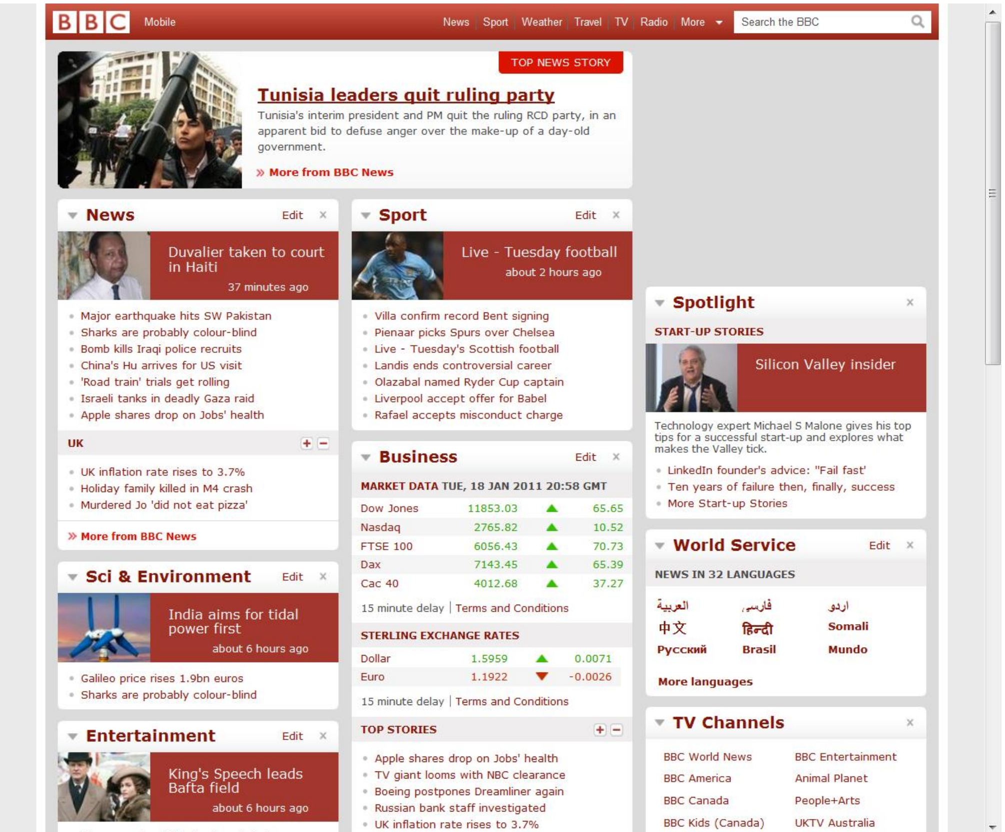
Slides by Michael Rubenstein, Ariel Shamir, and Shai Avidan

Hollywood argues retargeting

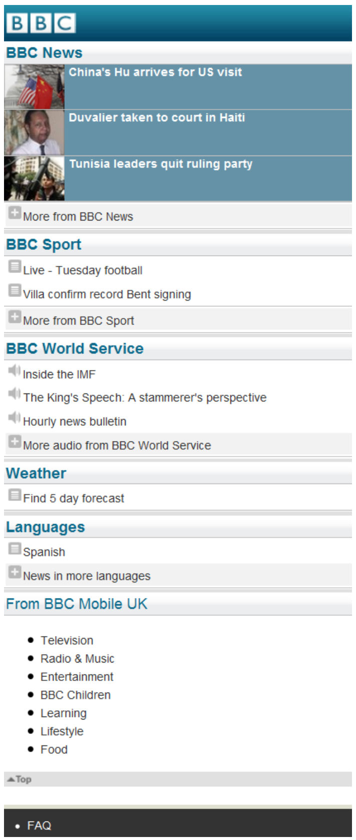
- **Widescreen vs letterbox video**

Display Devices

Content Retargeting



PC



iPhone

Page Layout

Page layout - Wikipedia, the free encyclopedia - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://en.wikipedia.org/wiki/Page_layout
Page Layout
Page layout - Wikipedia, the ...
Log in / create account
Article Discussion
Read Edit View history
Search
WIKIPEDIA
The Free Encyclopedia
Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact Wikipedia
Toolbox
Print/export
Languages
Česky
Dansk
Deutsch
Ελληνικά
Esperanto
فارسی
Français
Italiano
Nederlands
日本語
Polski
Português
Suomi
Page layout
From Wikipedia, the free encyclopedia
For the Wikipedia guideline on an articles' layout, see Wikipedia:Manual of Style (layout).
This article does not cite any references or sources.
Please help improve this article by adding citations to reliable sources. Unsourced material may be challenged and removed. (June 2007)
Page layout is the part of graphic design that deals in the arrangement and style treatment of elements (content) on a page.
Contents [hide]
1 History and development
2 Grids versus templates
3 Front-end versus back-end
4 See also
History and development [edit]
Beginning from early illuminated pages in hand-copied books of the Middle Ages and proceeding down to intricate modern magazine and catalog layouts, proper page design has long been a consideration in printed material. With print media, elements usually consist of type (text), images (pictures), and occasionally place-holder graphics for elements that are not printed with ink such as die/laser cutting, foil stamping or blind embossing.
Consumer magazine sponsored advertisements and covers rely heavily on professional page layout skills to compete for visual attention.
Since the advent of personal computing, page layout skills have expanded to electronic media as well as print media. The electronic page is better known as a graphical user interface (GUI) when interactive elements are included. Page layout for interactive media overlaps with (and is often called) interface design. This usually includes interactive elements and multimedia in addition to text and still images. Interactivity takes page layout skills from planning attraction and eye flow to the next level of planning user experience in collaboration with software engineers and creative directors.[citation needed]
A page layout may be designed in a rough paper and pencil sketch before producing, or produced during the design process to the final form. Both design and production may be achieved using hand tools or page layout software. Producing a web page may require knowledge of markup languages along with WYSIWYG editors to compensate for incompatibility between platforms. Special considerations must be made for how the layout of an HTML page will change (reflow) when resized by the end-user. Cascading style sheets are often required to keep the page layout consistent between web browsers.
Grids versus templates [edit]
Grids and templates are page layout design patterns used in advertising campaigns and multiple-page publications, including websites.
Done
F110%

Simple Media Retargeting Operators

Letterboxing
Scaling

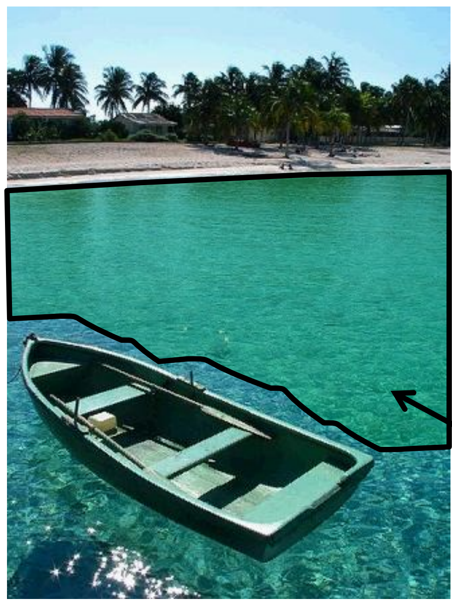
Content-aware Retargeting Operators

Content-
aware

"Important"
content

Content-
oblivious

Content-aware Retargeting



Input

“less-Important” content

Scale

Crop

Content-aware

Image Retargeting

- **Problem statement:**
  - Input Image I nxm, and new size n'xm'
  - Output Image I' of size n'xm' which will be "good representative" of the original image I

- **To date, no agreed definition, or measure, as to what a good representative is in this context!**

Image/Video Retargeting

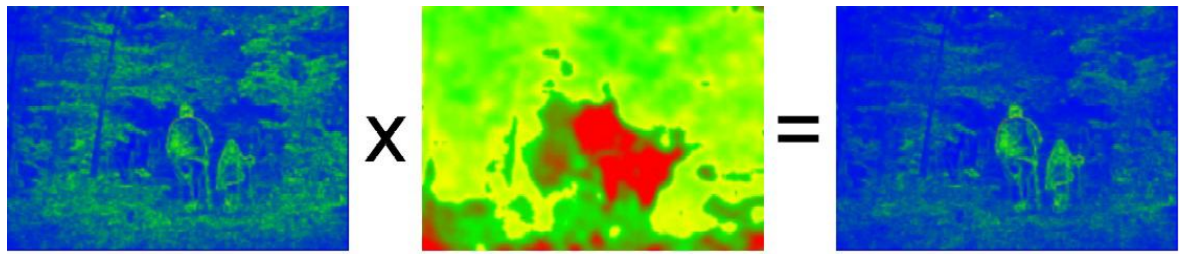
- **In large, we would expect:**

1. Adhere to the **geometric constraints** (display/aspect ratio)
2. Preserve the **important *content*** and *structures*
3. **Limit *artifacts***
4. **Perhaps a new representation that will support different sizes?**

- **Very Ill-posed!**
  - How do we define important? Is there a universal ground truth?
  - Would different viewers think the same about a retargeted image?
  - What about artistic impression in the original content?

Importance (Saliency) Measures

- **A function S: p → [0,1]**



Wang et al. 2008

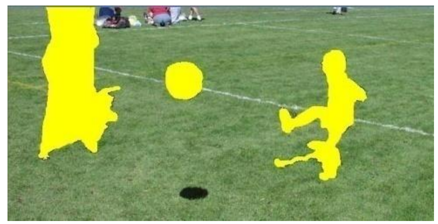
- **More sophisticated: attention models, eye tracking (gazing studies), face detectors, …**

Judd et al. ICCV09 *Learning to predict where people look*

General Retargeting Framework

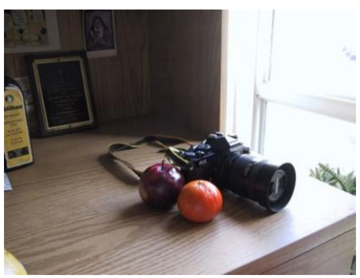
1. Define an energy function **E(I)** (interest, importance, saliency)

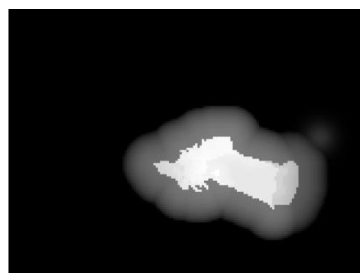
2. Use some operator(s) to change the image **I**



Recompose

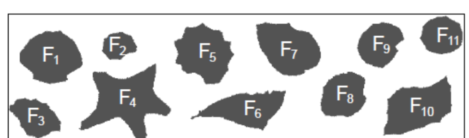
Setlur et al. [2005]





Crop

Santella et al. [2005]



Warp

Gal et al. [2006]

Previous Retargeting Approaches

- **Optimal Cropping Window**

- **For videos: “Pan and scan”**

  Still done manually in the movie industry

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

Michael Rubinstein — MIT CSAIL – mrub@mit.edu

Cropping

Seam Carving

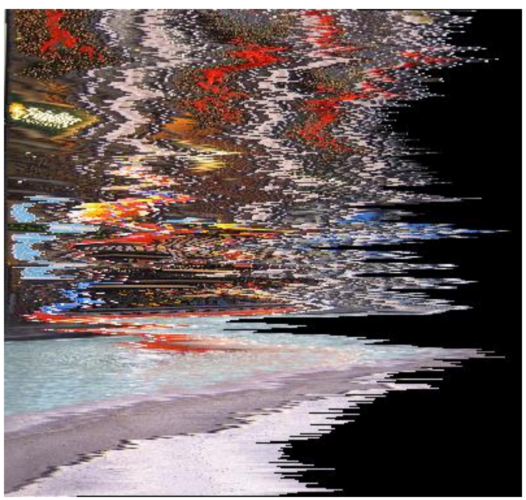
- **Assume m x n → m x n', n'<n (summarization)**

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

$$E_1(\mathbf{I}) = |\frac{\partial}{\partial x}\mathbf{I}| + |\frac{\partial}{\partial y}\mathbf{I}|.$$

- **Intuition for gradient-based energy:**
  - Preserve strong contours
  - Human vision more sensitive to edges – so try remove content from smoother areas
  - Simple, enough for producing some nice results
  - See their paper for more measures they have used

