



# Seam-Carving

and Content-driven Retargeting of Images  
(and Video)

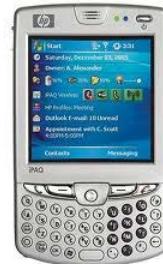
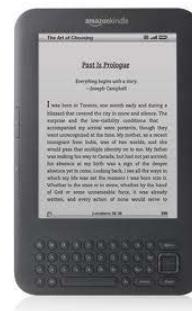


**Michael Rubinstein**  
**MIT**

Some slides borrowed from Ariel Shamir and Shai Avidan



# Display Devices





# Content Retargeting

BBC Mobile

News Sport Weather Travel TV Radio More Search the BBC

**Tunisia leaders quit ruling party**

Tunisia's interim president and PM quit the ruling RCD party, in an apparent bid to defuse anger over the make-up of a day-old government.

» More from BBC News

**News** Edit

Duvalier taken to court in Haiti  
37 minutes ago

- Major earthquake hits SW Pakistan
- Sharks are probably colour-blind
- Bomb kills Iraqi police recruits
- China's Hu arrives for US visit
- 'Road train' trials get rolling
- Israeli tanks in deadly Gaza raid
- Apple shares drop on Jobs' health

**Sport** Edit

Live - Tuesday football  
about 2 hours ago

- Villa confirm record Bent signing
- Pienaar picks Spurs over Chelsea
- Live - Tuesday's Scottish football
- Landsis ends controversial career
- Olazabal named Ryder Cup captain
- Liverpool accept offer for Babel
- Rafael accepts misconduct charge

**Business** Edit

**MARKET DATA TUE, 18 JAN 2011 20:58 GMT**

Dow Jones	11853.03	▲	65.65
Nasdaq	2765.82	▲	10.52
FTSE 100	6056.43	▲	70.73
Dax	7143.45	▲	65.39
Cac 40	4012.68	▲	37.27

15 minute delay | Terms and Conditions

**STERLING EXCHANGE RATES**

Dollar	1.5959	▲	0.0071
Euro	1.1922	▼	-0.0026

15 minute delay | Terms and Conditions

**Entertainment** Edit

King's Speech leads Bafta field  
about 6 hours ago

- Apple shares drop on Jobs' health
- TV giant laments with NBC clearance
- Boeing postpones Dreamliner again
- Russian bank staff investigated
- UK inflation rate rises to 3.7%

**Sci & Environment** Edit

India aims for tidal power first  
about 6 hours ago

- Galileo price rises 1.9bn euros
- Sharks are probably colour-blind

**World Service** Edit

NEWS IN 32 LANGUAGES

العربية	فارسی	اردو
中文	हिन्दी	Somali
Русский	Brasil	Mundo

More languages

**TV Channels** Edit

BBC World News	BBC Entertainment
BBC America	Animal Planet
BBC Canada	People+Arts
BBC Kids (Canada)	UKTV Australia

PC

**BBC News**

 China's Hu arrives for US visit

 Duvalier taken to court in Haiti

 Tunisia leaders quit ruling party

More from BBC News

**BBC Sport**

Live - Tuesday football

Villa confirm record Bent signing

More from BBC Sport

**BBC World Service**

 Inside the IMF

 The King's Speech: A stammerer's perspective

 Hourly news bulletin

More audio from BBC World Service

**Weather**

Find 5 day forecast

**Languages**

Spanish

News in more languages

---

**From BBC Mobile UK**

- Television
- Radio & Music
- Entertainment
- BBC Children
- Learning
- Lifestyle
- Food

[▲ Top](#)

---

- FAQ

iPhone



# Page Layout

Page layout - Wikipedia, the free encyclopedia - Mozilla Firefox

File Edit View History Bookmarks Tools Help

W http://en.wikipedia.org/wiki/Page\_layout

W 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

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.

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%

Consumer magazine sponsored advertisements and covers rely heavily on professional page layout skills to compete for visual attention.



# Simple Media Retargeting Operators

Let's begin



# Content-aware Retargeting Operators

Content-aware



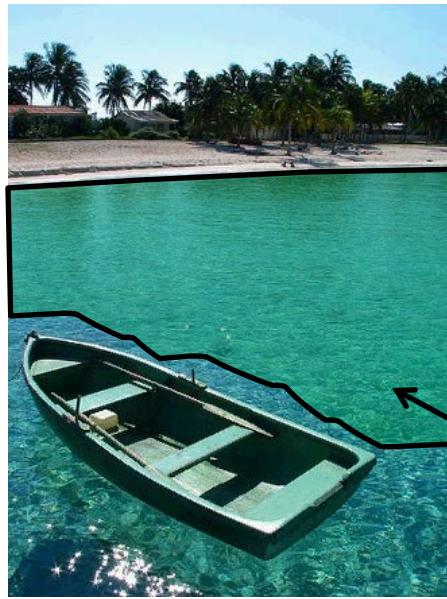
"Important"  
content



Content-  
oblivious



# Content-aware Retargeting



Input



Scale



Crop



Content-aware

“less-Important”  
content



# Image Retargeting

- **Problem statement:**
  - Input Image  $I$   $n \times m$ , and new size  $n' \times m'$
  - Output Image  $I'$  of size  $n' \times m'$  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!**

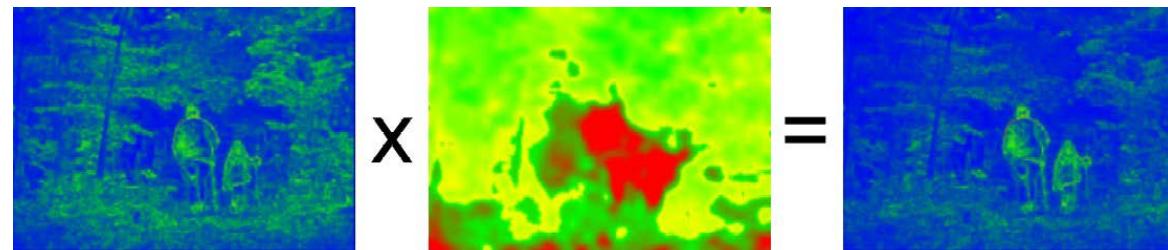


# Image/Video Retargeting

- 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  $S: p \rightarrow [0,1]$



Wang et al. 2008

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

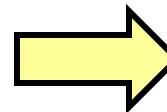


Judd et al. ICCV09 *Learning to predict where people look*

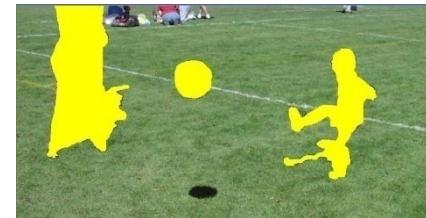
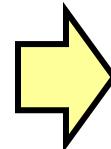


# General Retargeting Framework

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



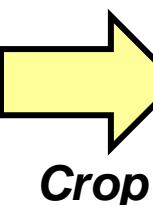
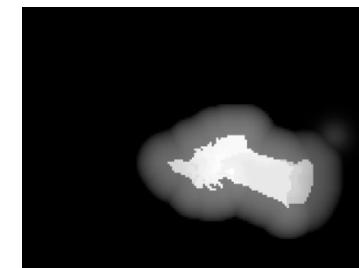
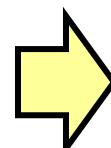
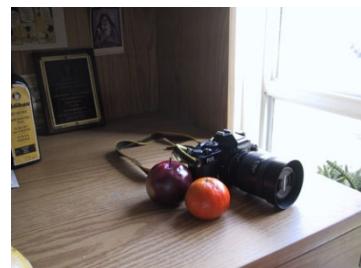
2. Use some operator(s) to change the image  $\mathbf{I}$



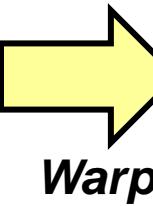
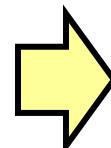
*Recompose*



Setlur et al.  
[2005]



Santella et  
al. [2005]



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)**

# Cropping





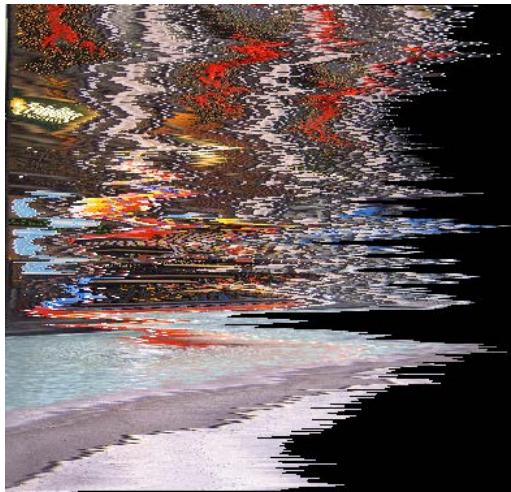
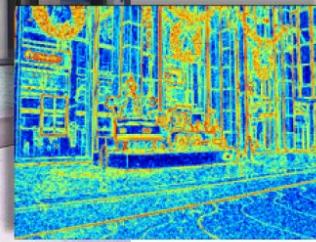
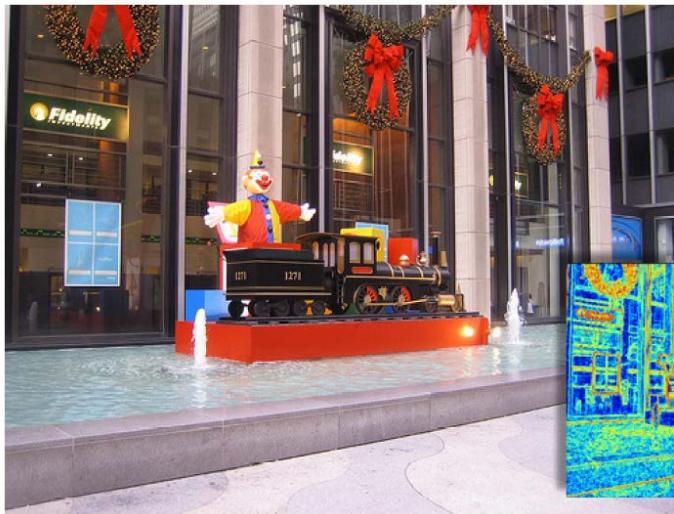
# Seam Carving

- Assume  $m \times n \rightarrow m \times n'$ ,  $n' < n$  (summarization)
- Basic Idea: remove unimportant pixels from the image
  - Unimportant = pixels with less “energy”

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

- 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)