Pixel Removal



Optimal

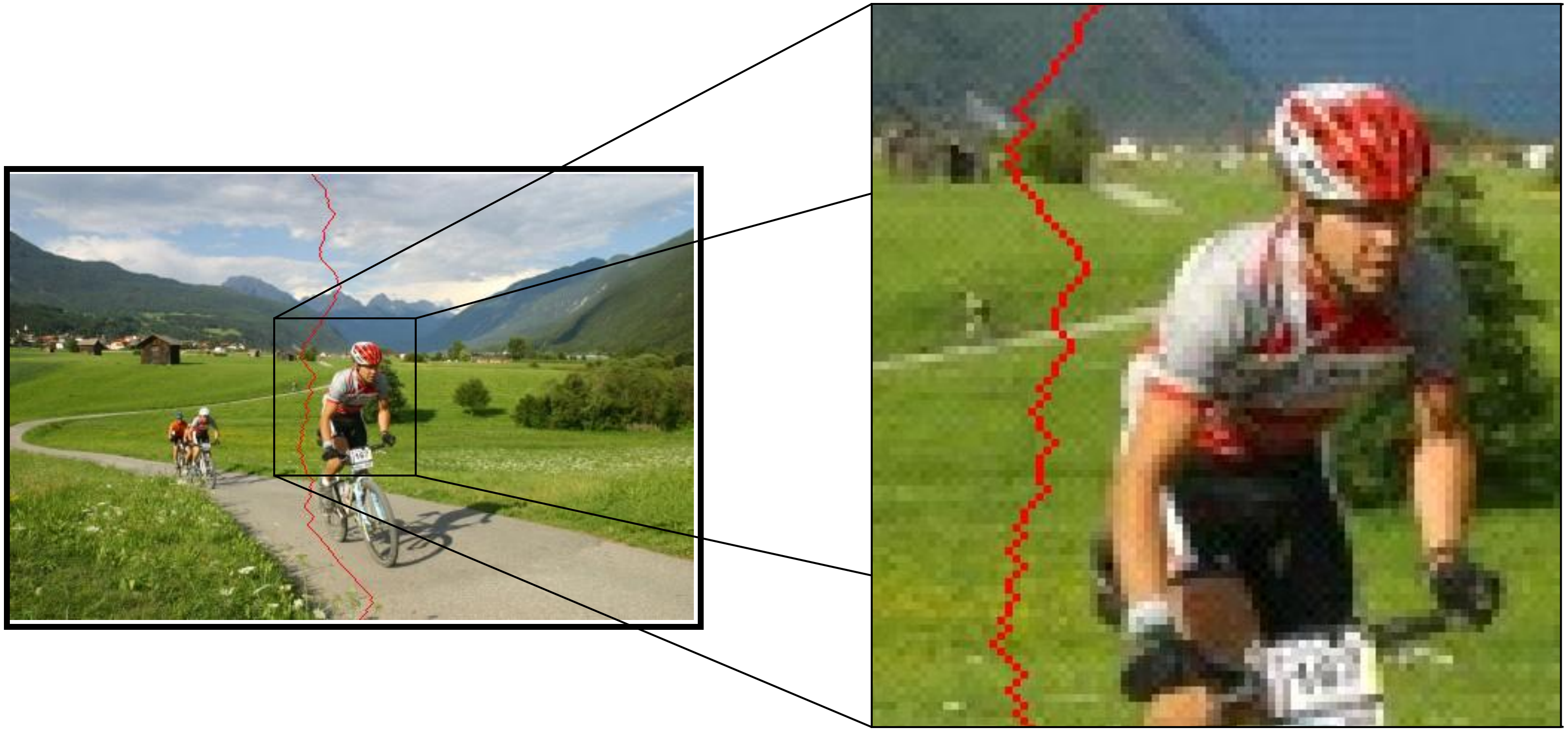
Least-energy pixels (per row)

Least-energy columns

A Seam

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

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



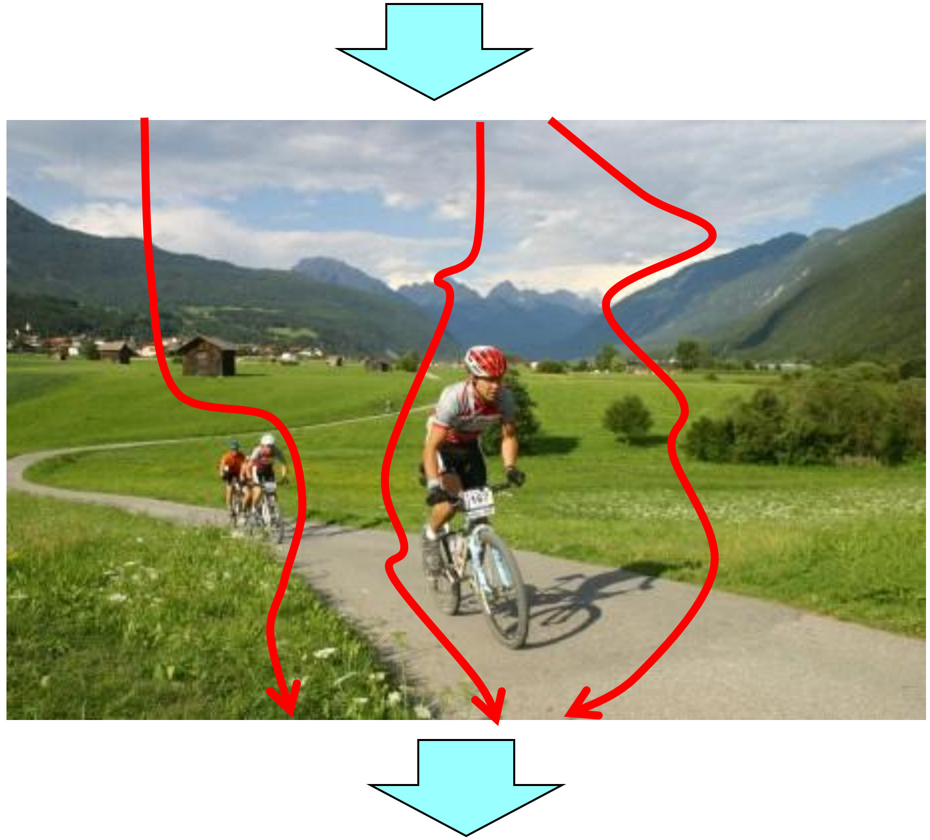
Seams in Images

- **Efros & Freeman [2001] – Texture synthesis**
- **Kwatra et al. [2003] – Image and video synthesis**
- **Agarwala et al. [2004] – Digital Photomontage**
- **Perez et**
- **Jia et al.**
- **Rother e**
- **Wang and Cohen [2006] – simultaneous matting and compositing**

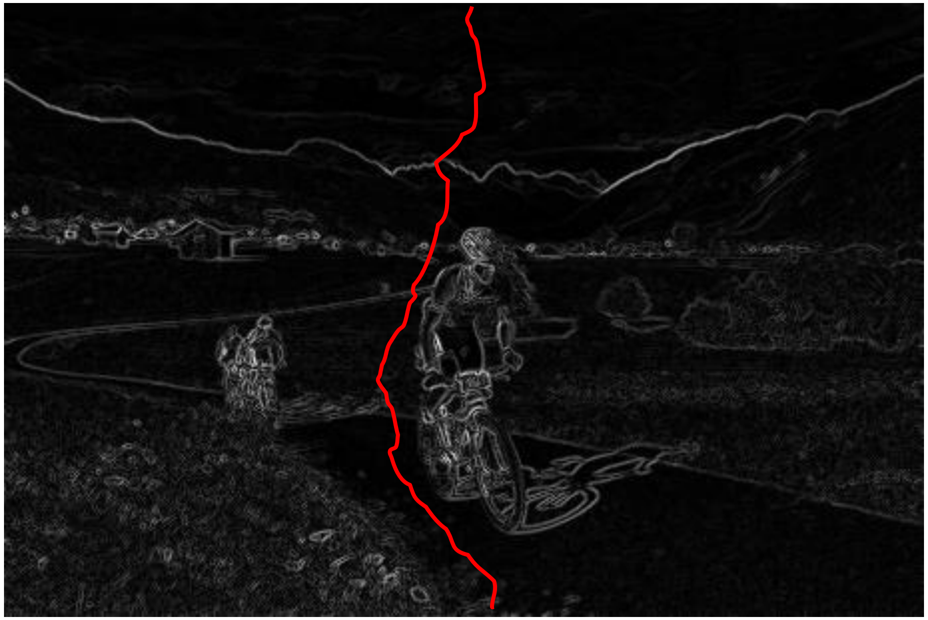
Mostly used for composition of two (or more) images or patches…

Agarwala et al., *Interactive Digital Photomontage*

Finding the Seam?



The Optimal Seam



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

The Optimal Seam

- **The recursion relation**

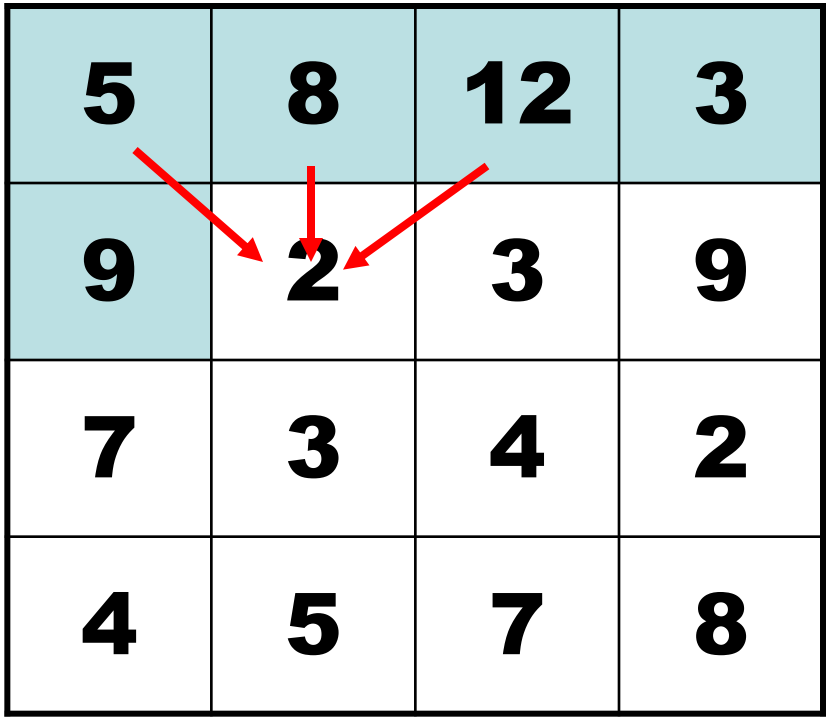
$$\mathbf{M}(i,j) = E(i,j) + \min\left(\mathbf{M}(i-1,j-1), \mathbf{M}(i-1,j), \mathbf{M}(i-1,j+1)\right)$$

- **Can be solved efficiently using dynamic programming in** $O(s \cdot n \cdot m)$

    (s=3 in the original algorithm)

Dynamic Programming

- **Invariant property:**
  – M(i,j) = minimal cost of a seam going through (i,j) (satisfying the seam properties)



Dynamic Programming

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

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

Dynamic Programming

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

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

Dynamic Programming

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

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

Michael Rubinstein — MIT CSAIL – mrub@mit.edu

Searching for Minimum

- **Backtrack (can store choices along the path, but do not have to)**

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

Backtracking the Seam

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

Backtracking the Seam

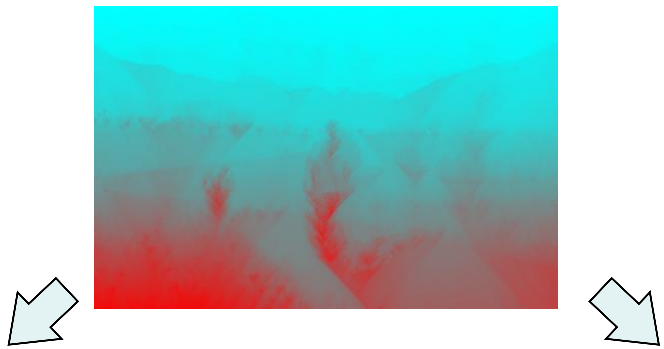
5	8	12	3
9	7	6	12
14	9	10	8
14	14	15	16

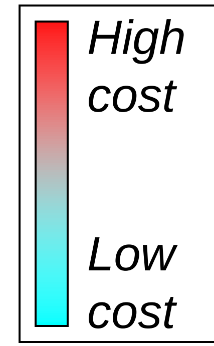
Michael Rubinstein — MIT CSAIL – mrub@mit.edu

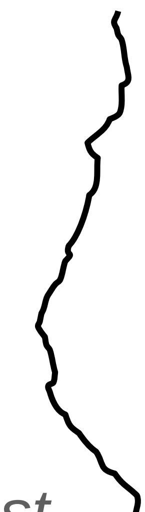
Backtracking the Seam

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

H & V Cost Maps







Horizontal Cost

Vertical Cost

Seam Carving

The Seam-Carving Algorithm

SEAM-CARVING(im,n') // size(im) = mxn

1. Do (n-n') times

  2.1. E ← Compute energy map on im

  2.2. s ← Find optimal seam in E

  2.3. im ← Remove s from im

2. Return im

- **For vertical resize: transpose the image**

- **Running time:**

  2.1 O(mn) 2.2 O(mn) 2.3 O(mn)

  ➔O(dmn)   d=n-n'

Changing Aspect Ratio

Changing Aspect Ratio

Original

Seam Carving

Scaling

Changing Aspect ratio

Cropping

Seams

Scaling

Changing Aspect Ratio

Original

Retarget

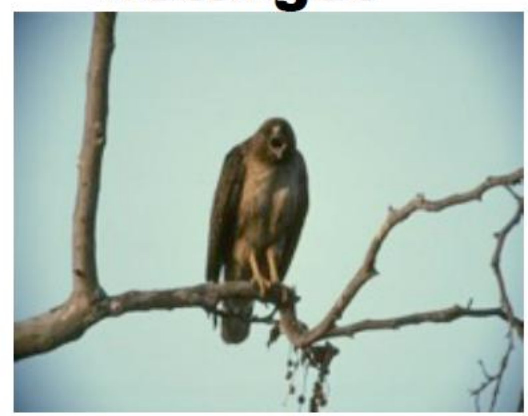
Scaling

Michael Rubinstein — MIT CSAIL – mrub@mit.edu

Changing Aspect Ratio

Original

Retarget



Scaling

Seam Carving in the Gradient Domain

Start of lecture 2 – complete aside

http://www.retronaut.co/2011/10/the-invisible-mother-1800s/

Questions?

- **Q: Will the result be the same if the image is flipped upside down?**
- **A: Yes (up to numerical stability)**

- **Q: Can we improve the running time?**
- **A: Yes. by factor (account for locality of operations)**

A Local Operator

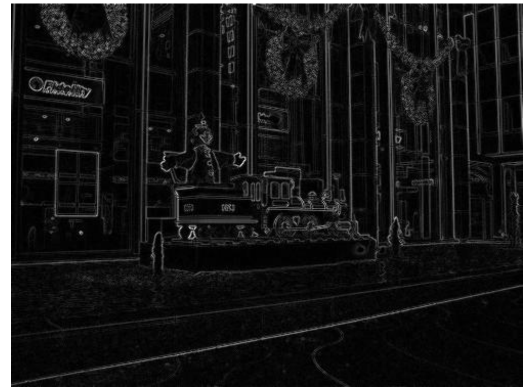
Questions?

- **Q: Will the result be the same if the image is flipped upside down?**
- **A: Yes (up to numerical stability)**

- **Q: Can we improve the running time?**
- **A: By factor (local operations)**

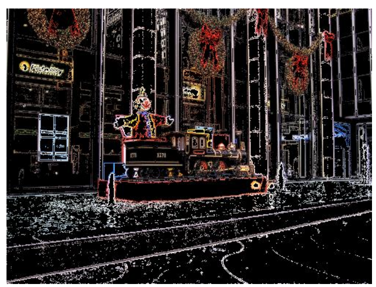
- **Q: What happens to the overall energy in the image during seam carving?**

Preserved Energy

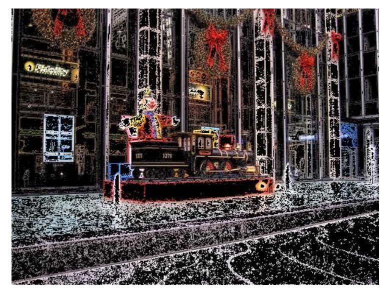


Energy

10%



30%



40%

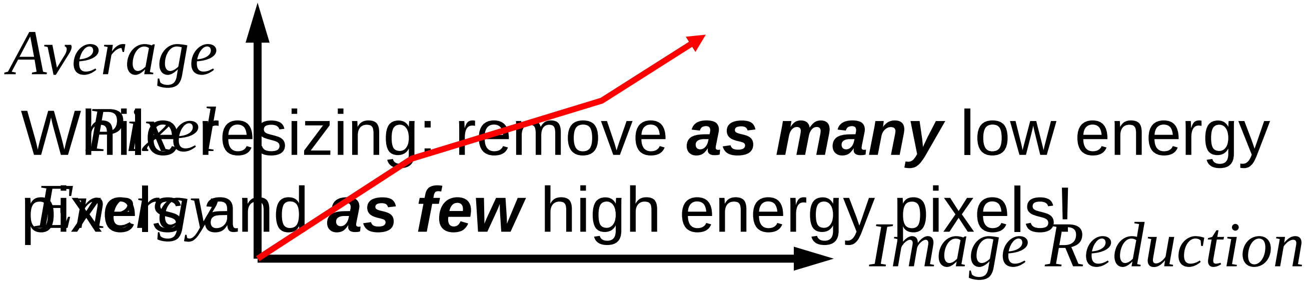
75%

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

Preserved Energy

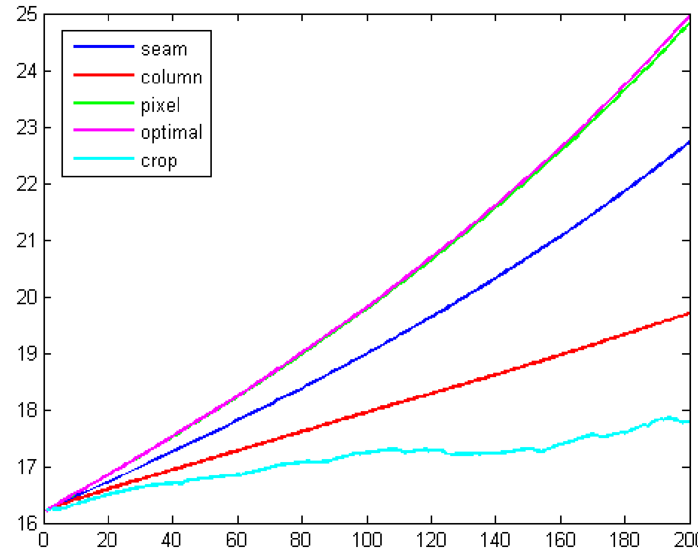
If we measure the average energy of pixels in the image after applying a resizing operator…

…the average should increase!



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

Preserved Energy

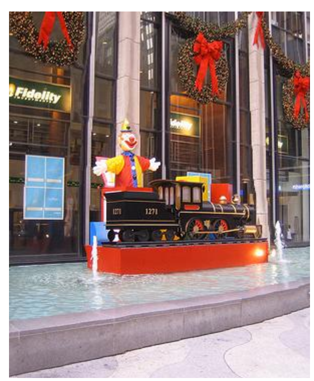


Average Pixel Energy

Image Reduction

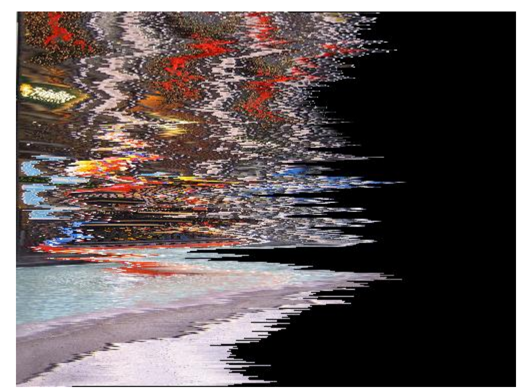
crop

column



seam

pixel



optimal

Michael Rubinstein — MIT CSAIL – mrub@mit.edu

Image Expansion (Synthesis)

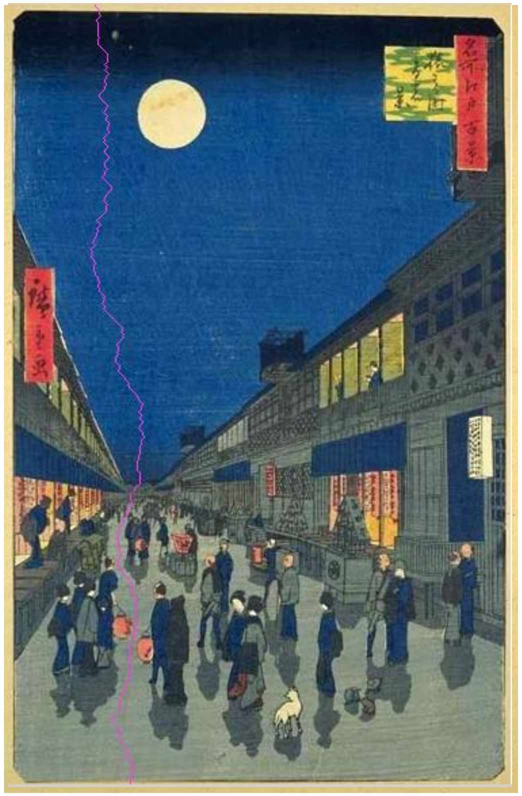
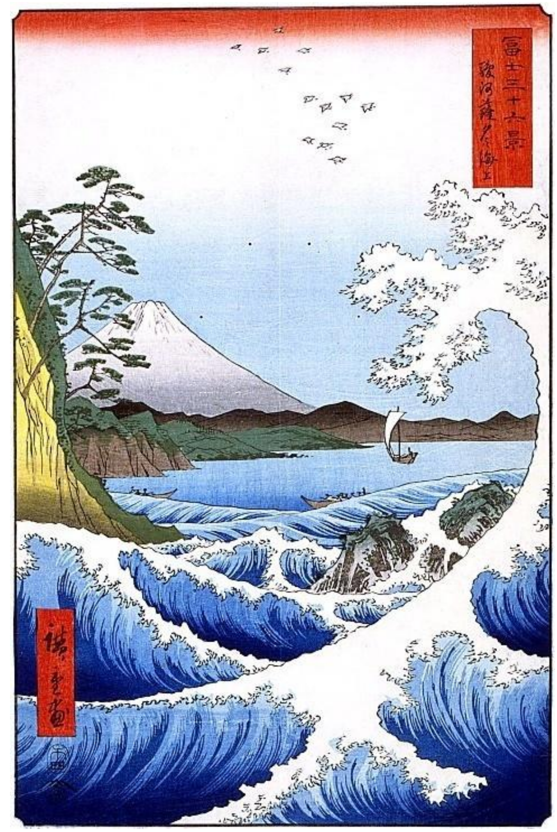


Image Expansion – take 2

Scaling

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!**

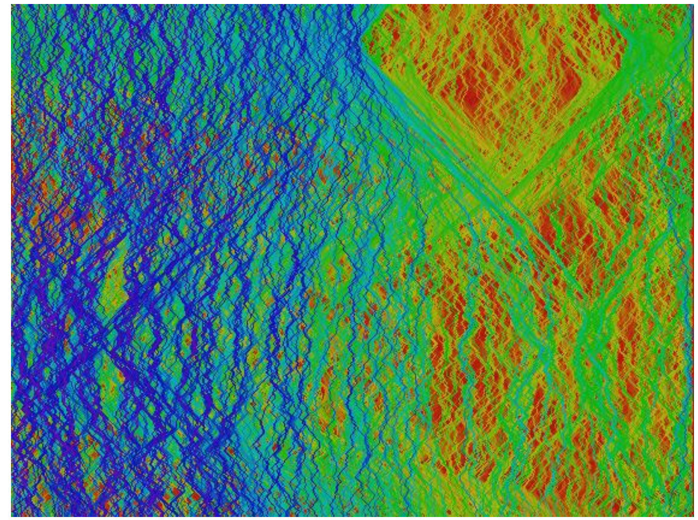
1. Precompute all seams once
2. Realtime resizing / transmit with content