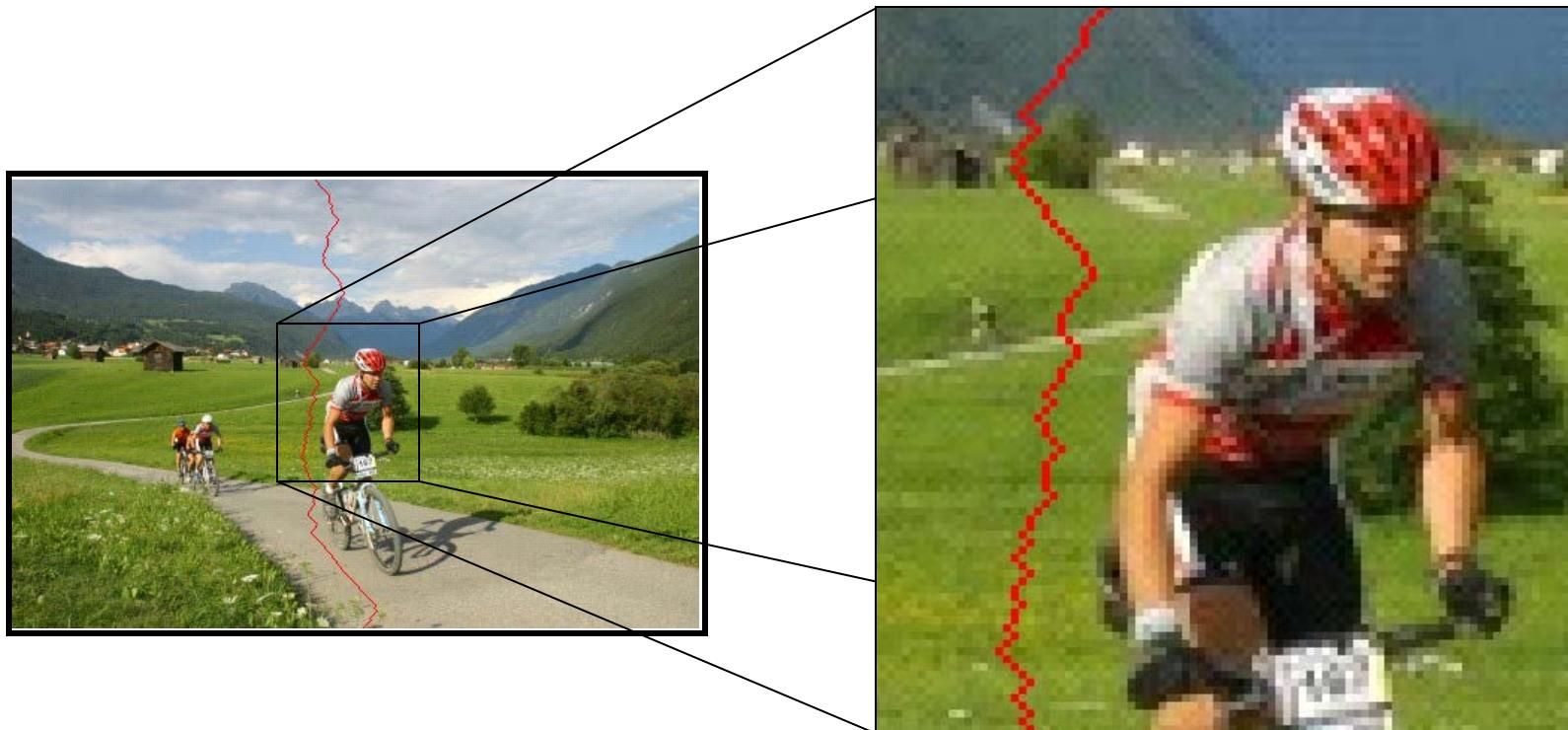
Least-energy columns

# 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

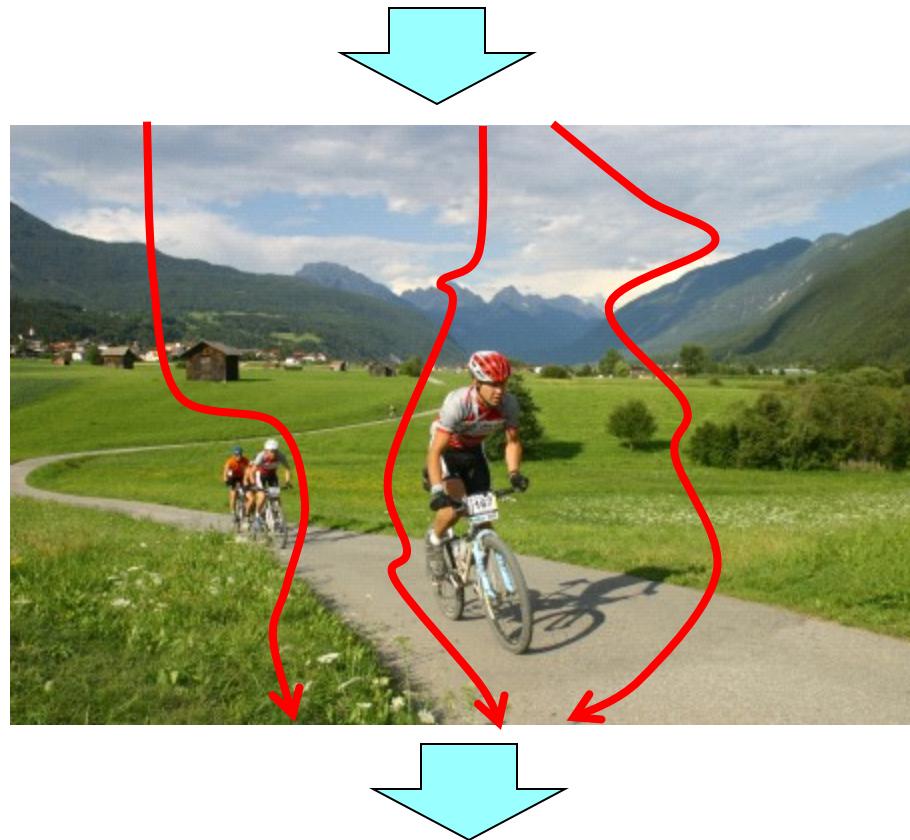
- Efros & Freeman [2001] – Texture synthesis
- Kwatra et al. [2003] – Image and video synthesis
- Agarwala et al. [2004] – Digital Photomontage
- Perez et al.
- Jia et al.
- Rother et al.
- Wang and Cohen [2006] – simultaneous matting and compositing

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

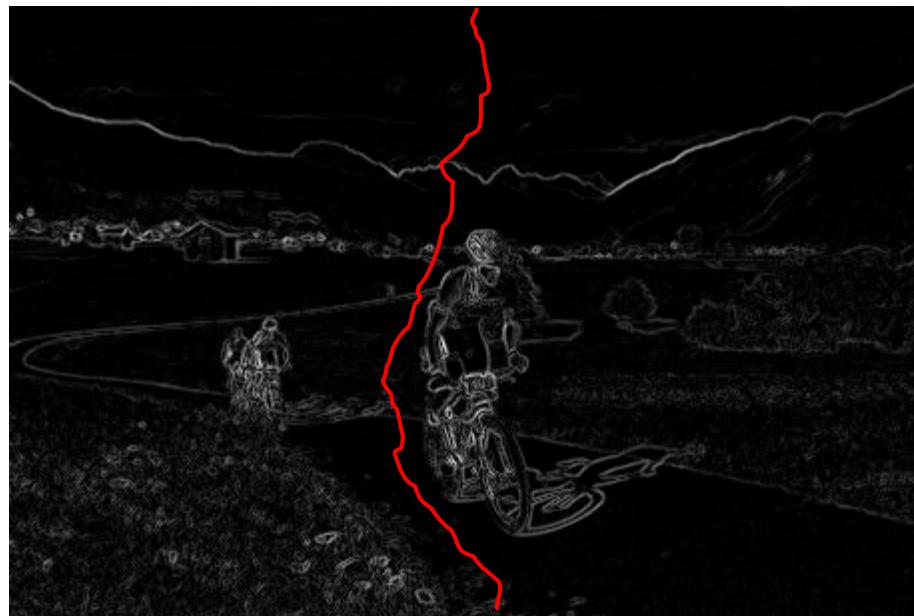


Agarwala et al., *Interactive Digital Photomontage*

# 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**

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

- **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)

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

A 4x4 grid of numbers representing costs. Red arrows point from the top-left cell (5) to the second column of the second row (2), and from the second column of the second row (2) to the third column of the second row (3).



# Dynamic Programming

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

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

A 4x4 grid of numbers. Red arrows point from the top row's values 5, 8, 12, 3 to the second row's value 2+5. The value 2+5 is highlighted with a red box.



# Dynamic Programming

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

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

A 4x4 grid of numbers representing a dynamic programming table. The top row contains 5, 8, 12, and 3. The second row contains 9, 7, 3+3, and 9. The third row contains 7, 3, 4, and 2. The bottom row contains 4, 5, 7, and 8. Red arrows point from the value 3 in the second row to the value 3+3 in the second row, and from the value 3+3 in the second row to the value 2 in the third row, illustrating the calculation of the minimum value for the cell (2,3).



# Dynamic Programming

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

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

# 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

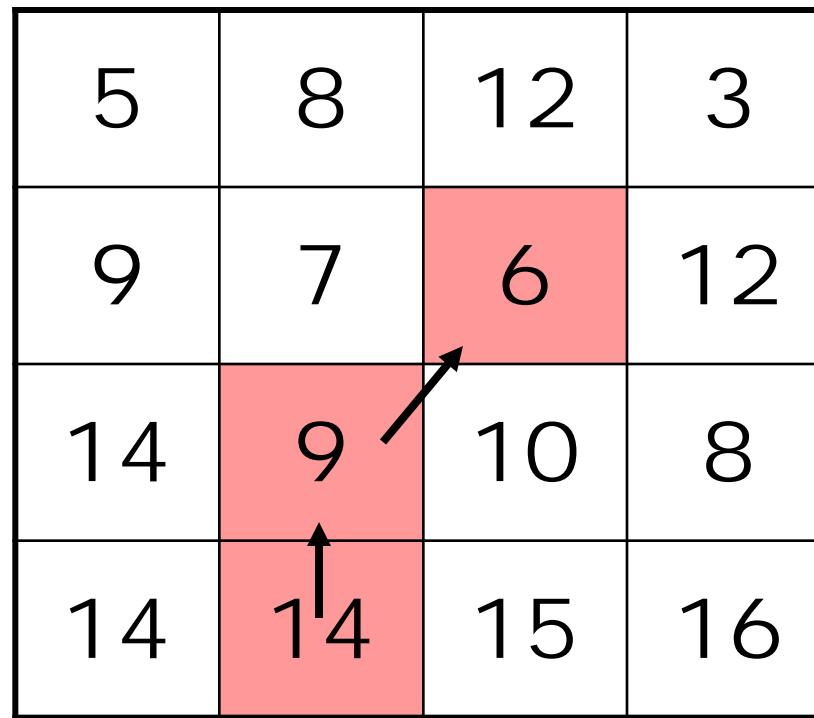
↑

# Backtracking the Seam

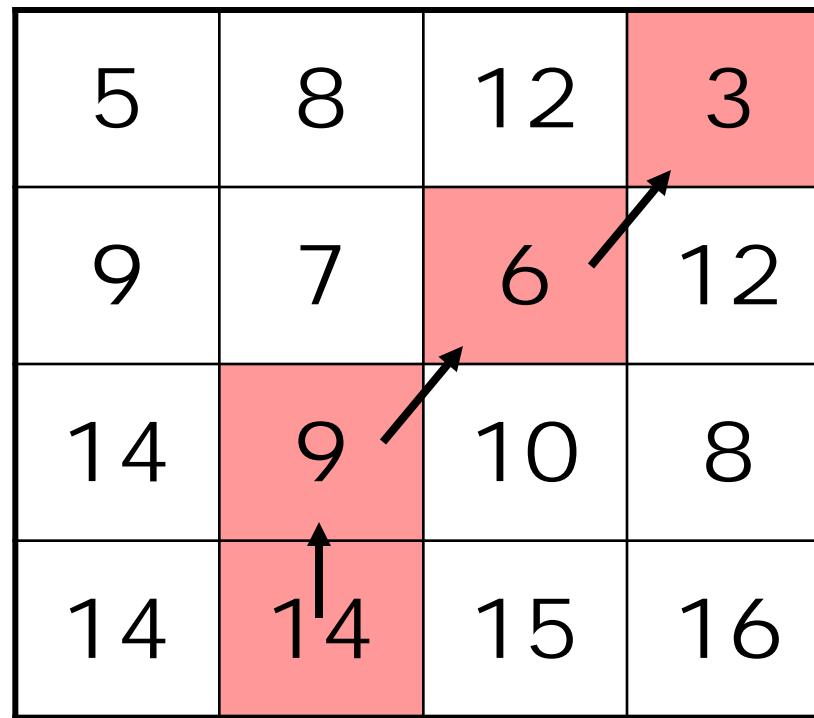
5	8	12	3
9	7	6	12
14	9	10	8
14	14	15	16

A 4x4 grid of numbers. The bottom-left cell (14) is highlighted in red. An arrow points from the bottom-left cell (14) up to the cell above it (9), indicating the direction of backtracking.

# Backtracking the Seam

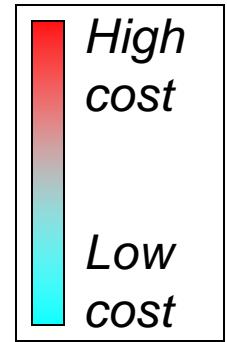
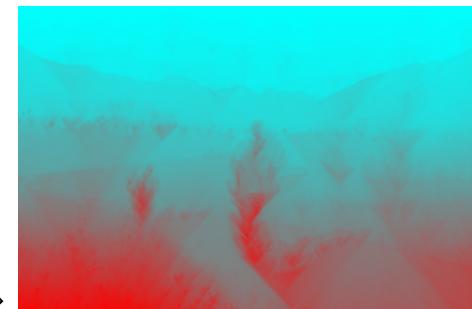


# Backtracking the Seam

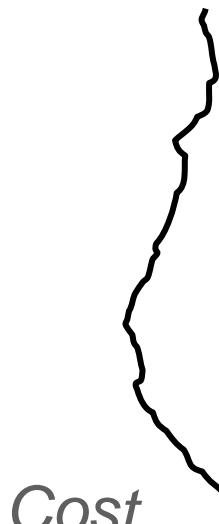




# H & V Cost Maps



*Horizontal Cost*



*Vertical Cost*

# Seam Carving



# The Seam-Carving Algorithm

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

## 1. Do (n-n') times

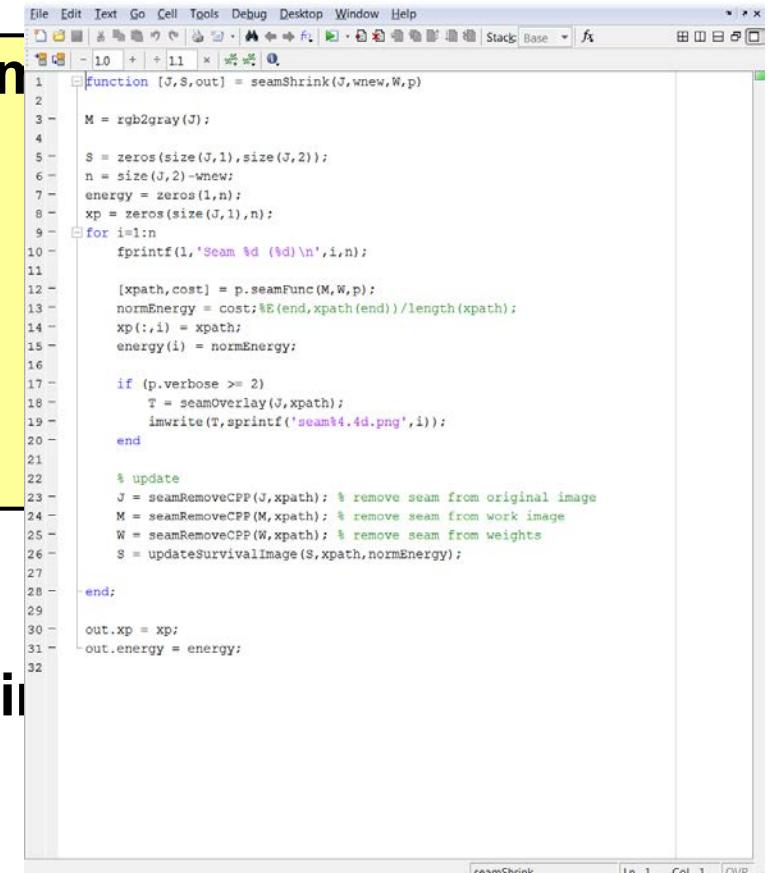
- 2.1. E  $\leftarrow$  Compute energy map on im
- 2.2. s  $\leftarrow$  Find optimal seam in E
- 2.3. im  $\leftarrow$  Remove s from im

## 2. Return im

- For vertical resize: transpose the input image

- Running time:

2.1 O(mn) 2.2 O(mn) 2.3 O(mn)  
 → O(dmn) d=n-n'



```

File Edit Text Go Cell Tools Debug Desktop Window Help
J S out
function [J,S,out] = seamShrink(J,wnew,W,p)
%
M = rgb2gray(J);
%
S = zeros(size(J,1),size(J,2));
n = size(J,2)-wnew;
energy = zeros(1,n);
xp = zeros(size(J,1),n);
%
for i=1:n
    fprintf(1,'Seam %d (%d)\n',i,n);

    [xpath,cost] = p.seamFunc(M,W,p);
    normEnergy = cost.*end.*xpath(end))/length(xpath);
    xp(:,i) = xpath;
    energy(i) = normEnergy;

    if (p.verbose >= 2)
        T = seamOverlay(J,xpath);
        imwrite(T,sprintf('seam%4.4d.png',i));
    end

    %
    J = seamRemoveCPP(J,xpath); % remove seam from original image
    M = seamRemoveCPP(M,xpath); % remove seam from work image
    W = seamRemoveCPP(W,xpath); % remove seam from weights
    S = updateSurvivalImage(S,xpath,normEnergy);

end;

out.xp = xp;
out.energy = energy;