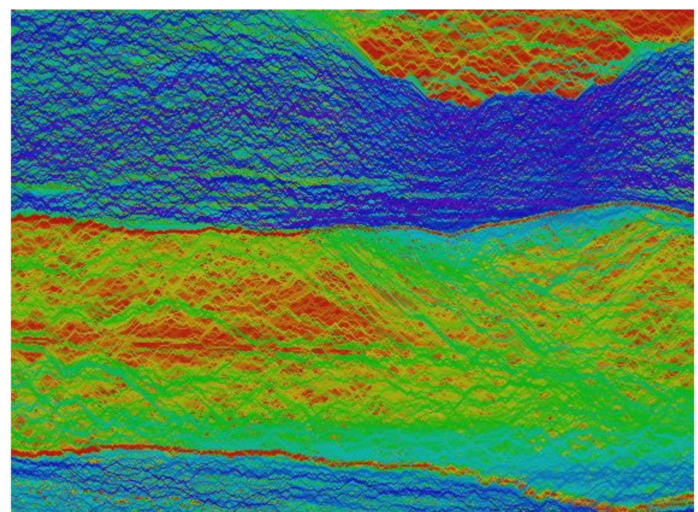
Multi-Size Images

First to be removed

•
•
•

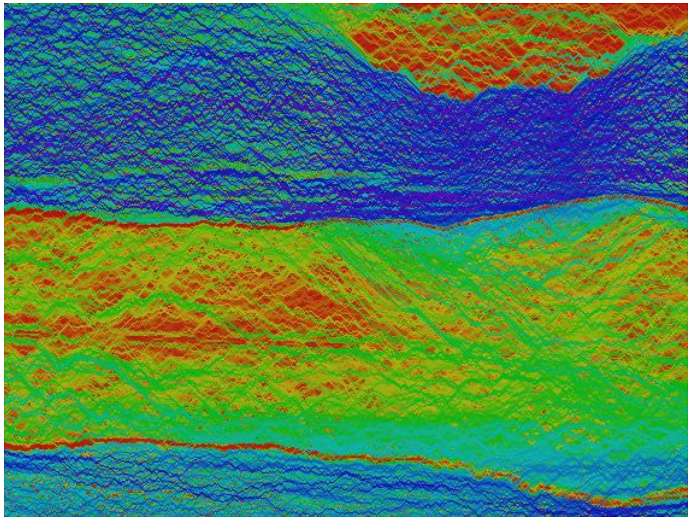
Last to be removed



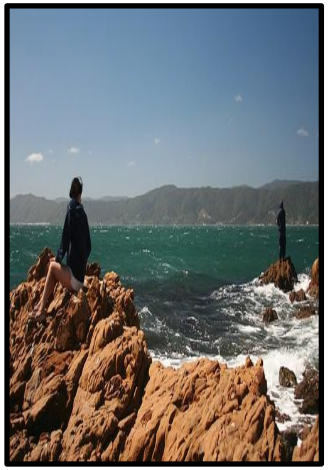


Multi-Size Image Representation

+



Multi-Size Image Representation



Auxiliary Energy

- **Recall our seam equation**

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

With face detector

With User Constraints

Object Removal

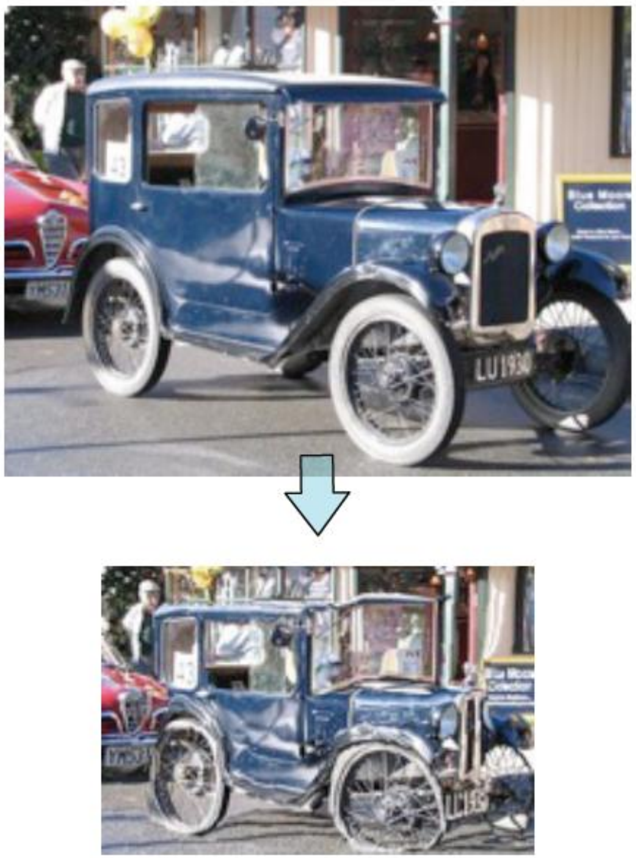
Michael Rubinstein — MIT CSAIL – mrub@mit.edu

Object Removal

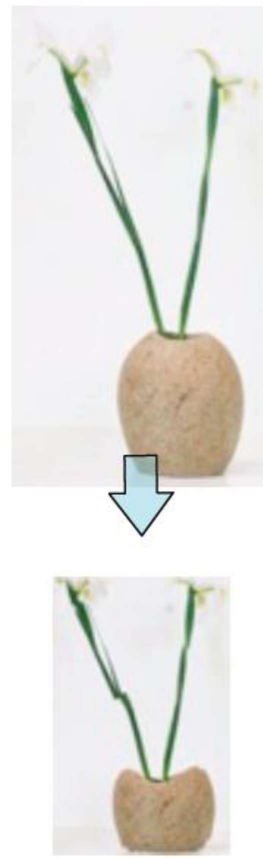
Input | Retargeted | Pigeon Removed | Girl Removed

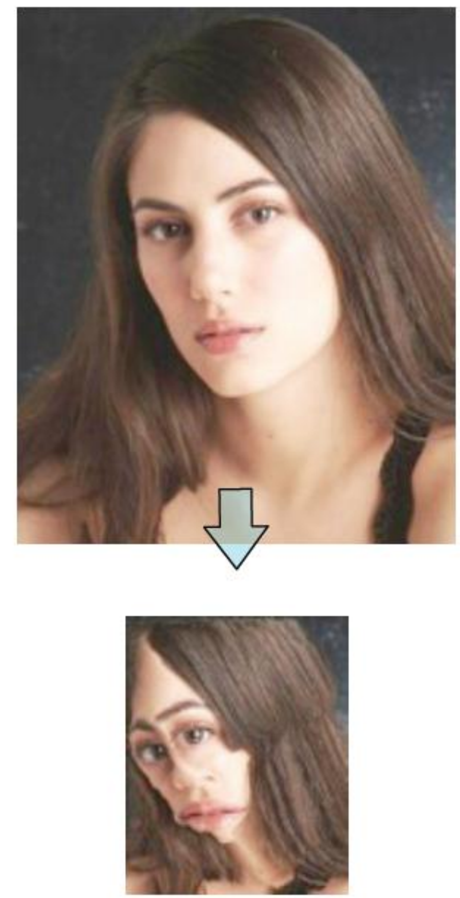
Limitations

Content



Structure





Preserved Energy - Revisited

Average Pixel Energy

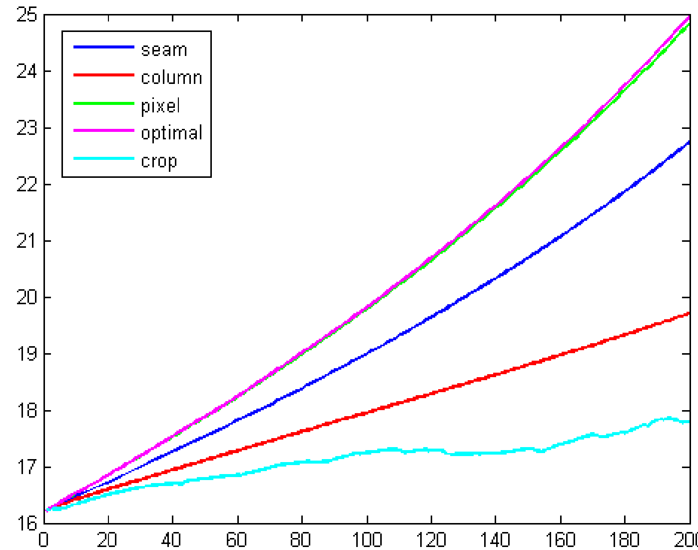


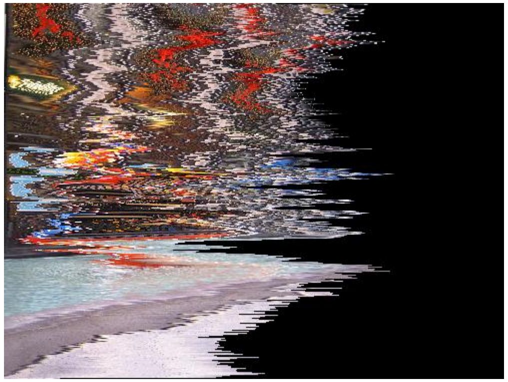
Image Reduction

crop

column

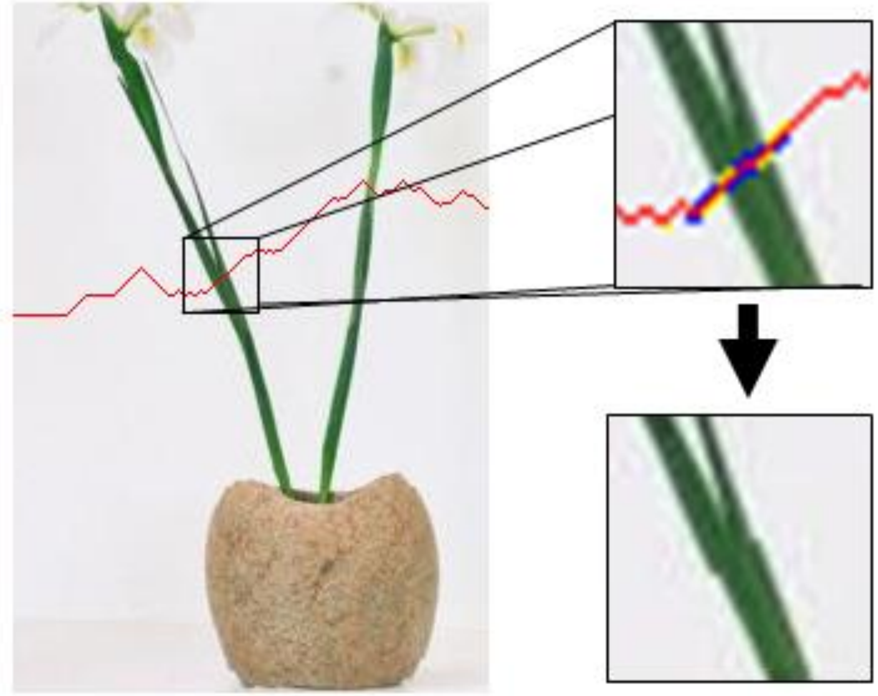
seam

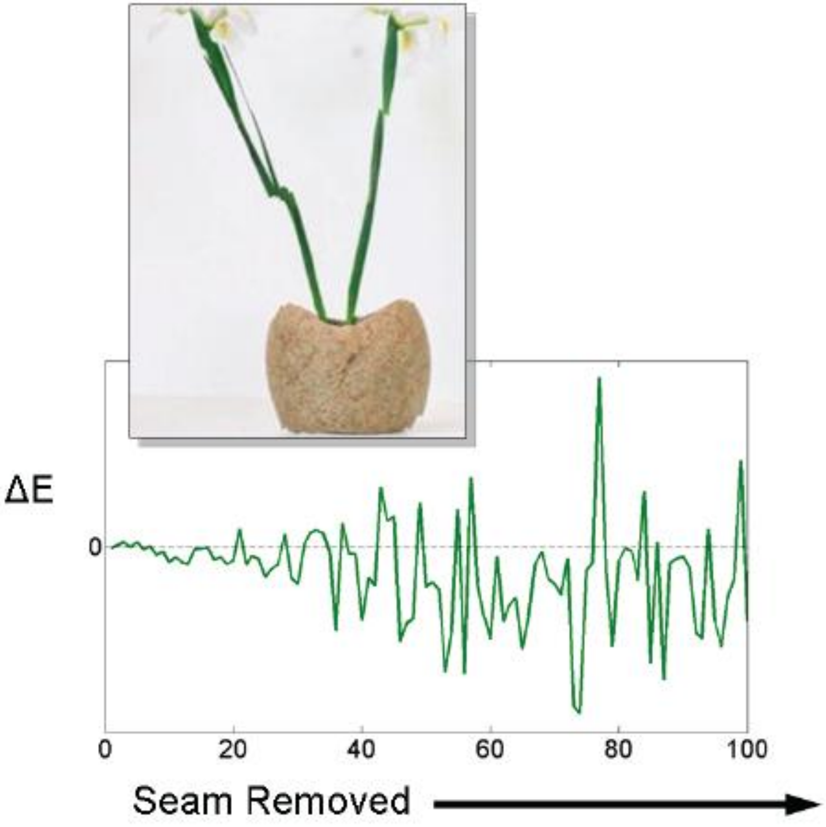
pixel



optimal

Inserted Energy

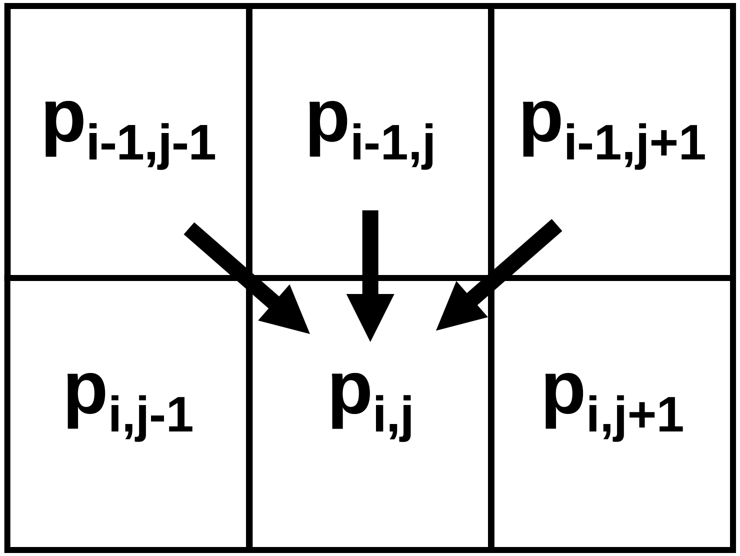




Minimize Inserted Energy

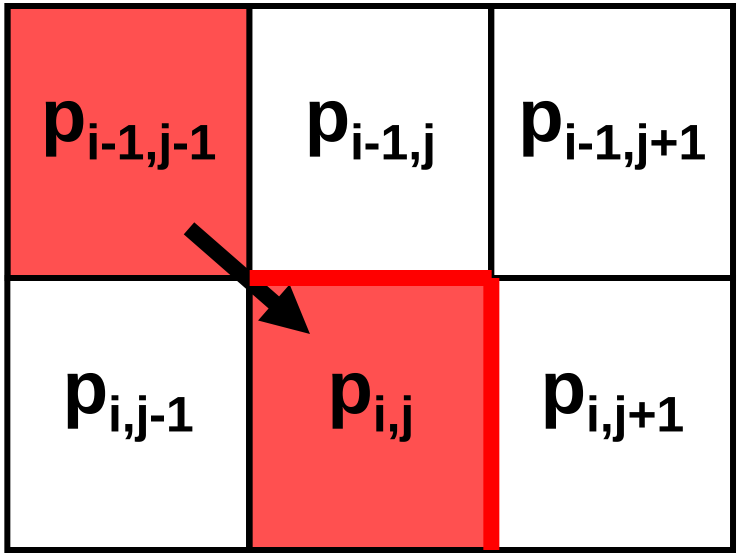
- **Instead of removing the seam of least energy, remove the seam that *inserts the least energy* to the image !**

Tracking Inserted Energy



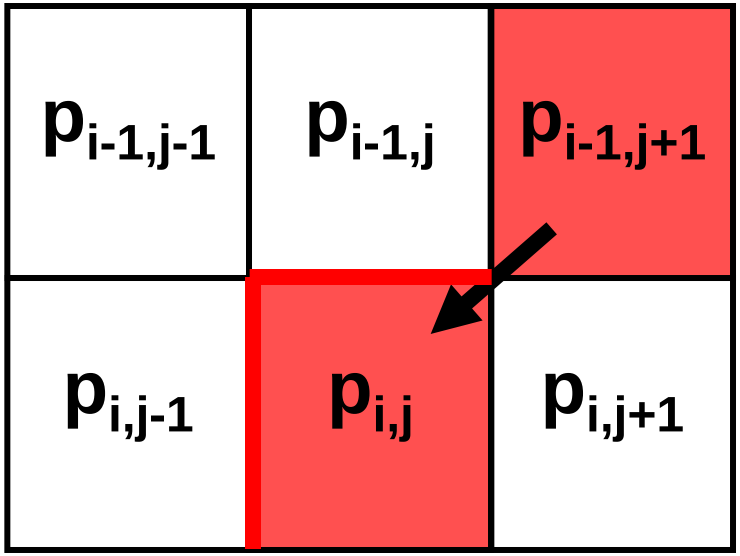
- **Three possibilities when removing pixel $P_{i,j}$**

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



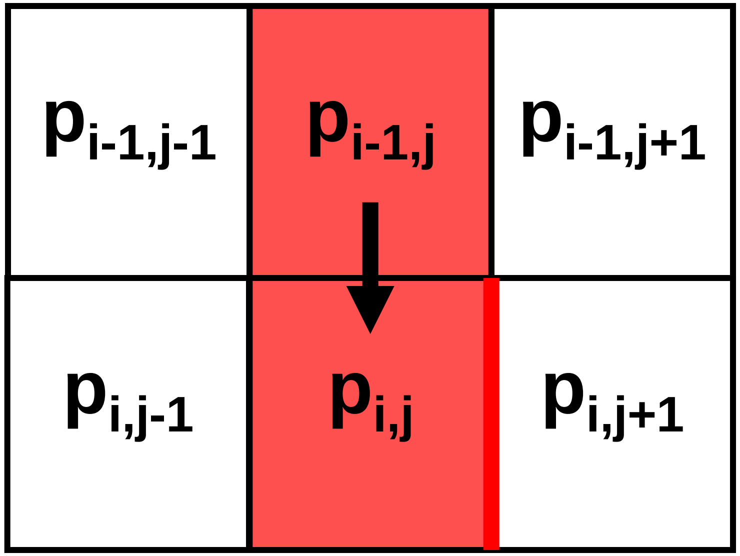
$$C_L(i,j) = |I(i,j+1) - I(i,j-1)| + |I(i-1,j) - I(i,j-1)|$$

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



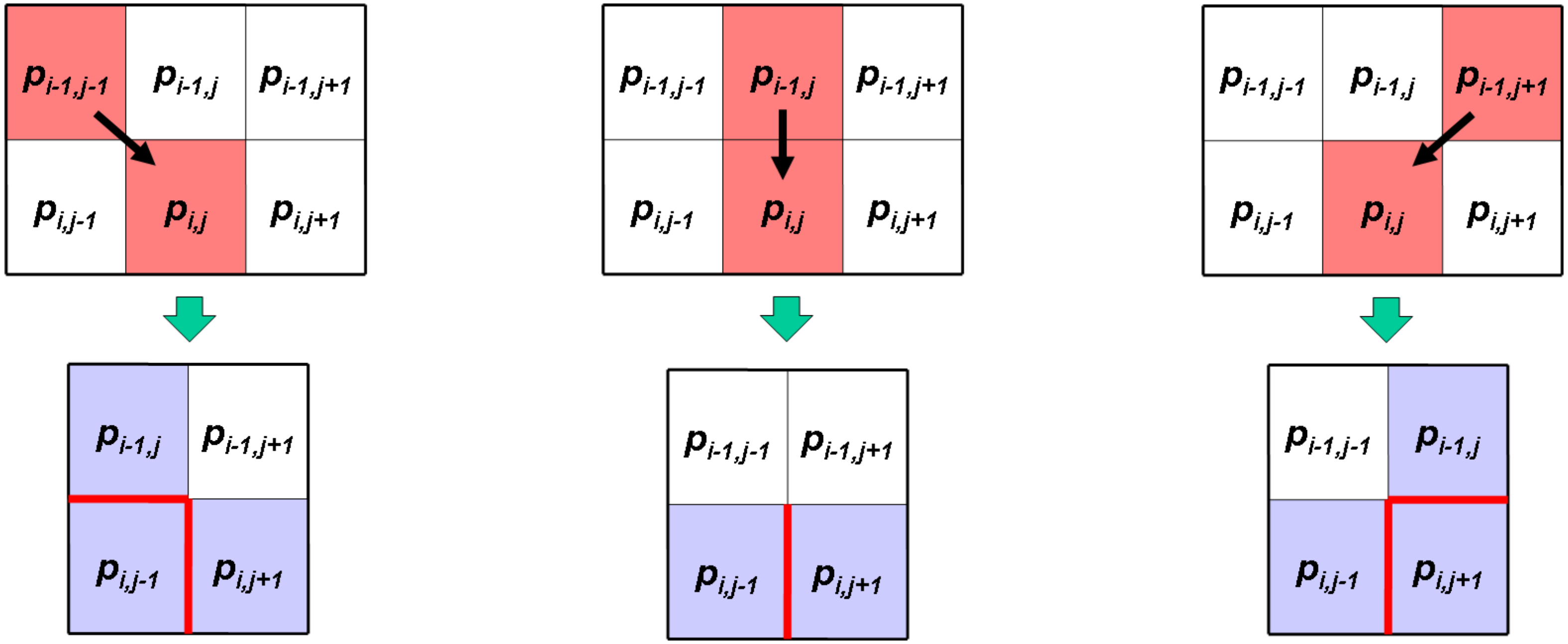
$$C_V(i,j) = |I(i,j+1) - I(i,j-1)|$$

Old "Backward" Energy

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

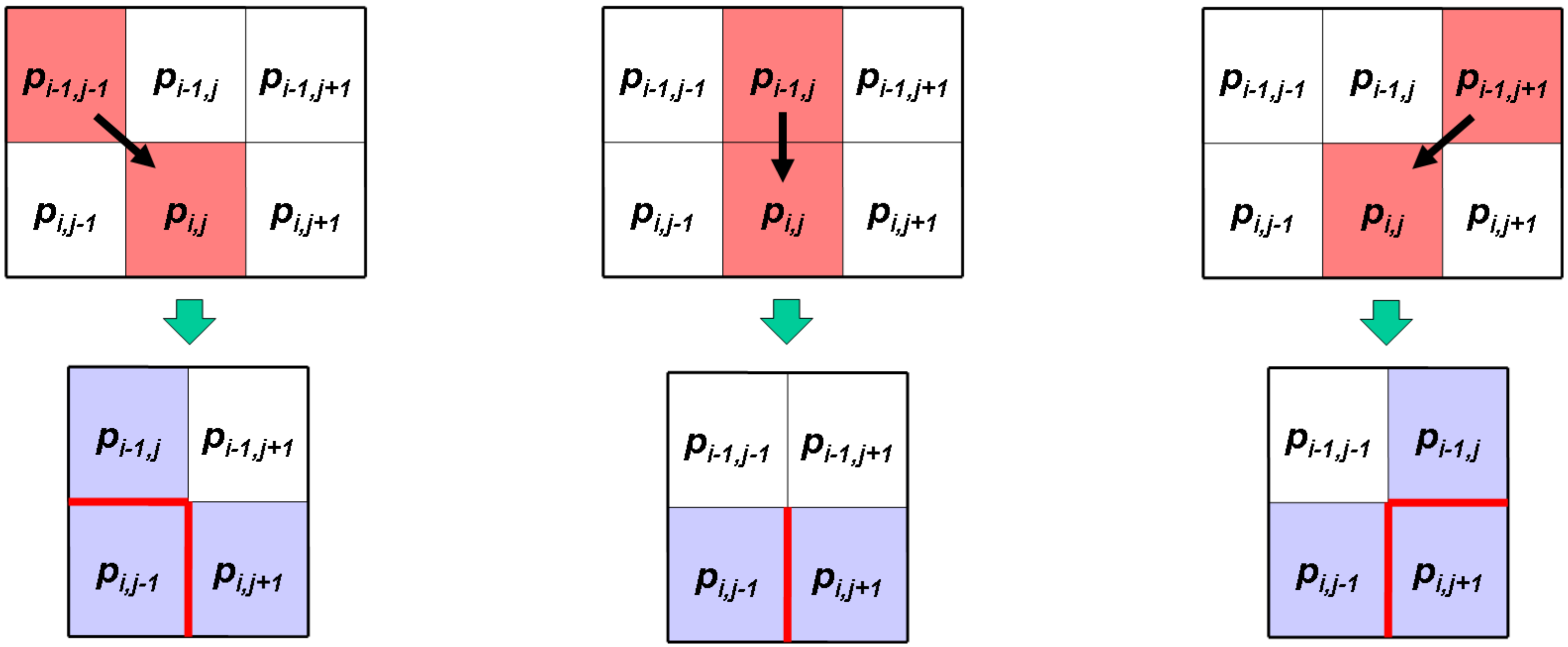
New Forward Looking Energy

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

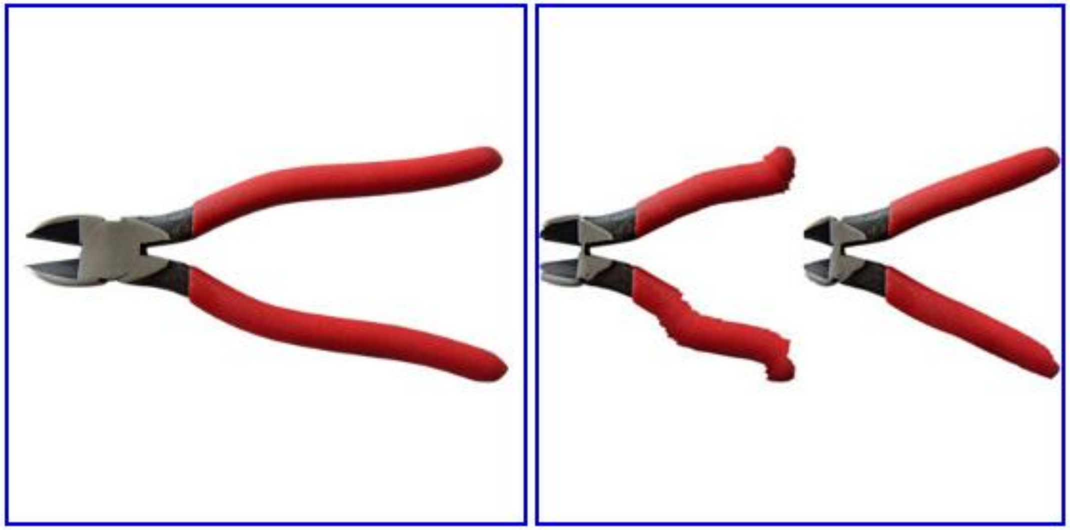


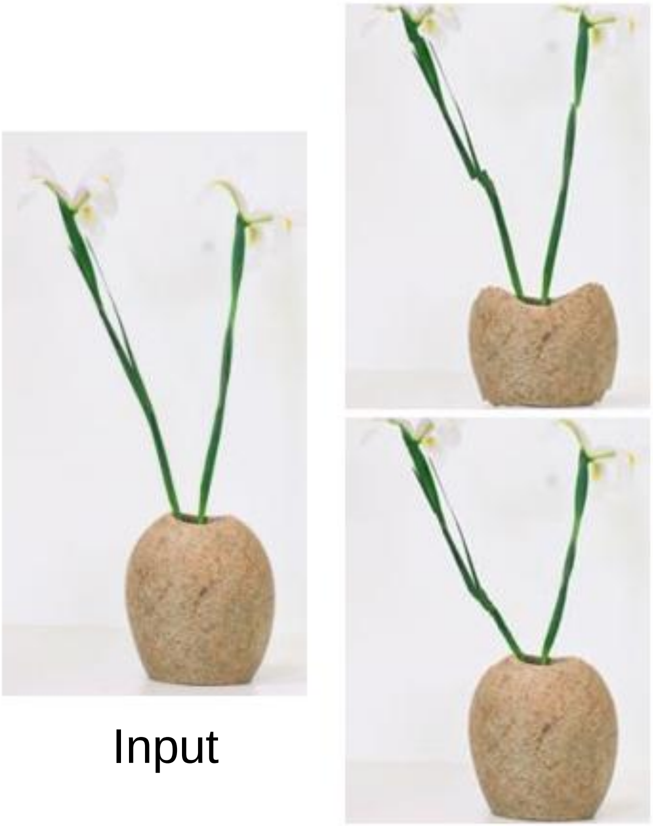
Adding "Pixel Energy"