```

# Changing Aspect Ratio





# Changing Aspect Ratio



*Original*



*Seam Carving*



*Scaling*



# Changing Aspect ratio



*Cropping*



*Seams*



*Scaling*

# Changing Aspect Ratio



**Original**



**Retarget**



*Scaling*

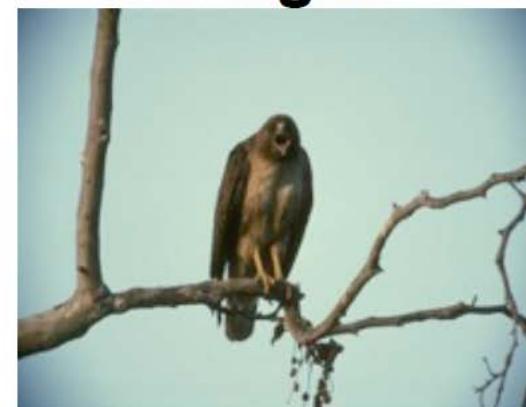
# Changing Aspect Ratio



**Original**



**Retarget**



*Scaling*

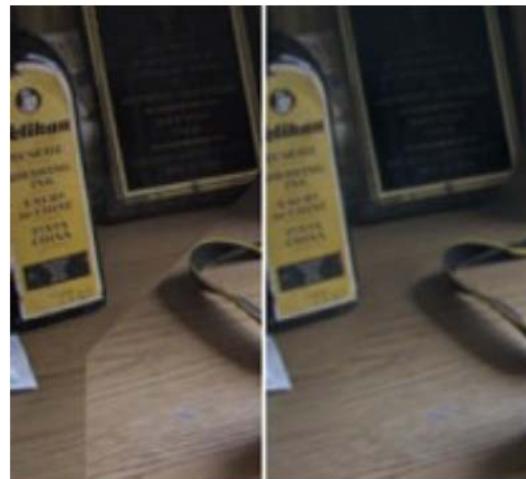


# Content Enhancement





# Seam Carving in the Gradient Domain





# 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?

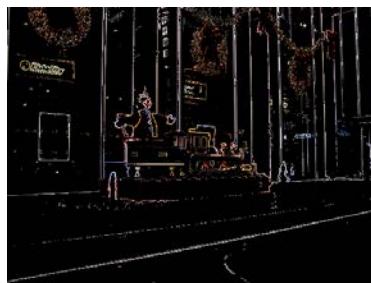
- Q: Will the result be the same if the image is flipped upside down?
- A: Yes (up to numerical stability)
  
- Q: Can we improve the running time?
- A: By factor (local operations)
  
- Q: What happens to the overall energy in the image during seam carving?



# Preserved Energy



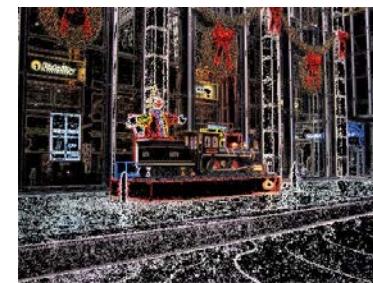
*Energy*



10%



30%



40%



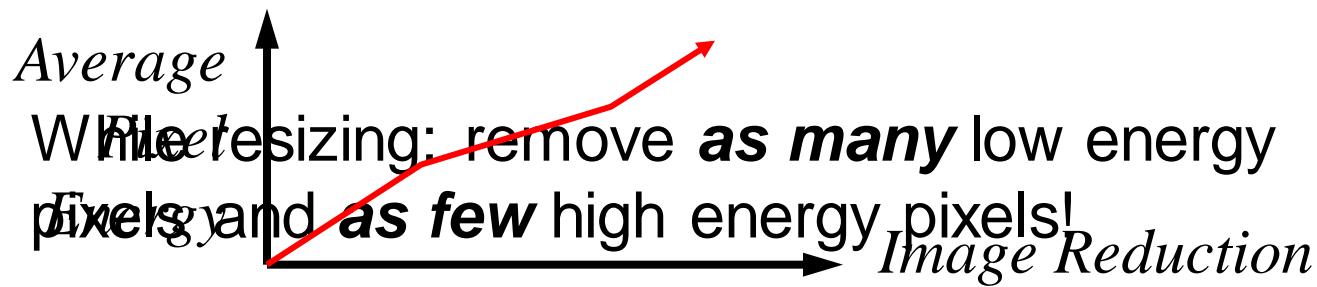
75%

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!





# Preserved Energy



Average  
Pixel  
Energy

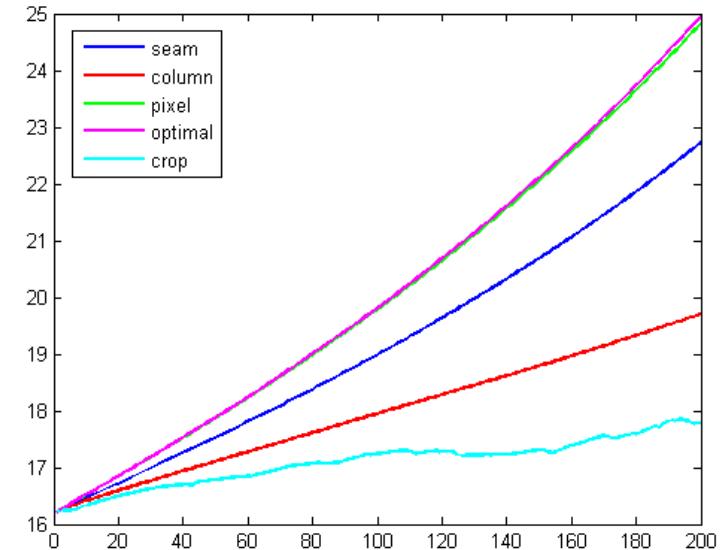


Image Reduction →



crop



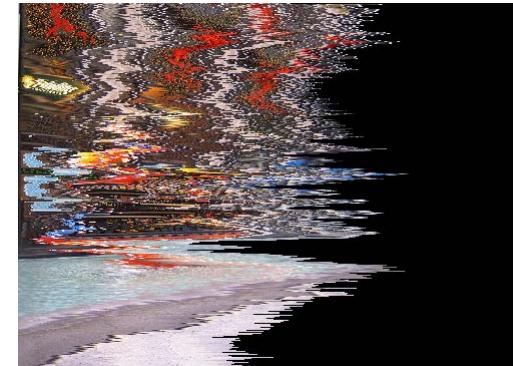
column



seam



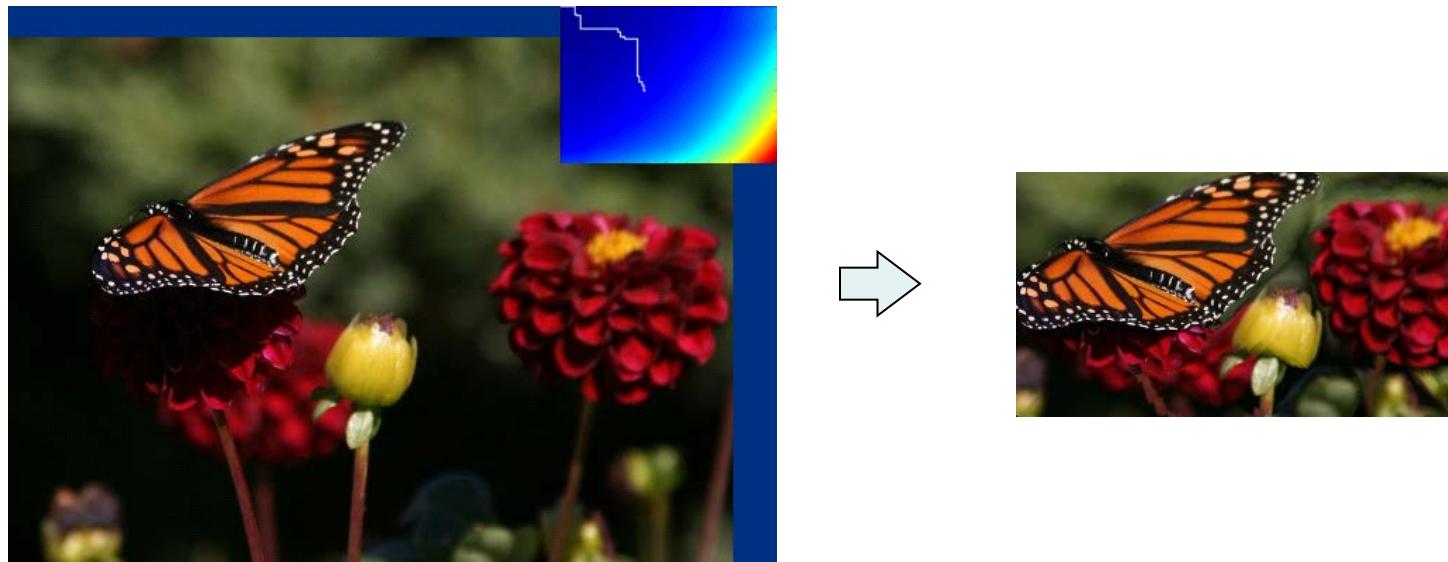
pixel



optimal

# Both Dimensions?

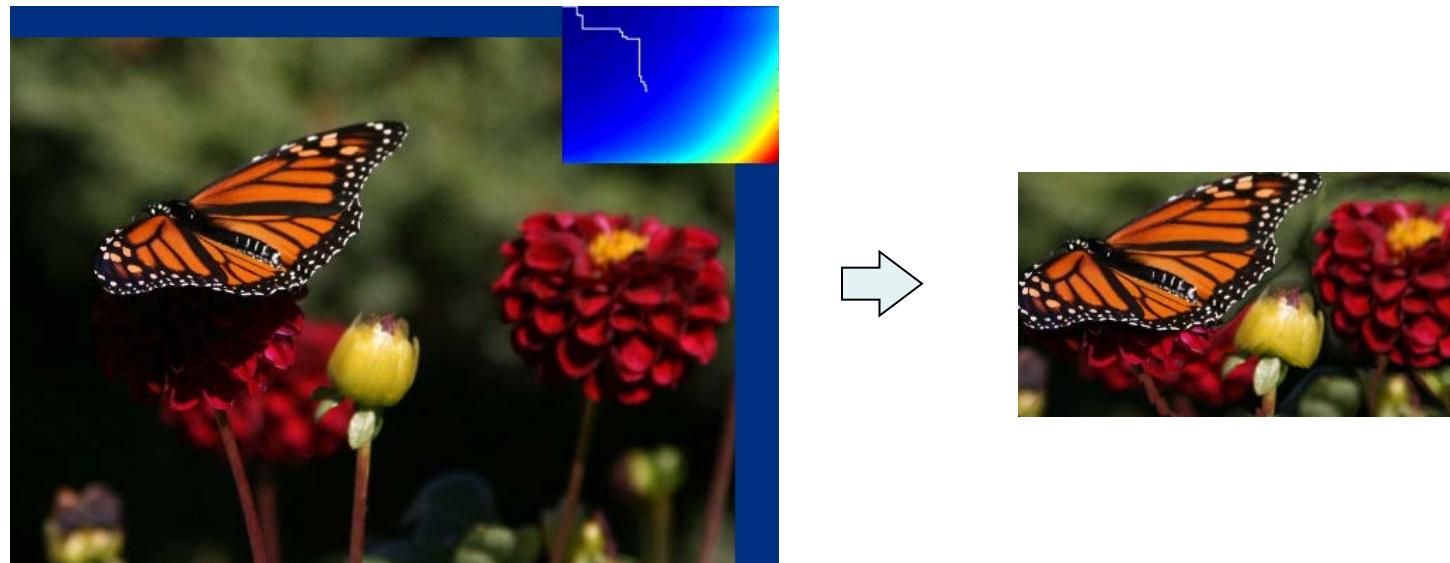
- $m \times n \rightarrow m' \times n'$
- Remove horizontal seam first?
- Remove vertical seams first?
- Alternate between the two?
- The optimal order can be found! → Dynamic Prog (again)



# Retargeting in Both Dimensions

- The recursion relation:

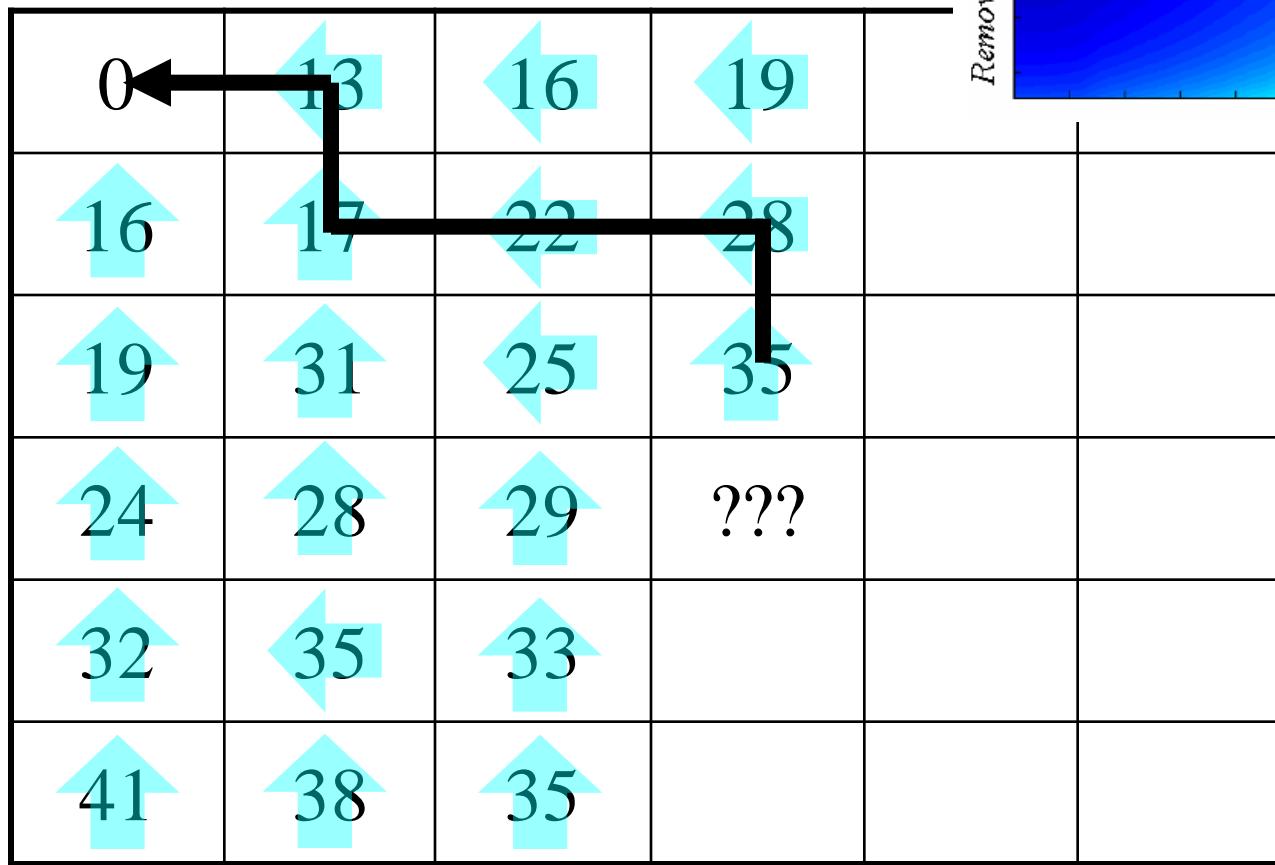
$$T(r, c) = \min(T(r - 1, c) + E(s^x(I_{n-r-1 \times m-c})), T(r, c - 1) + E(s^y(I_{n-r \times m-c-1})))$$



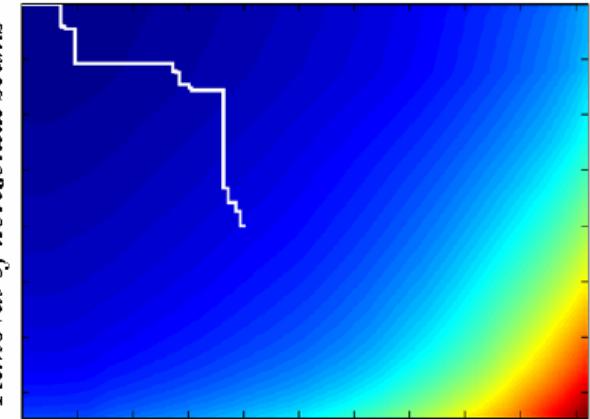
# Optimal Order Map

*Removal of horizontal seams*

*Removal of vertical seams*



*Removal of horizontal seams*



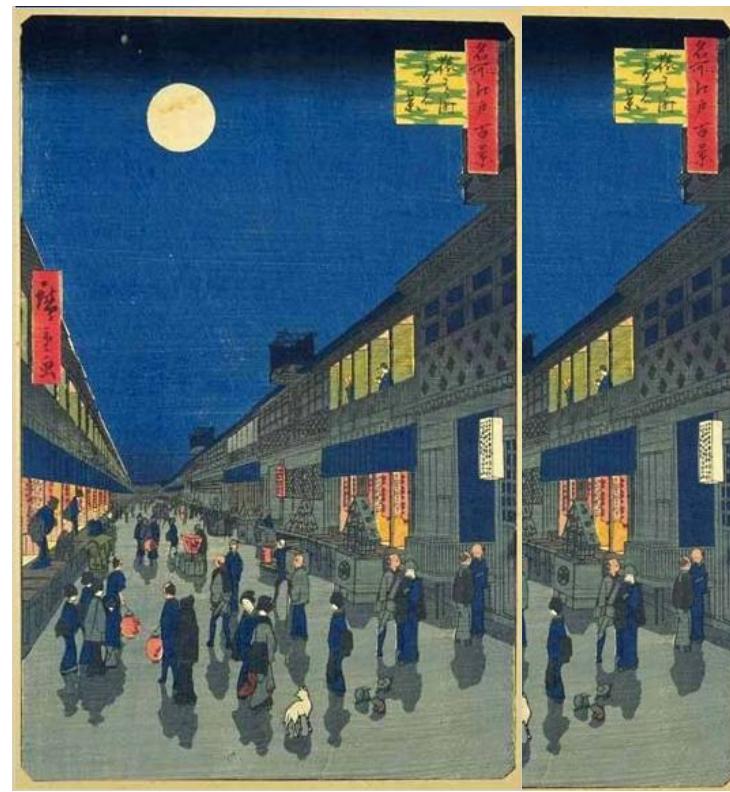
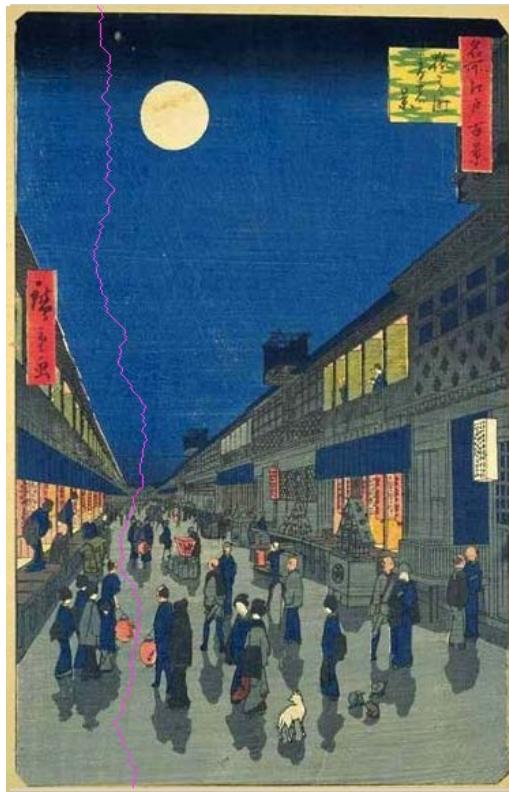
# Optimal?

- In a greedy sense...
- Seam costs are not absolute
  - Consider HVV (how many possibly orderings?)
  - Cost(V) on HV not necessarily equal Cost(V) on VH
  - But we keep track of only one:  $\min(HV, VH)\dots$

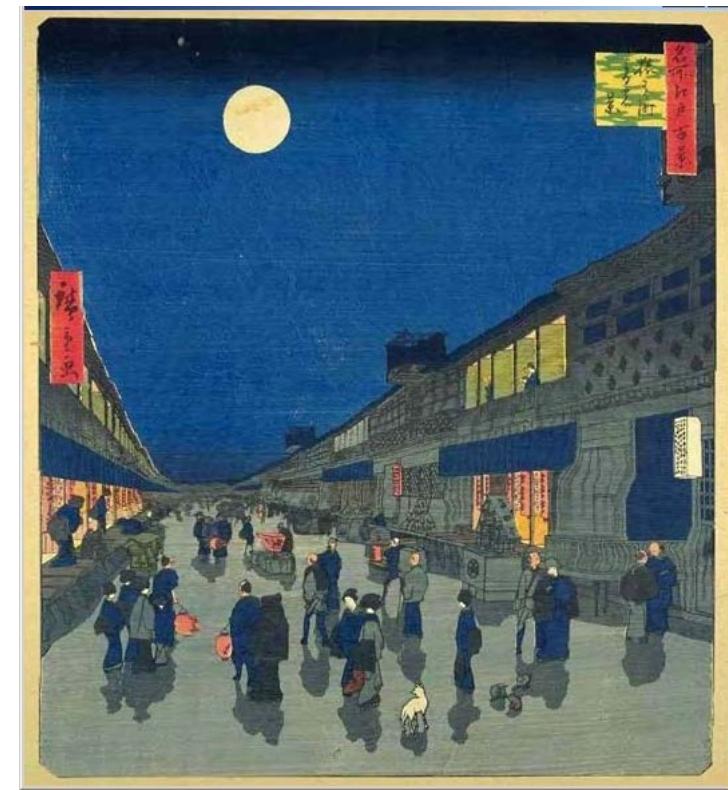
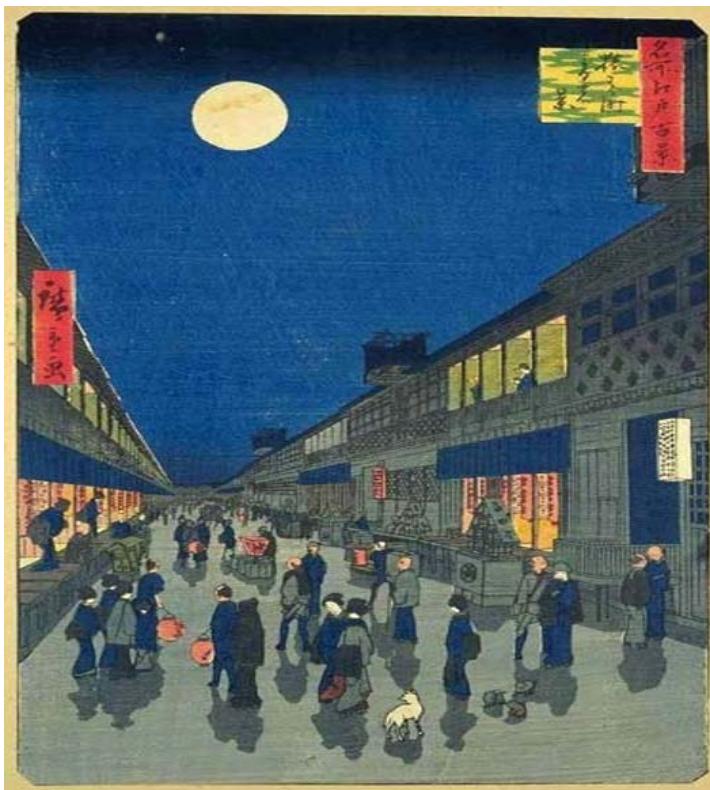
Horizontal Seams		Vertical Seams
0	V	VV
H	$\min(HV, VH)$	?



# Image Expansion (Synthesis)

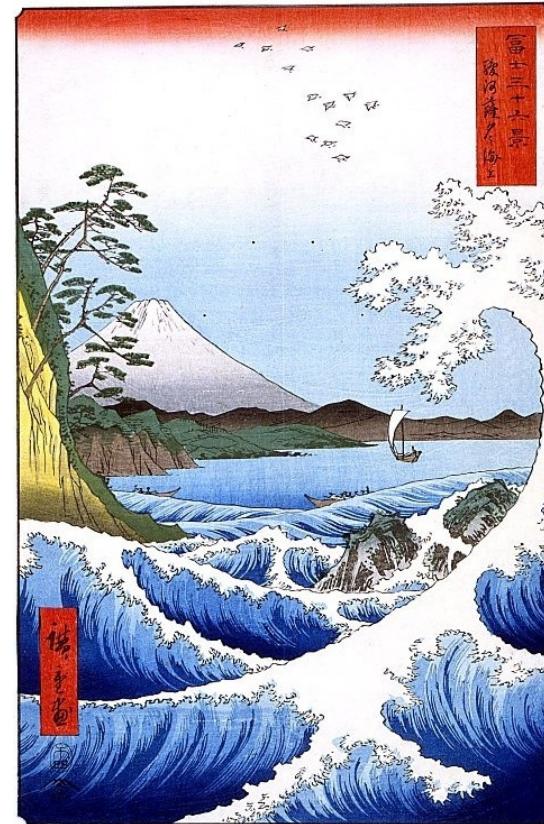


# Image Expansion – take 2



Scaling

# Enlarged or Reduced?



# Combined Insert and Remove



*Insert & remove seams*



*Scaling*



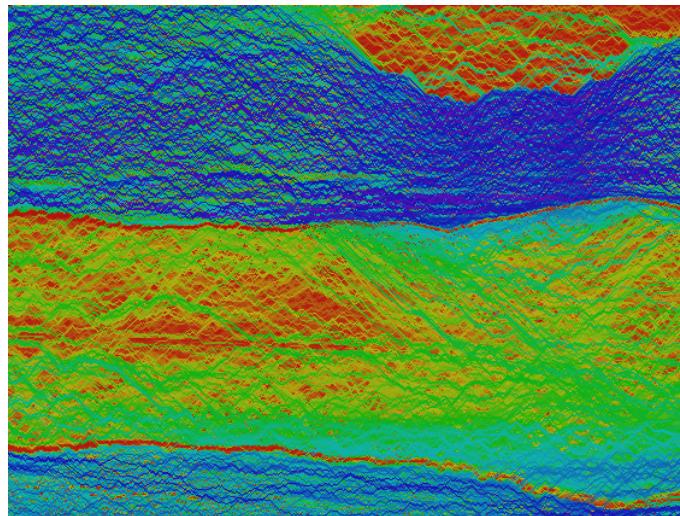
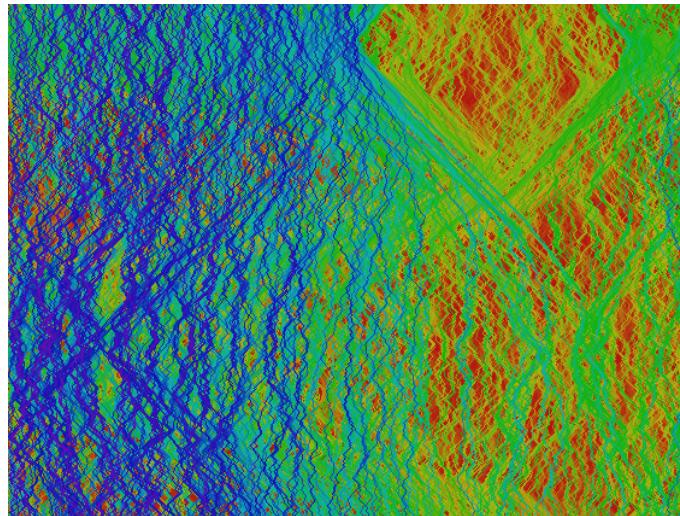
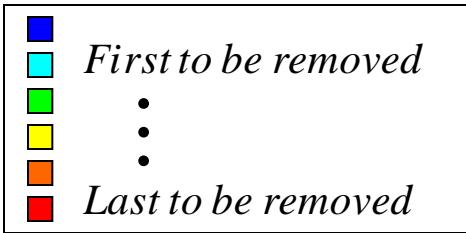
# Questions?