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

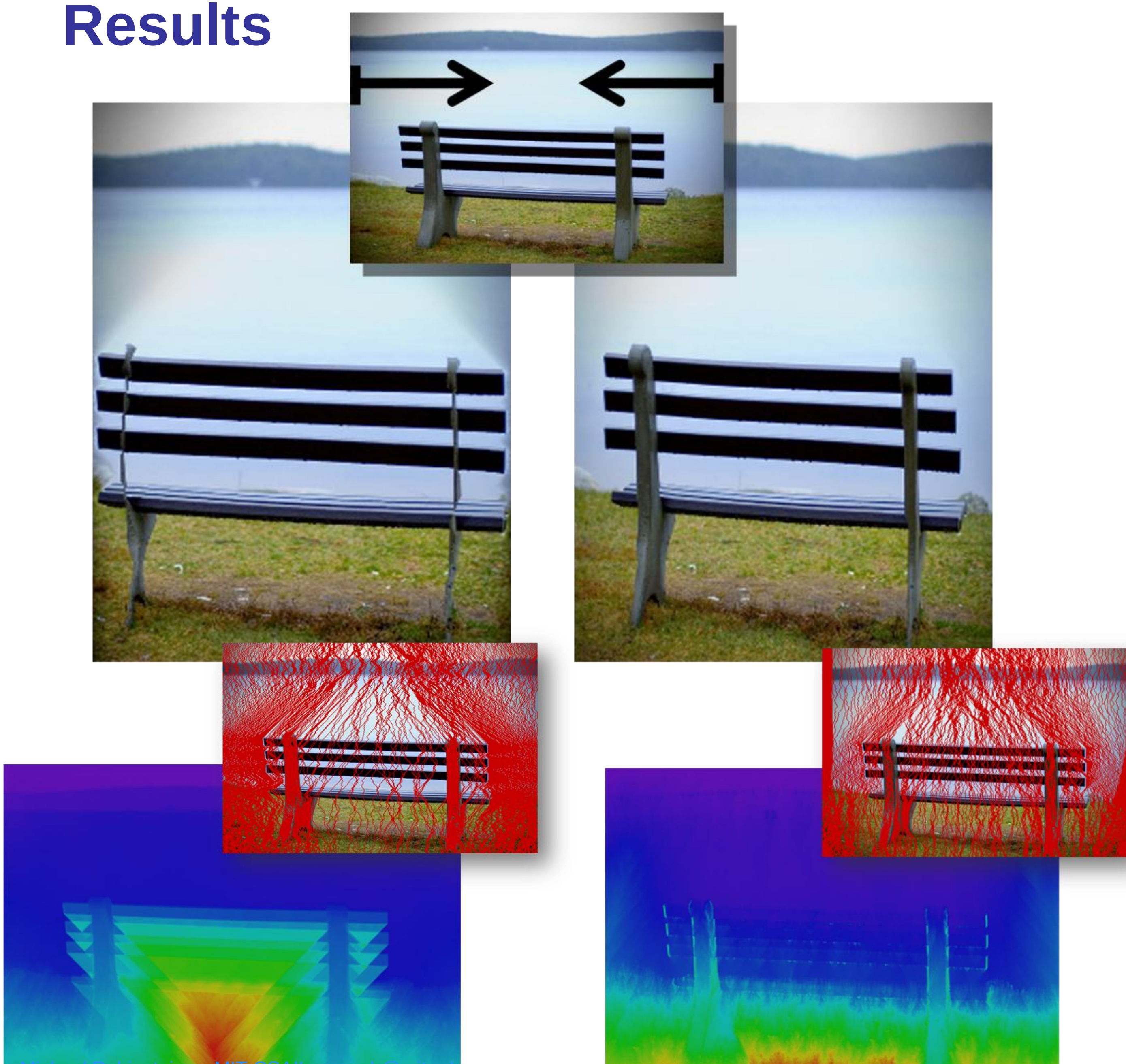


Results





Results

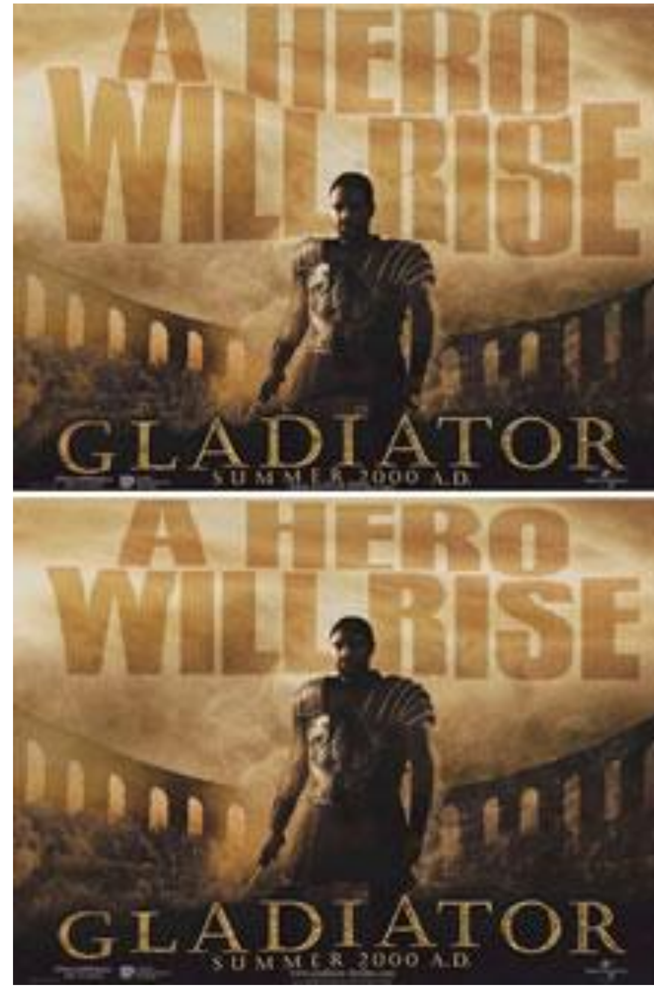


Backward vs. Forward

Backward

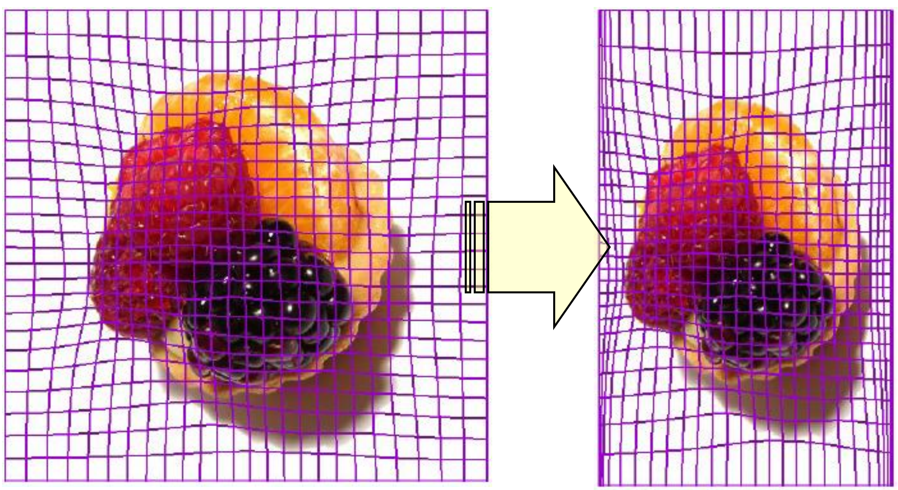
Forward

Results


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

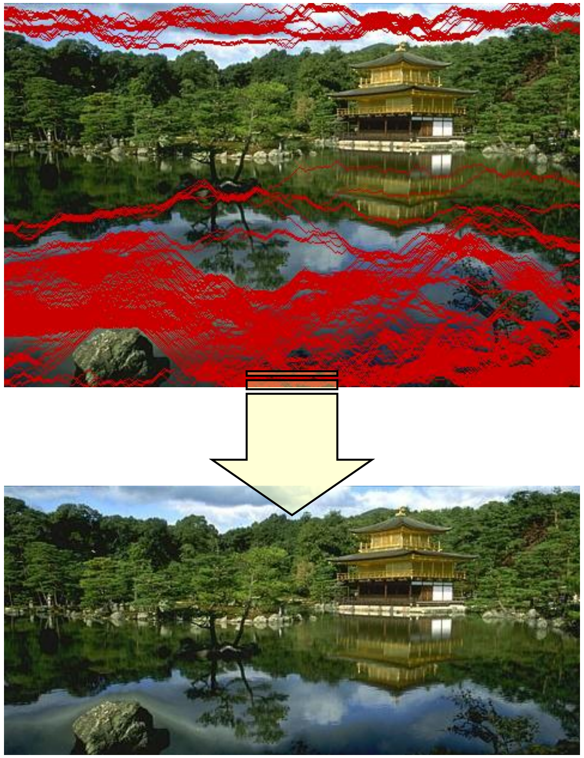
Discrete vs. Continuous

[Wang et al 2008]



Continuous

[Avidan and Shamir 2007]



Discrete

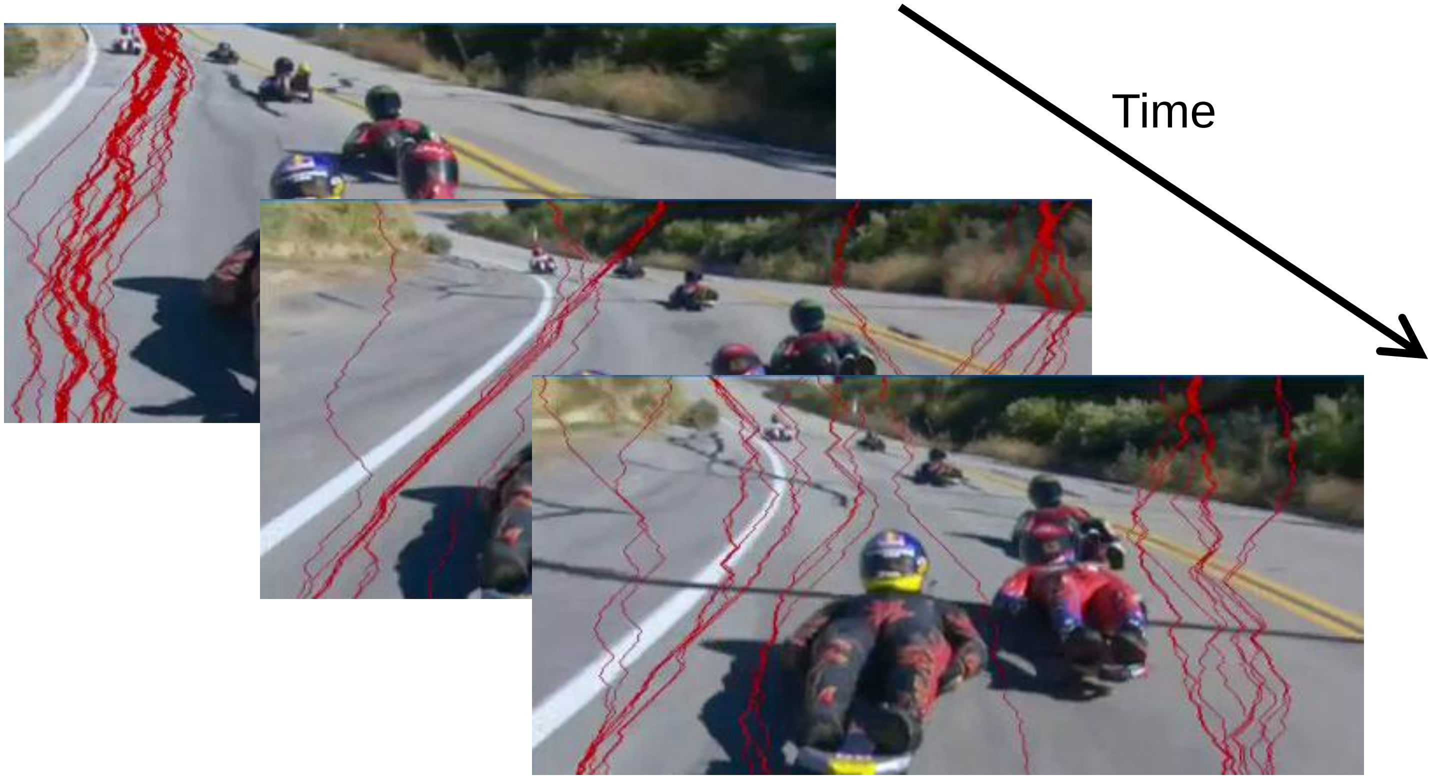
From Images to Videos

- **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 → **30 hours !!**
2. **Dimensionality/algorithmic**
   – Temporal coherency: human visual system is highly sensitive to motion!