# 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

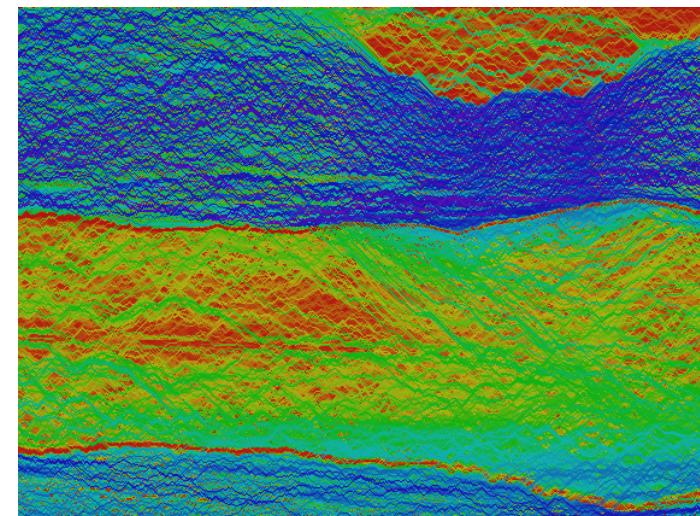




# Multi-Size Image Representation



+





# Multi-Size Image Representation





# 2D Multi-Size Representation?

- **Consistent index maps (see appendix in their paper)**
- **Alternatives**
  - Use seams in one direction, row/column seams in the other direction
  - Compute seams in one direction and use them to constrain seams in the other direction
    - If you can do that you'll have a nice publication 😊



# Auxiliary Energy

- Recall our seam equation

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

# With face detector





# With User Constraints



# Object Removal



# Object Removal



Input

Retargeted

Pigeon Removed

Girl Removed



# Find the Missing Shoe!



# Solution



# Limitations

## Content



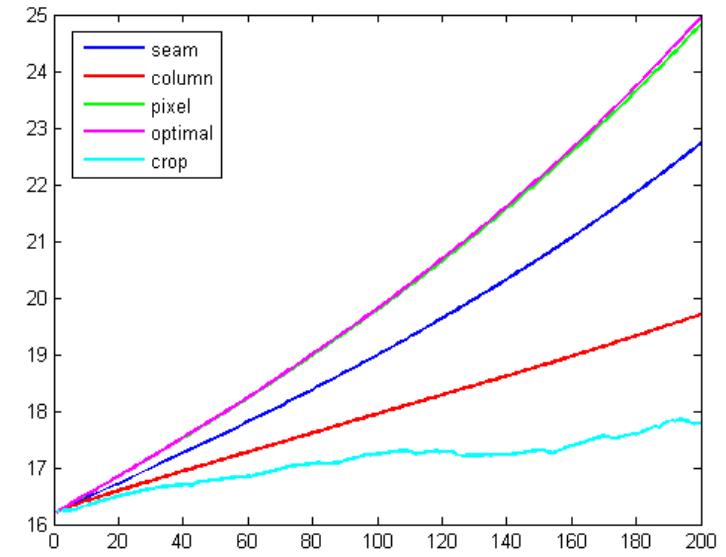
## Structure



# Preserved Energy - Revisited



*Average*  
*Pixel*  
*Energy*



*Image Reduction* →



**crop**



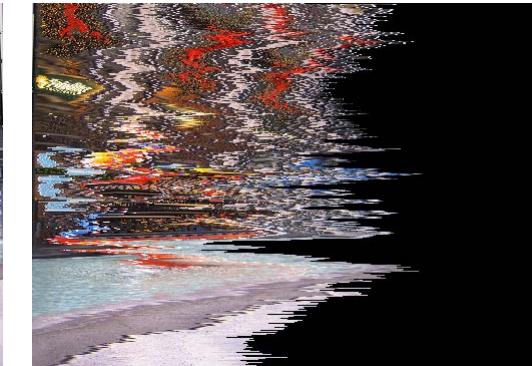
**column**



**seam**

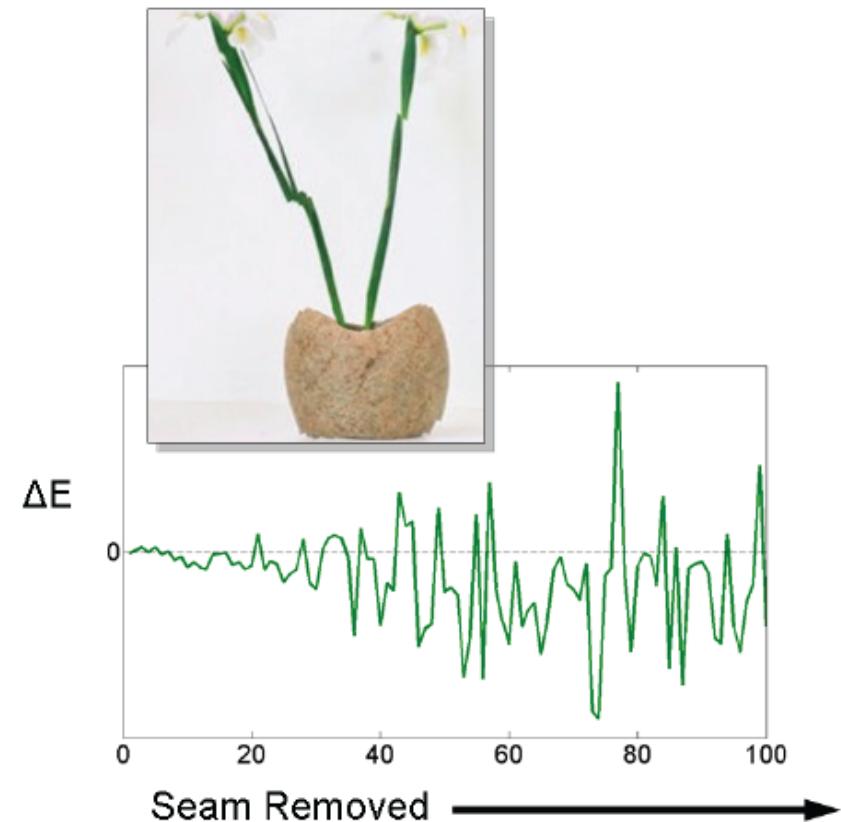
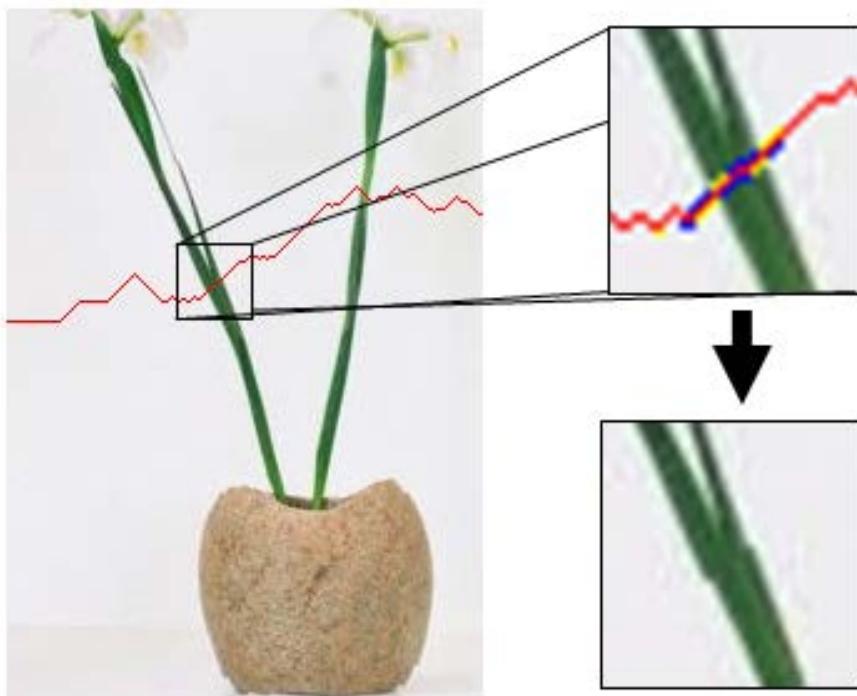


**pixel**



**optimal**

# Inserted Energy

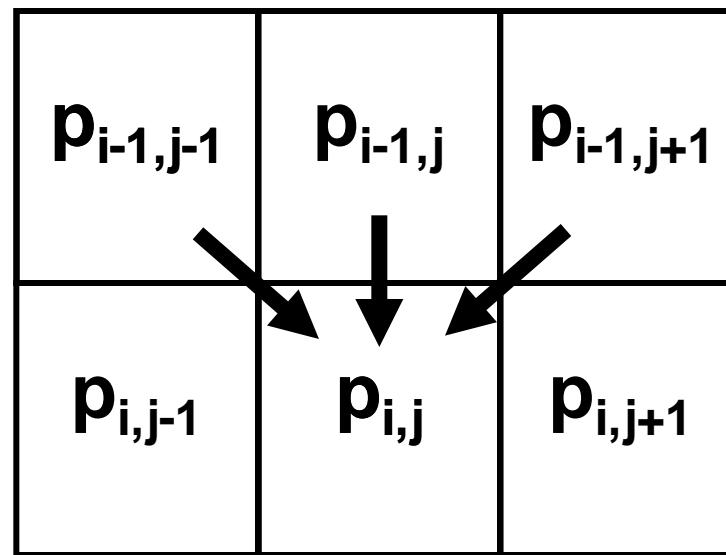




# Minimize Inserted Energy

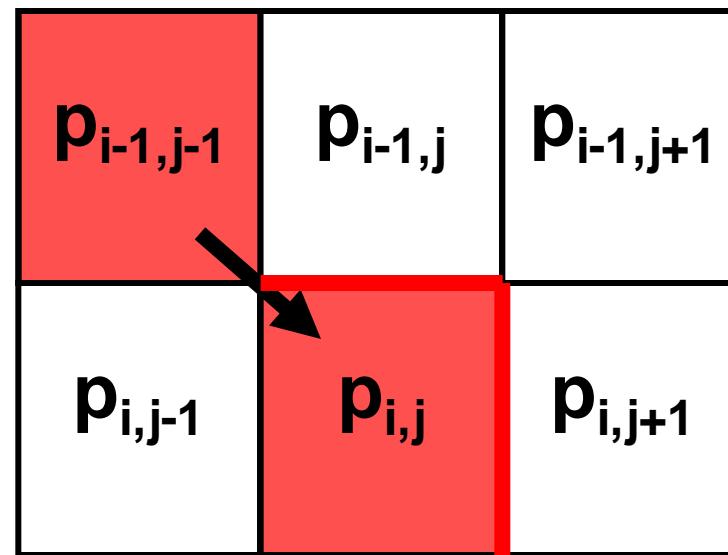
- Instead of removing the seam of least energy, remove the seam that *inserts the least energy* to the image !

# Tracking Inserted Energy



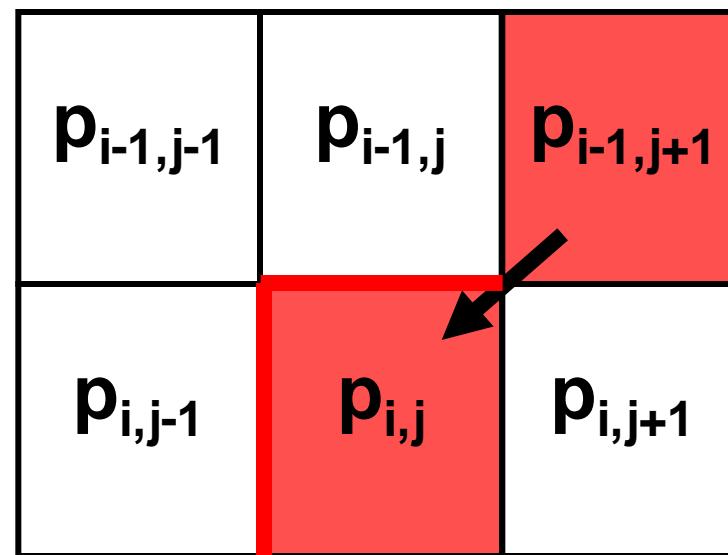
- Three possibilities when removing pixel  $P_{i,j}$

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



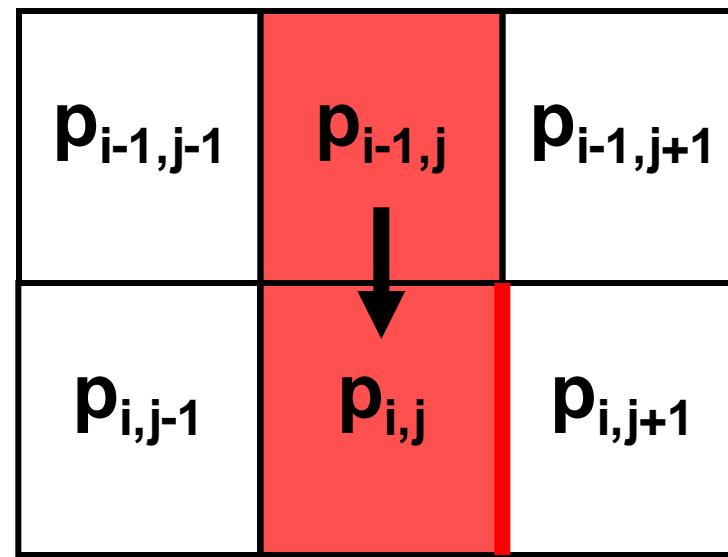
$$C_L(i, j) = |I(i, j + 1) - I(i, j - 1)| + |I(i - 1, j) - I(i, j - 1)|$$