Seam-Carving Video?

- **Naive… frame by frame independently**

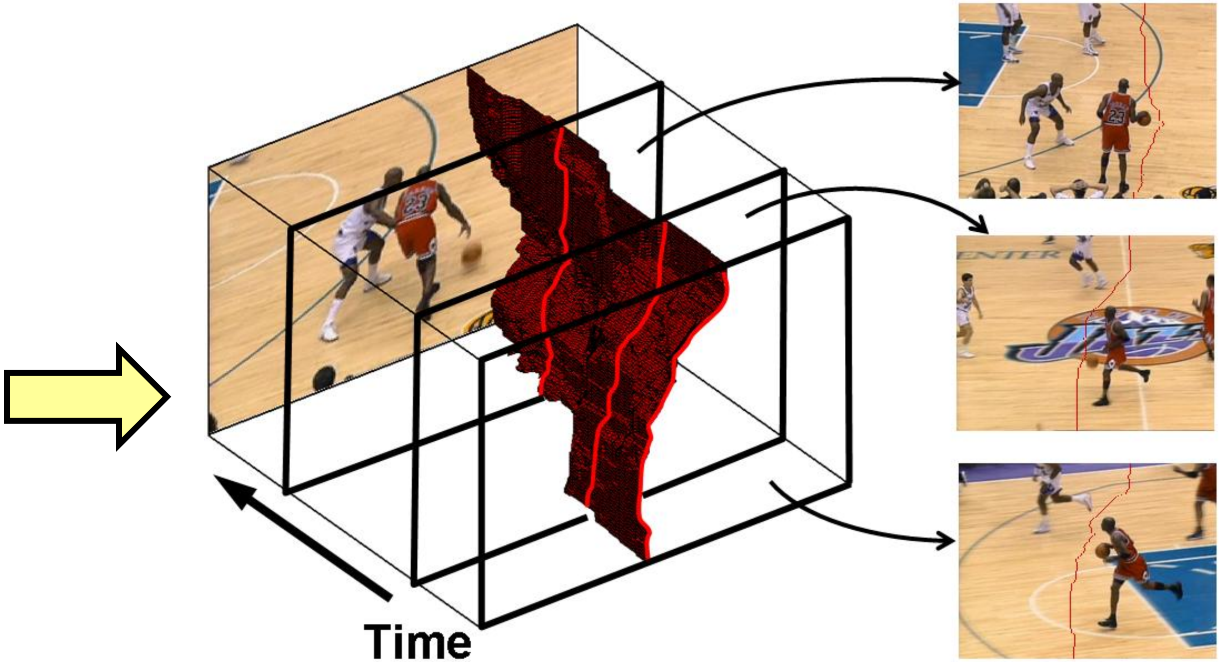


Frame-by-frame Seam-Carving

*Representative seams

From 2D to 3D

1D paths in images



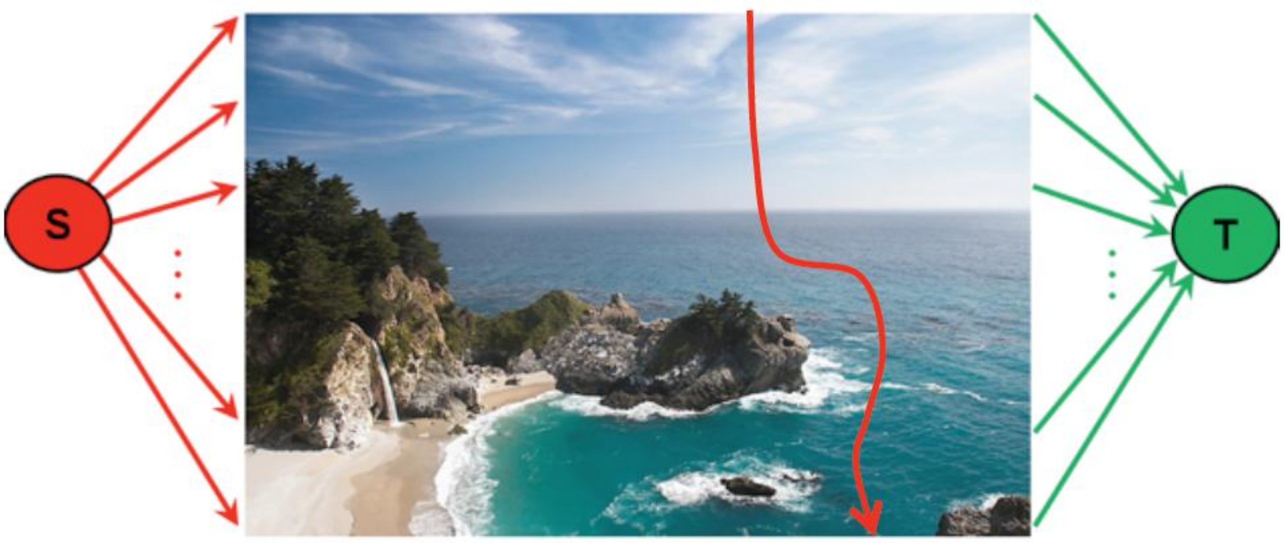
2D manifolds in video cubes

Challenges

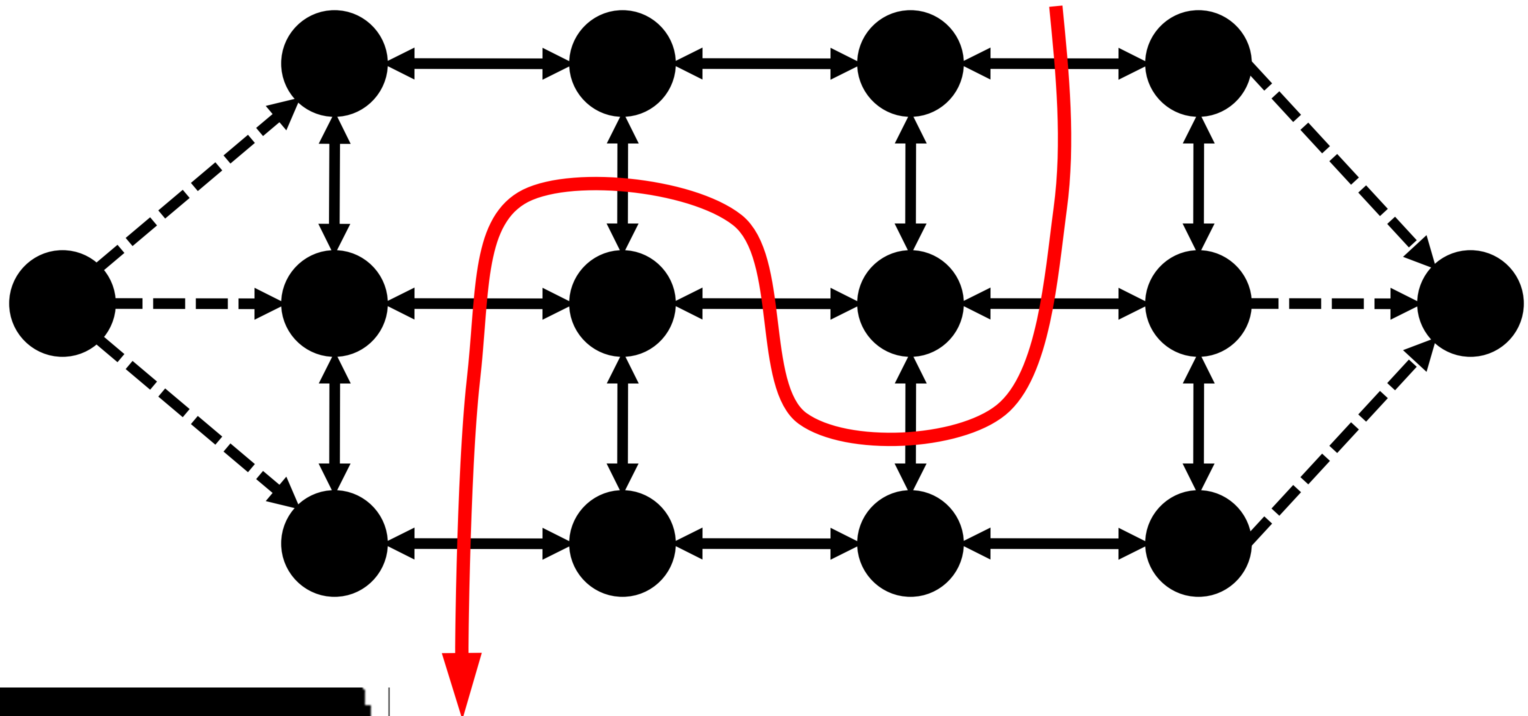
- **Dynamic Programming no longer applicable**
  - Reduction to min-cut graph problem

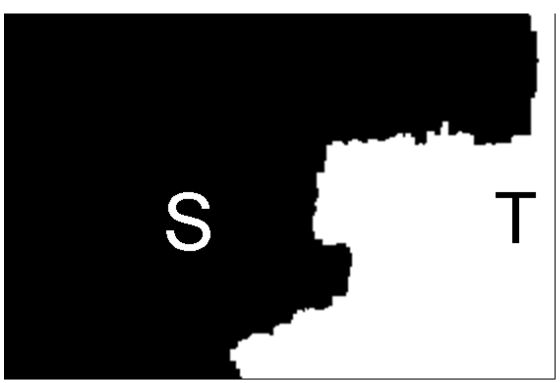
- **Cut must fulfill seam constraints**
  1. *Monotonic* (cut each row exactly once)
  2. Connected

Cut should be a function!