# 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



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

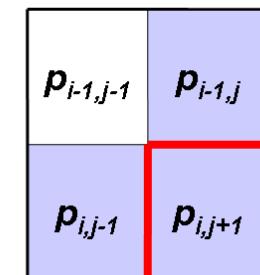
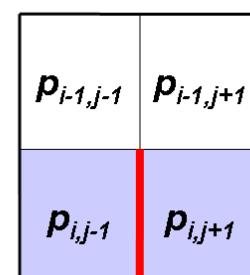
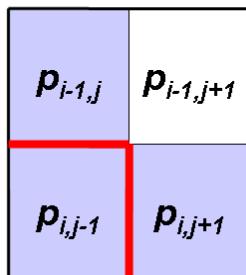
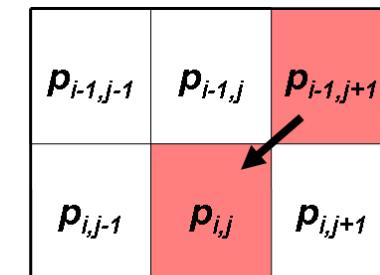
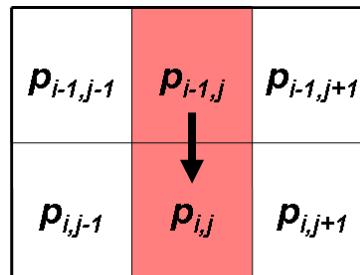
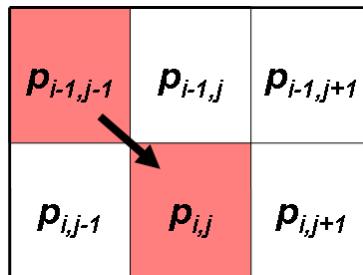


# Old “Backward” Energy

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

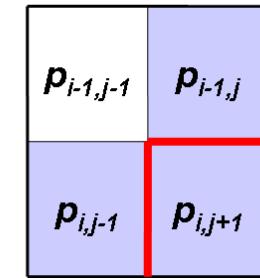
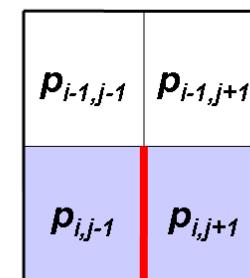
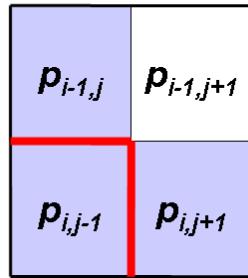
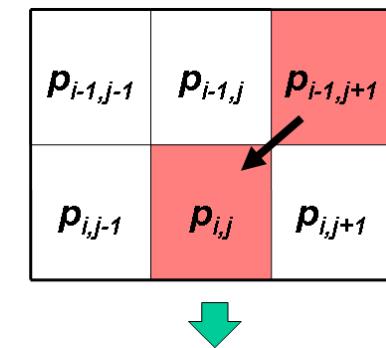
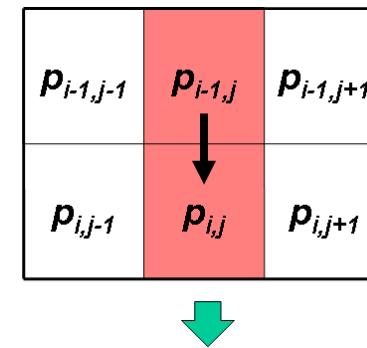
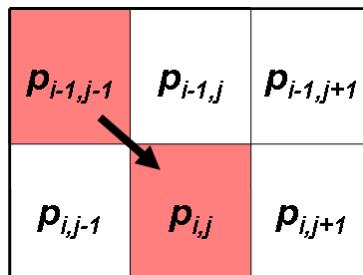
# New Forward Looking Energy

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



# 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



Input



Backward

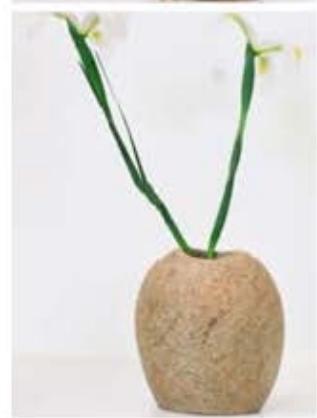
Forward



Input

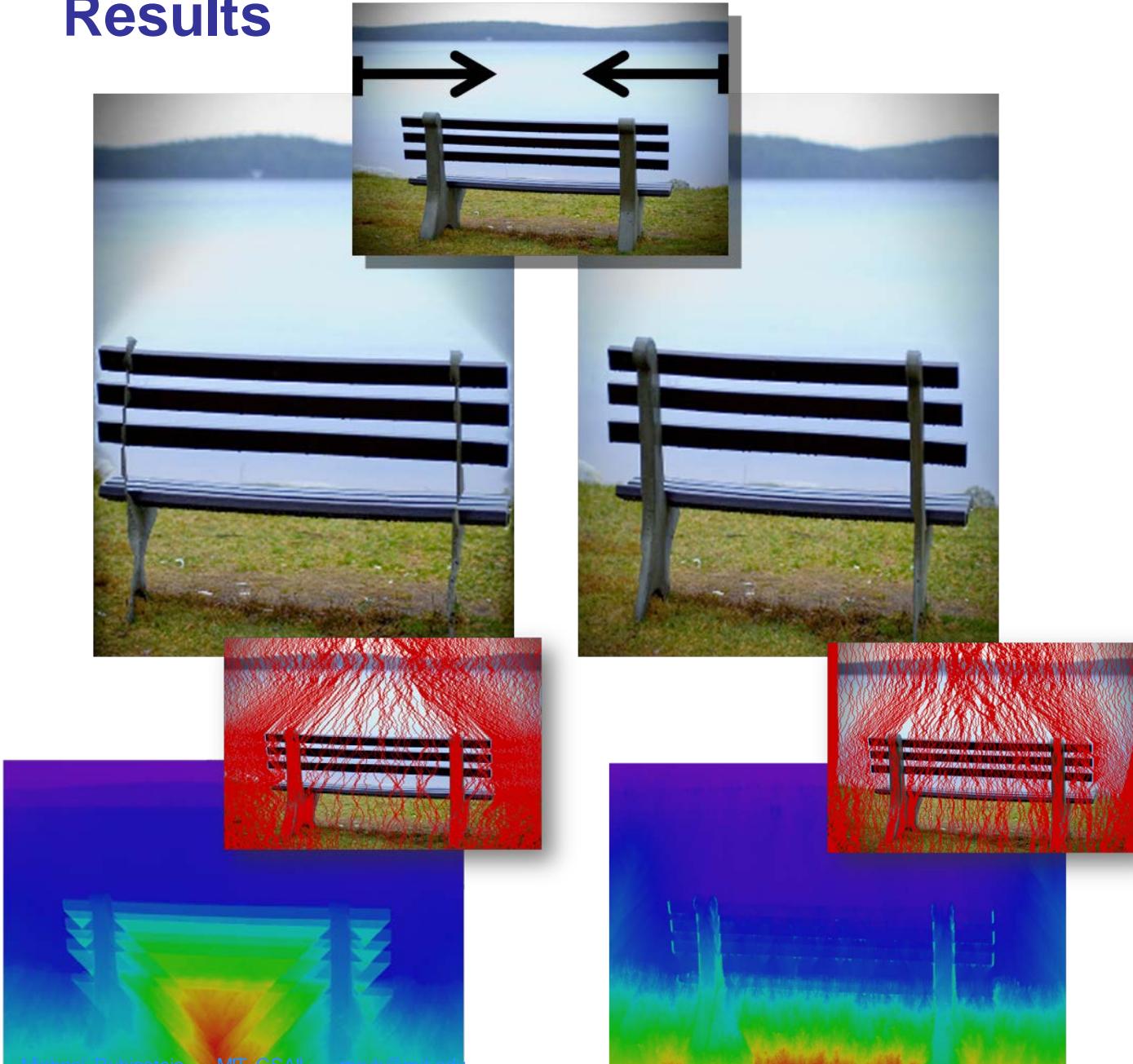


Backward



Forward

# Results



# Backward vs. Forward

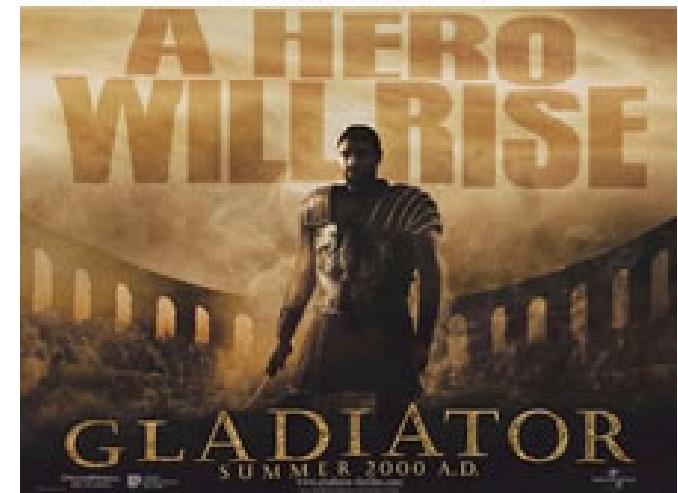
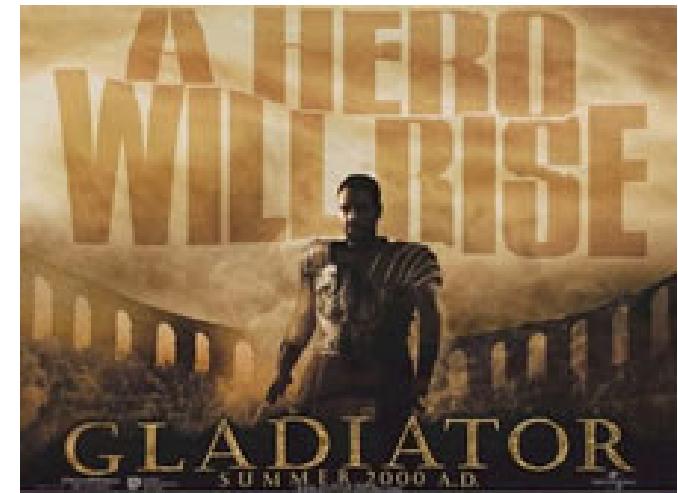
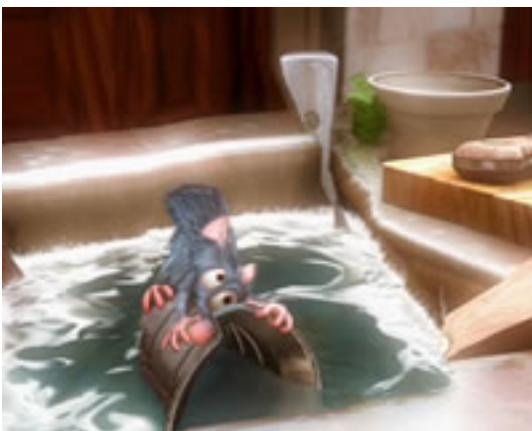


Backward



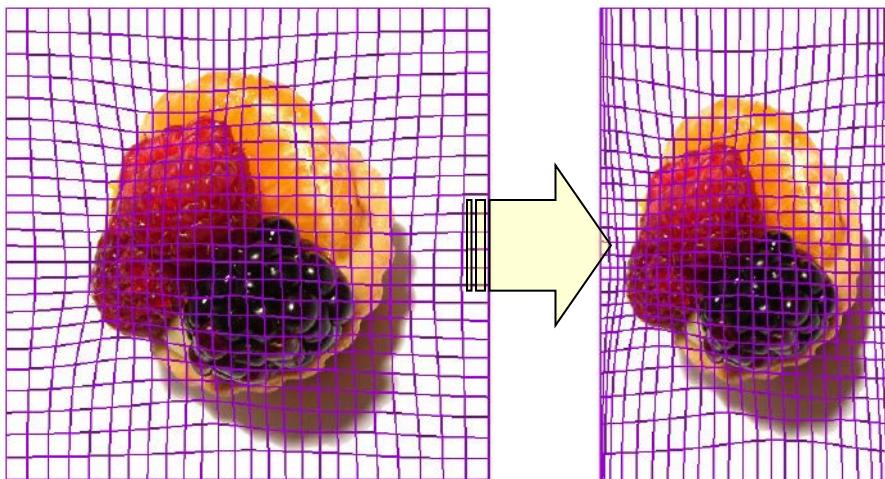
Forward

# Results



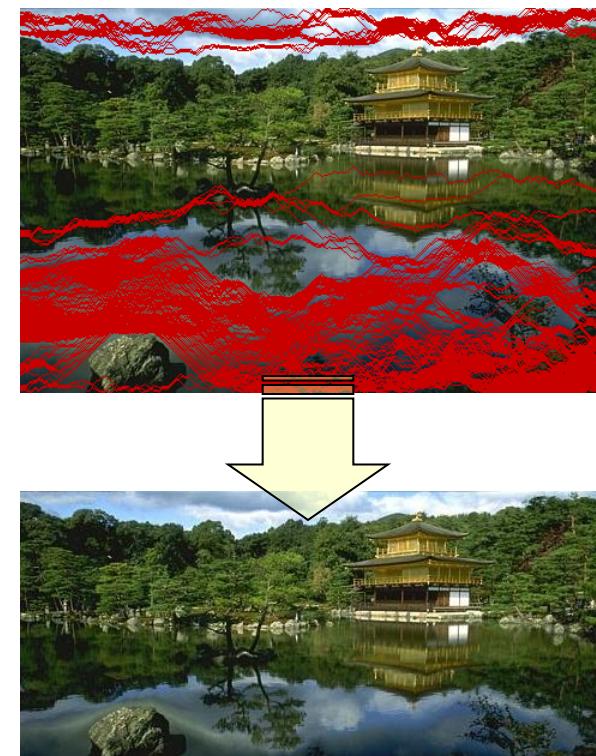
# Discrete vs. Continuous

[Wang et al 2008]



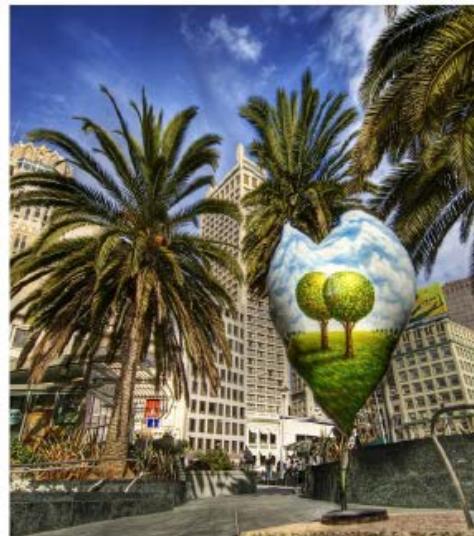
**Continuous**

[Avidan and Shamir 2007]



**Discrete**

# Non-homogeneous Scaling

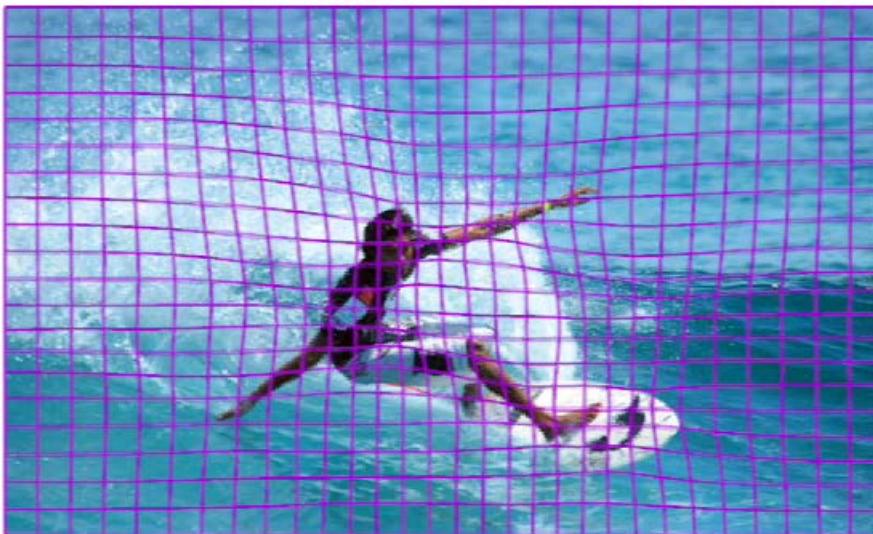


Seam Carving



Scale-and-Stretch

# Non-homogeneous Scaling



[Wang et al 2008]

# Bidirectional Similarity [Simakov et al. CVPR 08]

$$d(S, T) = \overbrace{\frac{1}{N_S} \sum_{P \subset S} \min_{Q \subset T} D(P, Q)}^{d_{\text{complete}}(S, T)} + \overbrace{\frac{1}{N_T} \sum_{Q \subset T} \min_{P \subset S} D(Q, P)}^{d_{\text{cohere}}(S, T)}$$

Source



**complete  
and  
coherent**



~~complete~~  
~~coherent~~



~~complete~~  
~~coherent~~



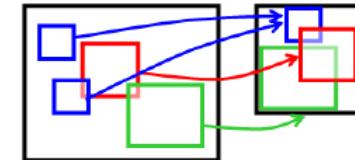
$d_{\text{complete}}(S, T)$

$d_{\text{cohere}}(S, T)$

Completeness:

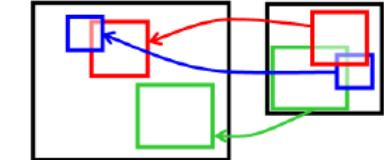
*(a) The bidirectional spatial (image) similarity:*

Input image  
(source)      Output image  
(target)



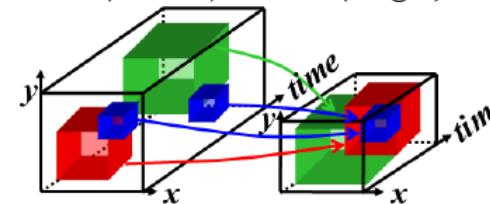
Coherence:

Input image  
(source)      Output image  
(target)

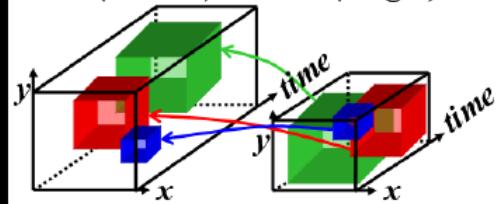


*(b) The bidirectional space-time (video) similarity:*

Input video  
(source)      Output video  
(target)

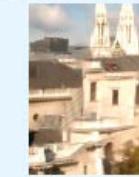


Input video  
(source)      Output video  
(target)





# Bidirectional Similarity





# From Images to Videos

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

## 1. Cardinality

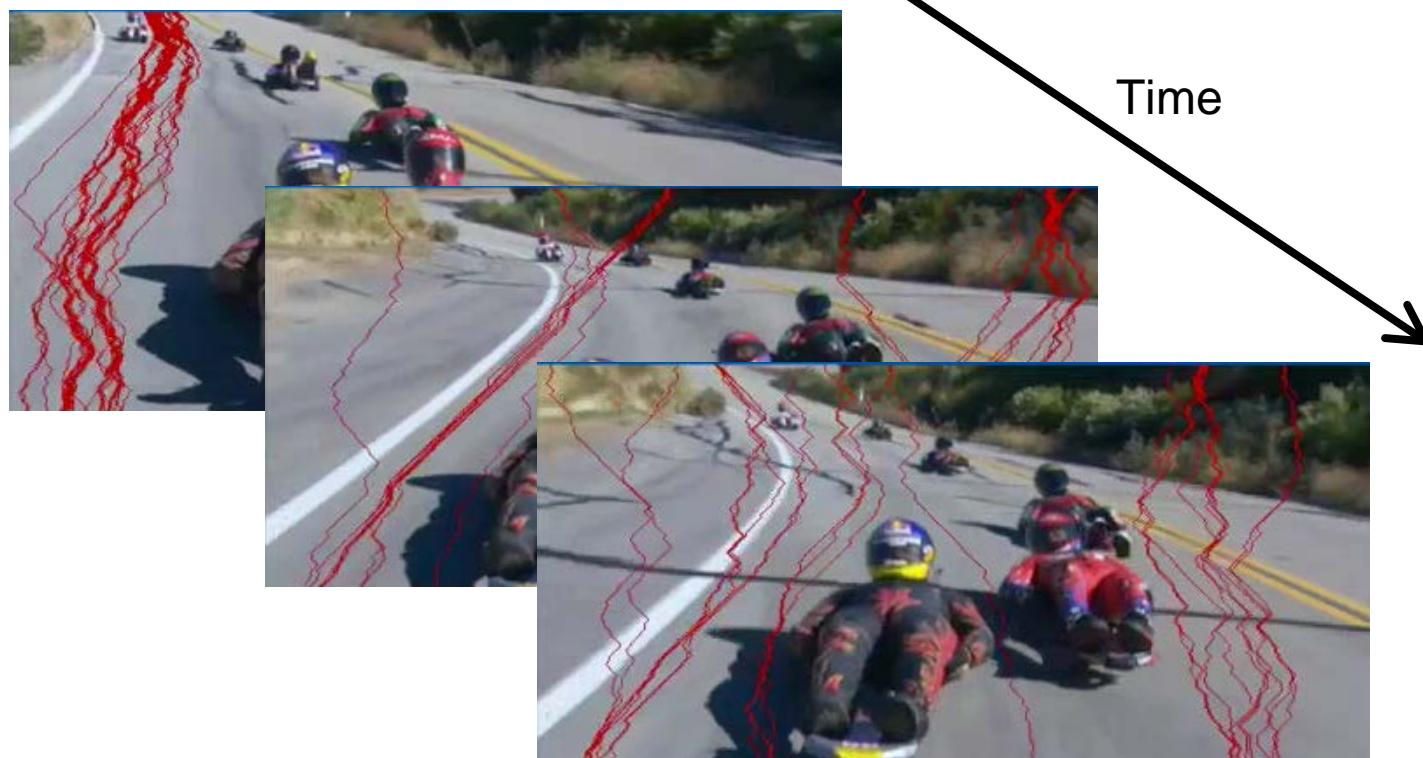
- Suppose 1min of video x 30 fps = 1800 frames
- Say your algorithm processes an image in 1 minute → **30 hours !!**

## 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

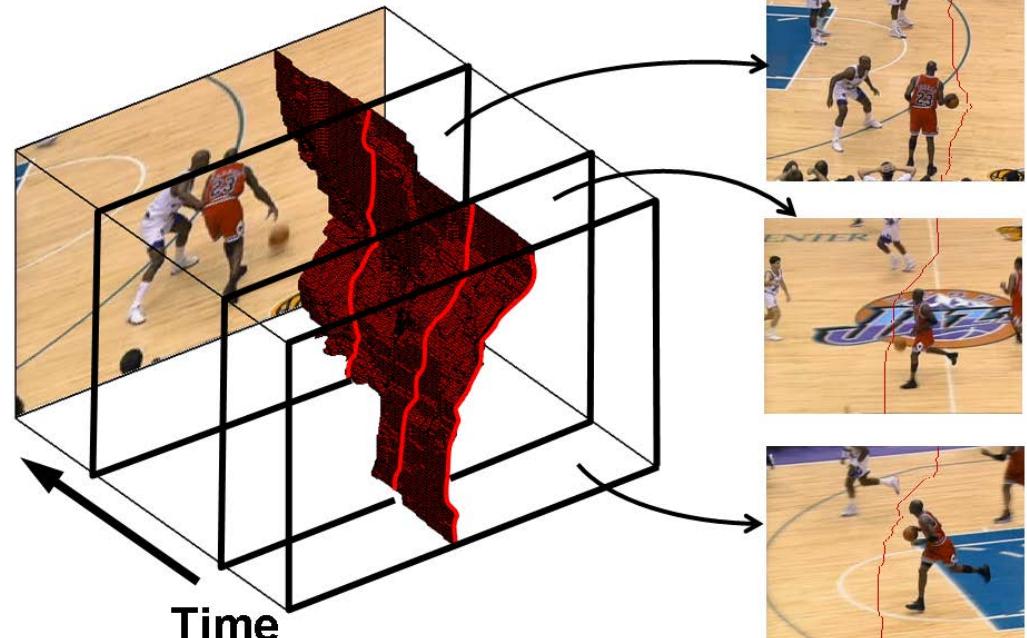
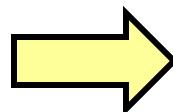




# From 2D to 3D



1D paths in images



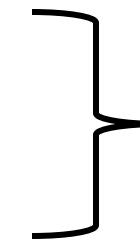
2D manifolds in video cubes

# Challenges

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

- **Cut must fulfill seam constraints**

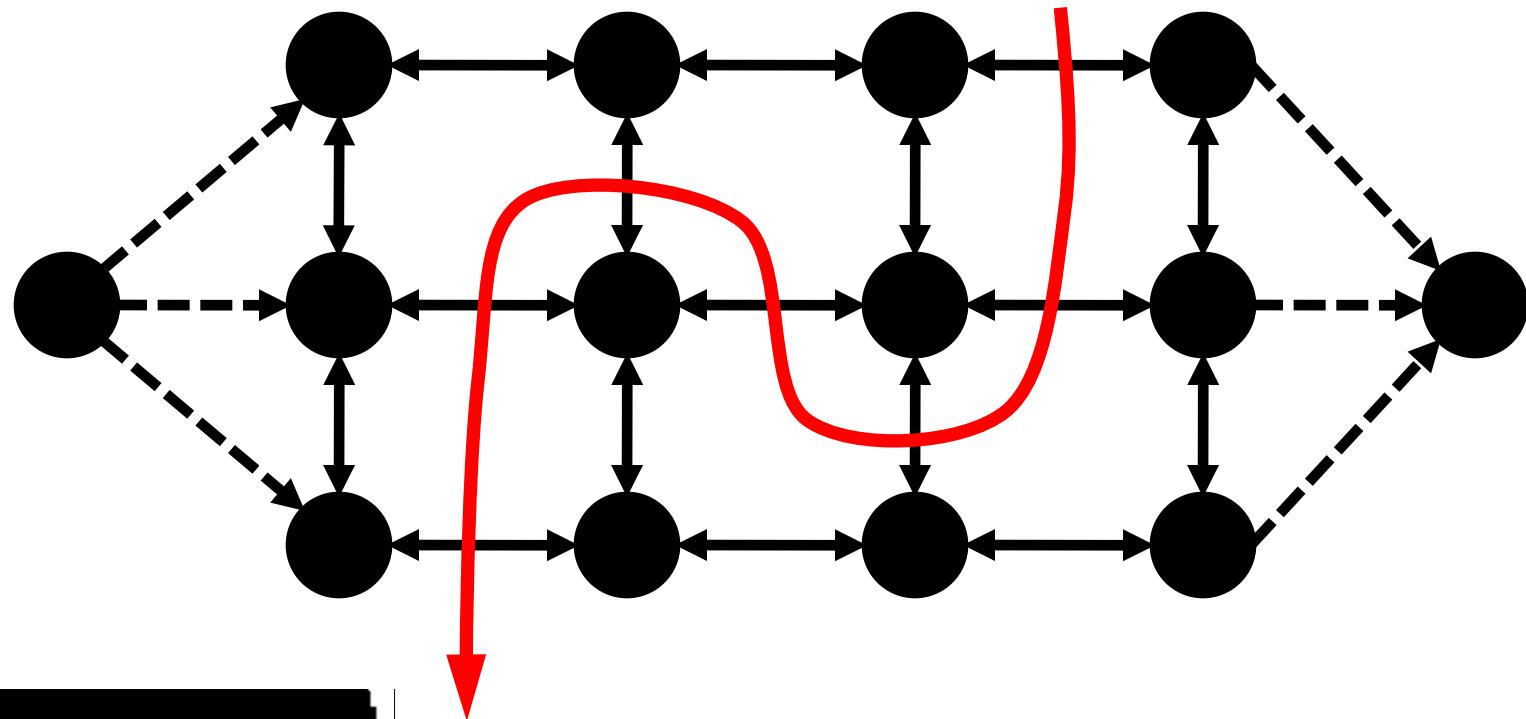
1. *Monotonic* (cut each row exactly once)
2. Connected



Cut should be a function!