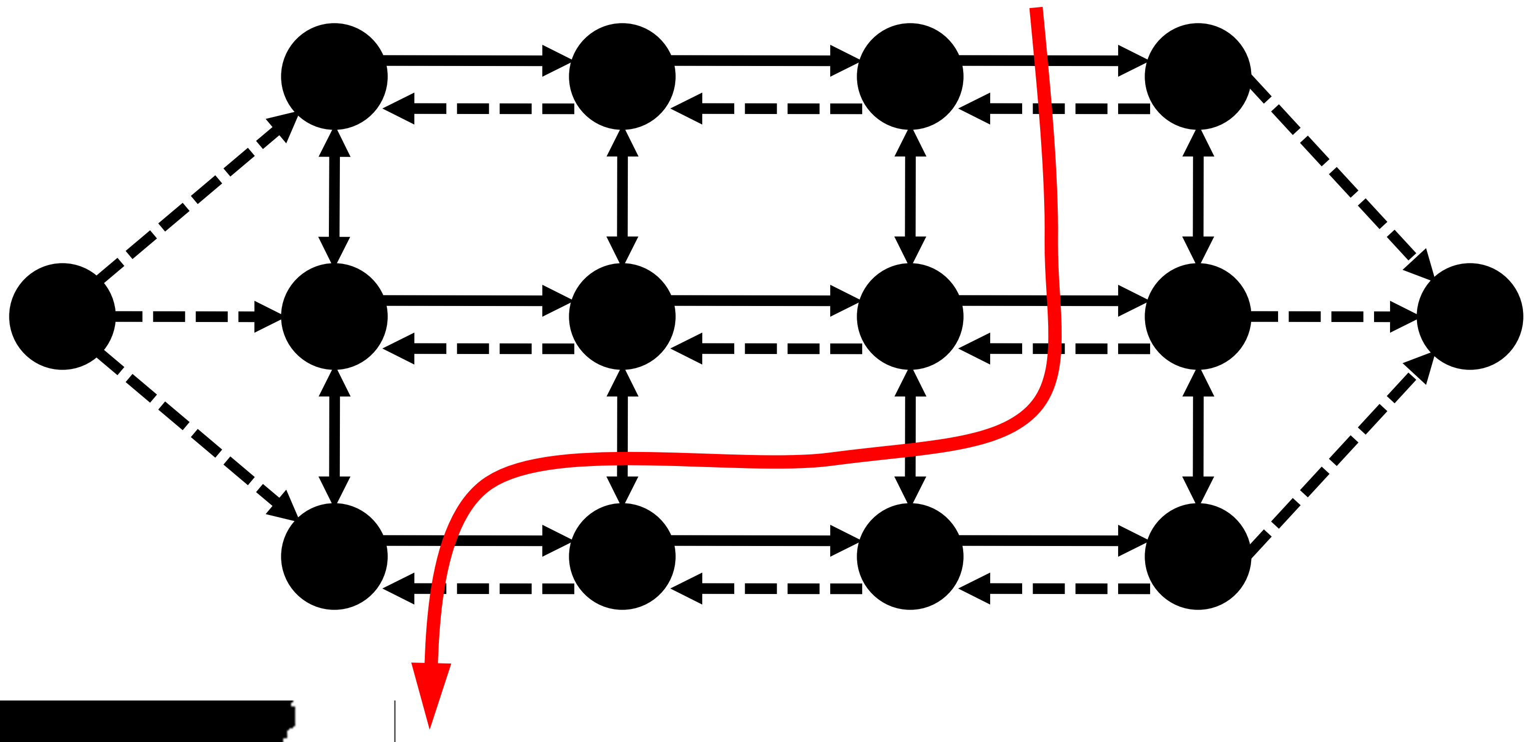
Graph Construction

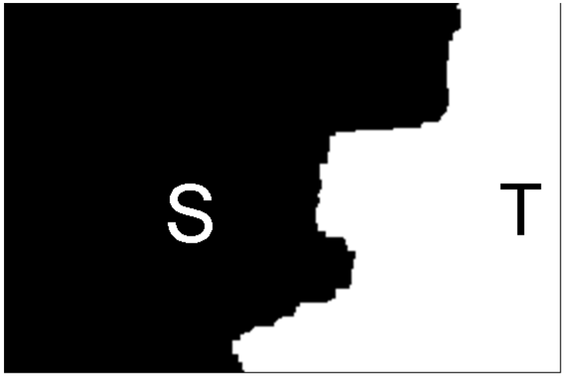




Not monotonic

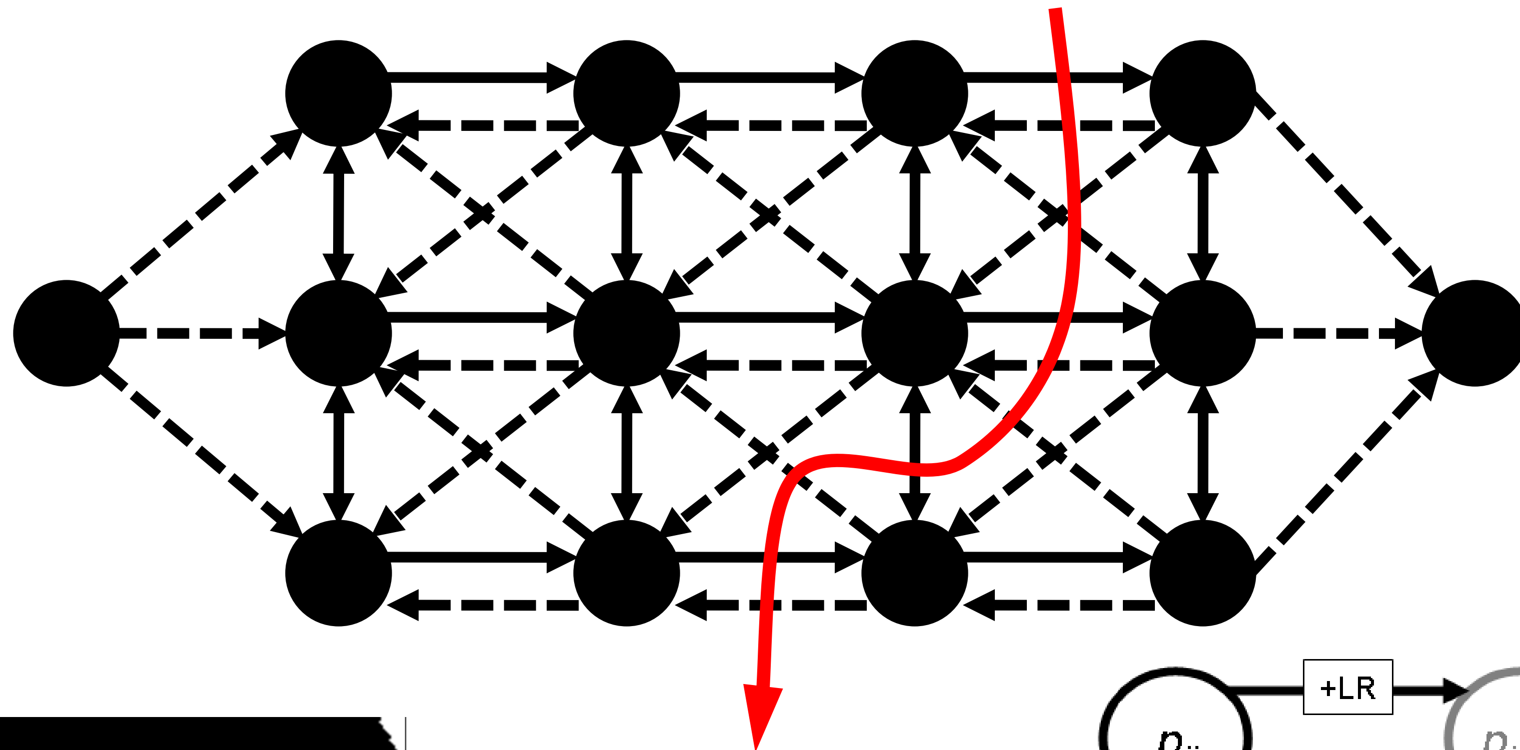
Graph Construction





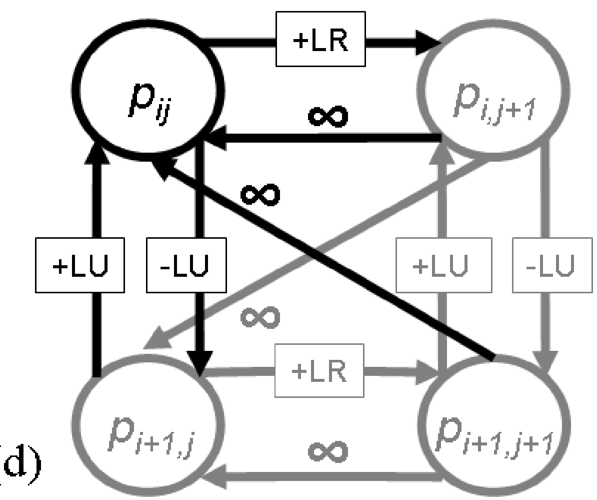
Monotonic, not connected

Graph Construction



Backward Energy

Forward Energy



(d)

Multiresolution Graph Cut

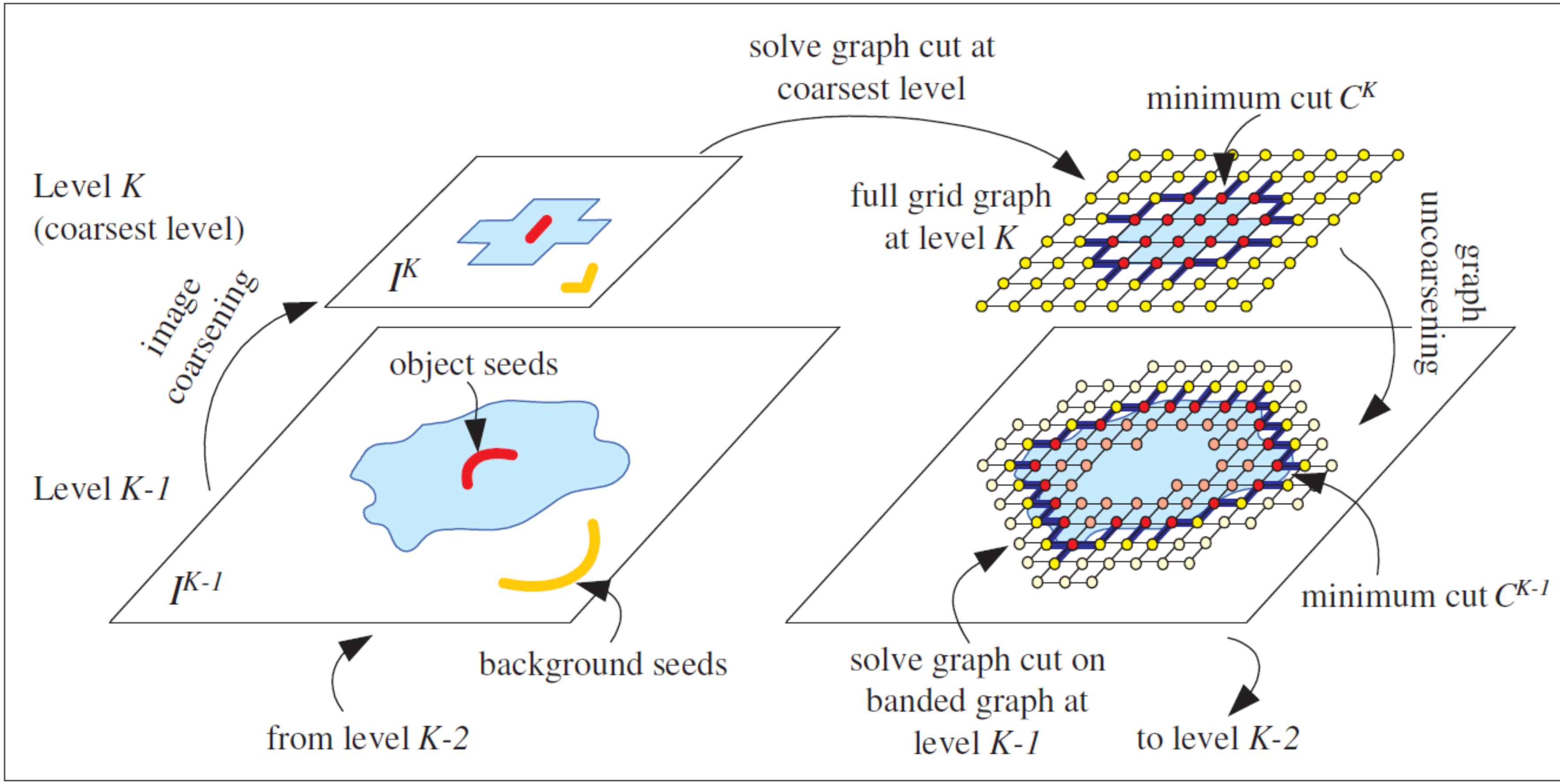


Figure 1. Multilevel banded graph cuts algorithm

Lombaert et al. [2005]

Video Retargeting Video

References

RetargetMe A Benchmark for Image Retargeting

Michael Rubinstein
MIT CSAIL

Diego Gutierrez
Universidad de Zaragoza

Olga Sorkine
New York University

Ariel Shamir
The Interdisciplinary Center

Does retargeting actually work?

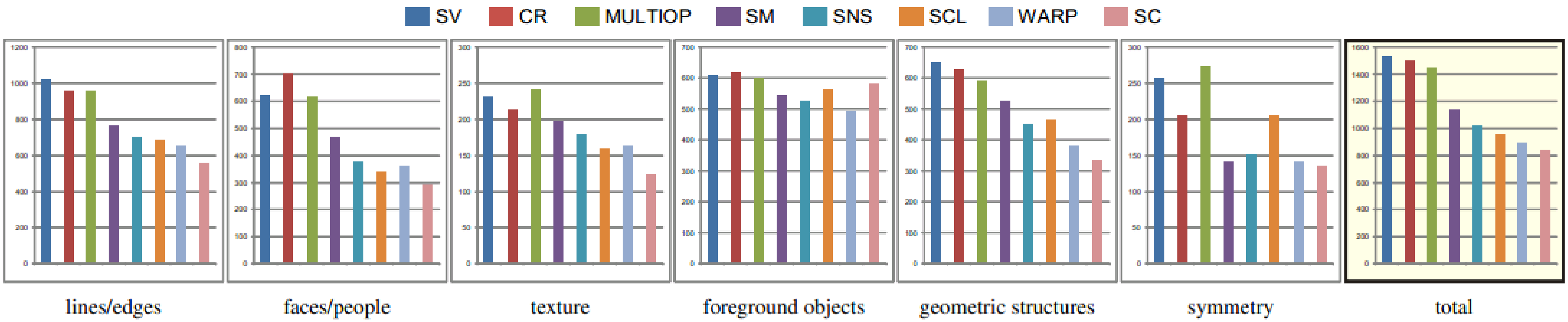


Figure 3: *The number of votes and total ranking (rightmost) of the eight methods per attribute, when the reference (original) image was shown. We notice that three operators, namely* SV, MULTIOP *and* CR *consistently rank better than the others.*

- **CR is cropping**
- **Michael Rubinstein, Diego Gutierrez, Olga Sorkine, Ariel Shamir
A Comparative Study of Image Retargeting
ACM Transactions on Graphics, Volume 29, Number 5 (Proc. SIGGRAPH Asia) 2010**

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**