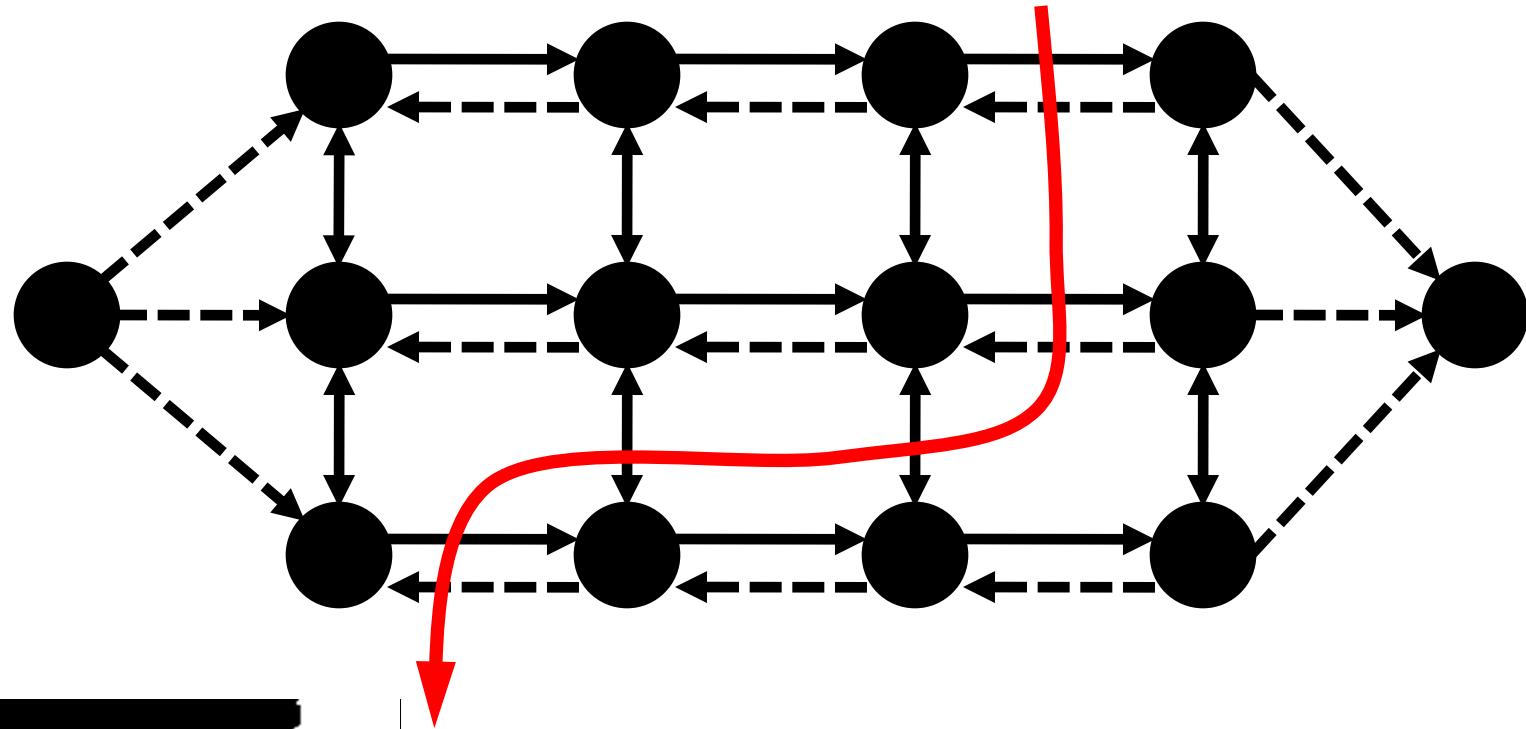
# Graph Construction



Not monotonic

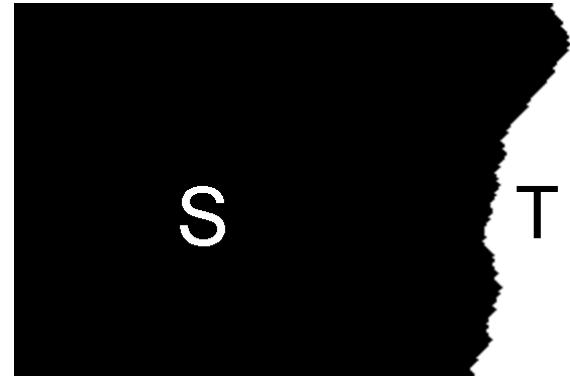
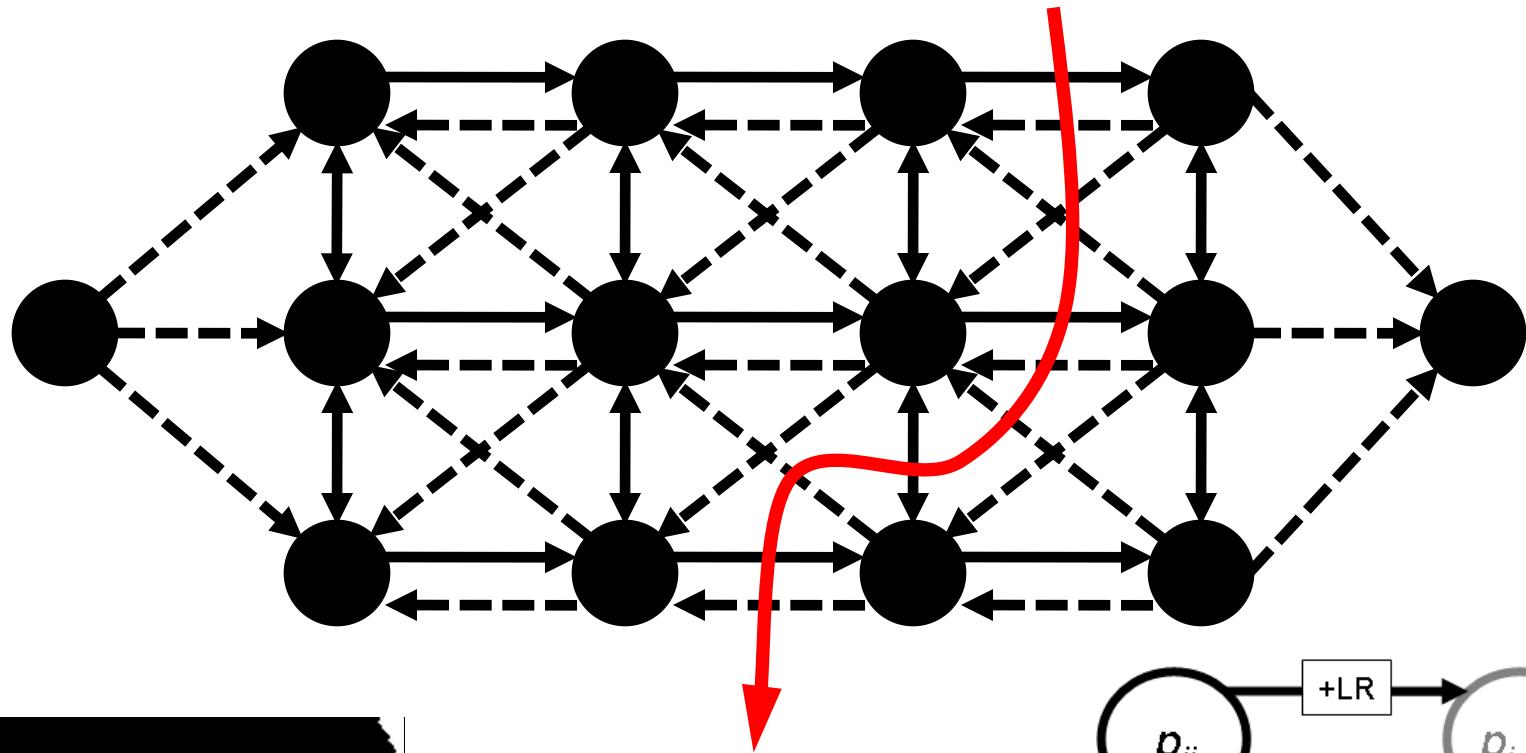


# Graph Construction

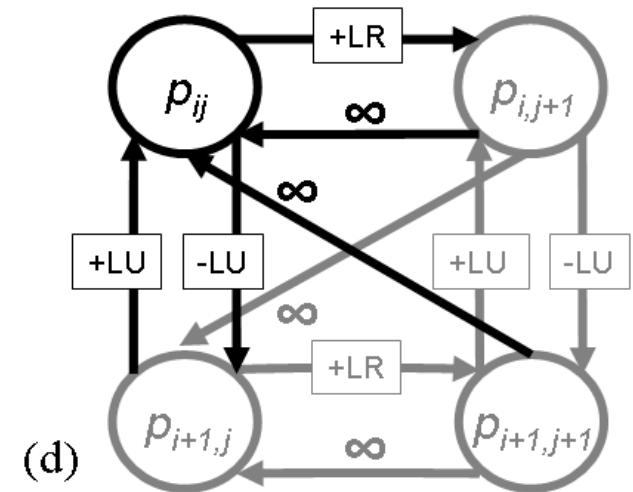


Monotonic, not connected

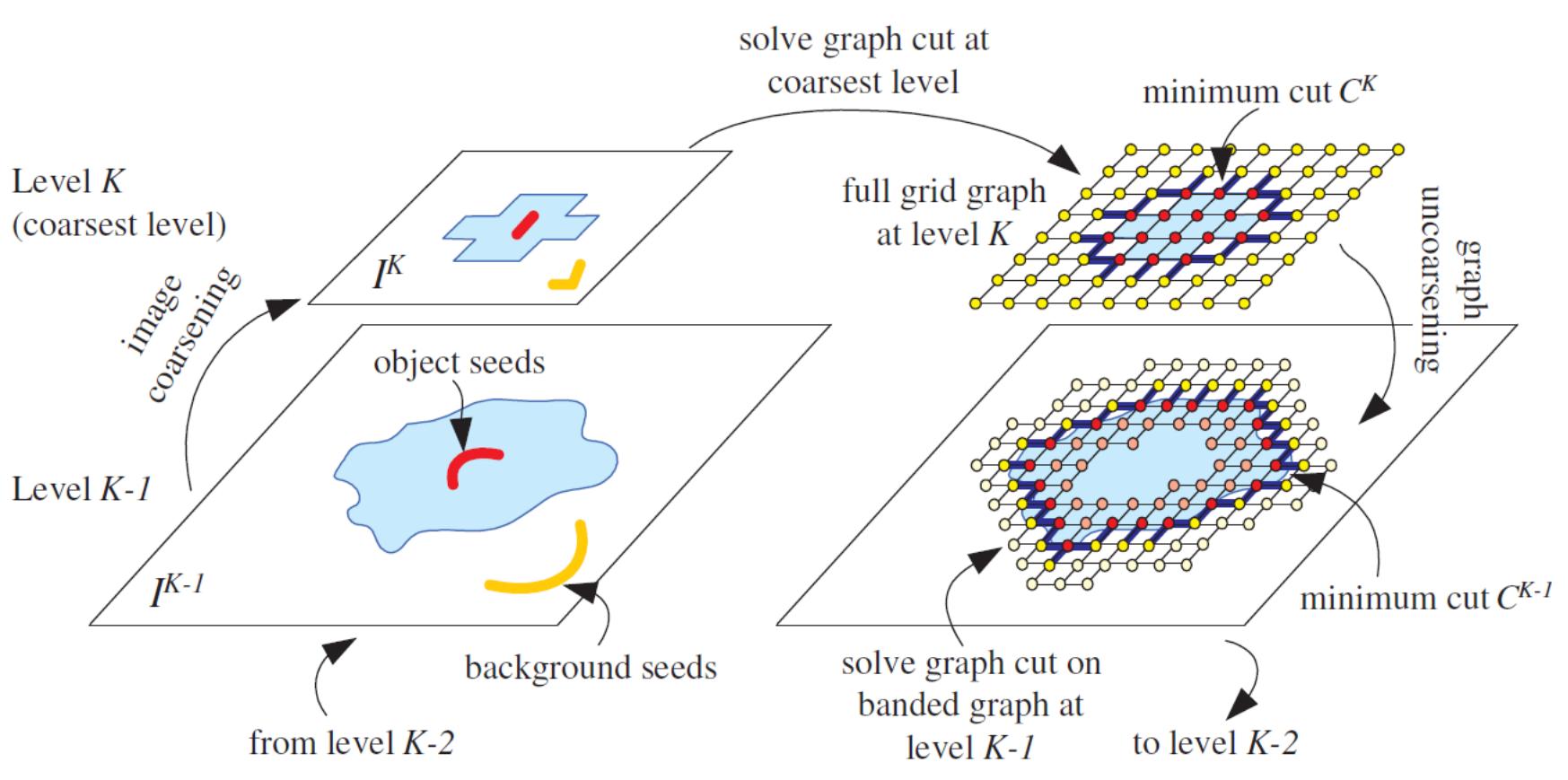
# Graph Construction



Backward  
Energy



# Multiresolution Graph Cut



**Figure 1. Multilevel banded graph cuts algorithm**

Lombaert et al. [2005]

# Results

Rescale



Retarget



# Results



Backward



Forward



Seams

A comparison of three photographs showing the same group of people standing in a row on a green field. The first photograph is labeled "Backward" and shows the group from behind, with the letters "S" visible on the ground in front of each person. The second photograph is labeled "Forward" and shows the group from the front, with the letters "S" visible on the ground in front of each person. The third photograph is labeled "Seams" and shows the group from the front with red lines overlaid, highlighting the seams or boundaries between the different people in the group.

# Multi-size Video



674

3

# Object Removal...





# 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](http://help.adobe.com/en_US/Photoshop/11.0/WS6F81C45F-2AC0-4685-8FFD-DBA374BF21CD.html)

# 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

