Lecture 10:
2D Transformation & Alignment

COMP 590/776: Computer Vision

Instructor: Soumyadip (Roni) Sengupta

TA: Mykhailo (Misha) Shvets

Course Website:
Scan Me!

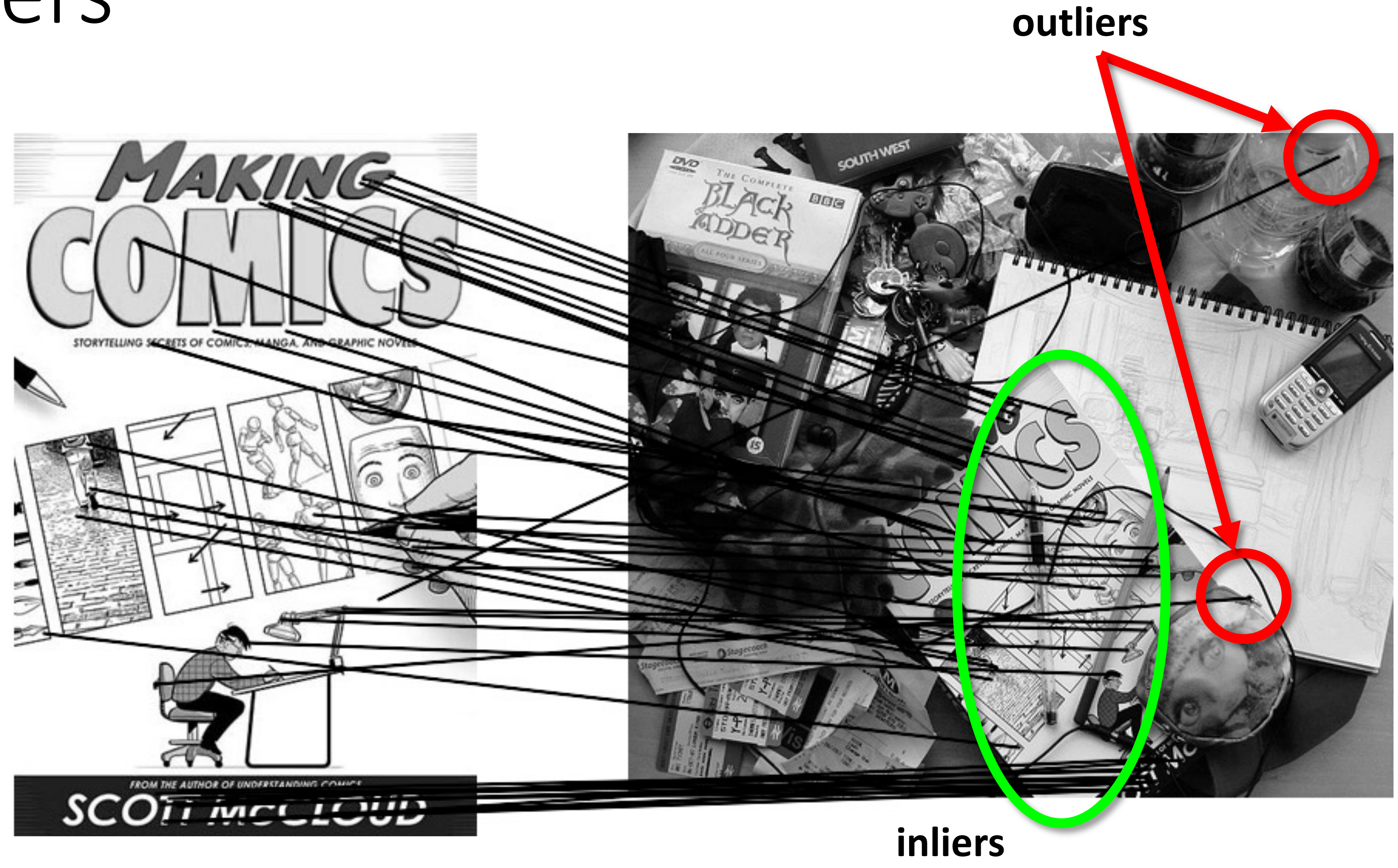
Today's class

- Fitting with outliers – RANSAC
- Warping
- Blending
- HW3 Motivation

Today's class

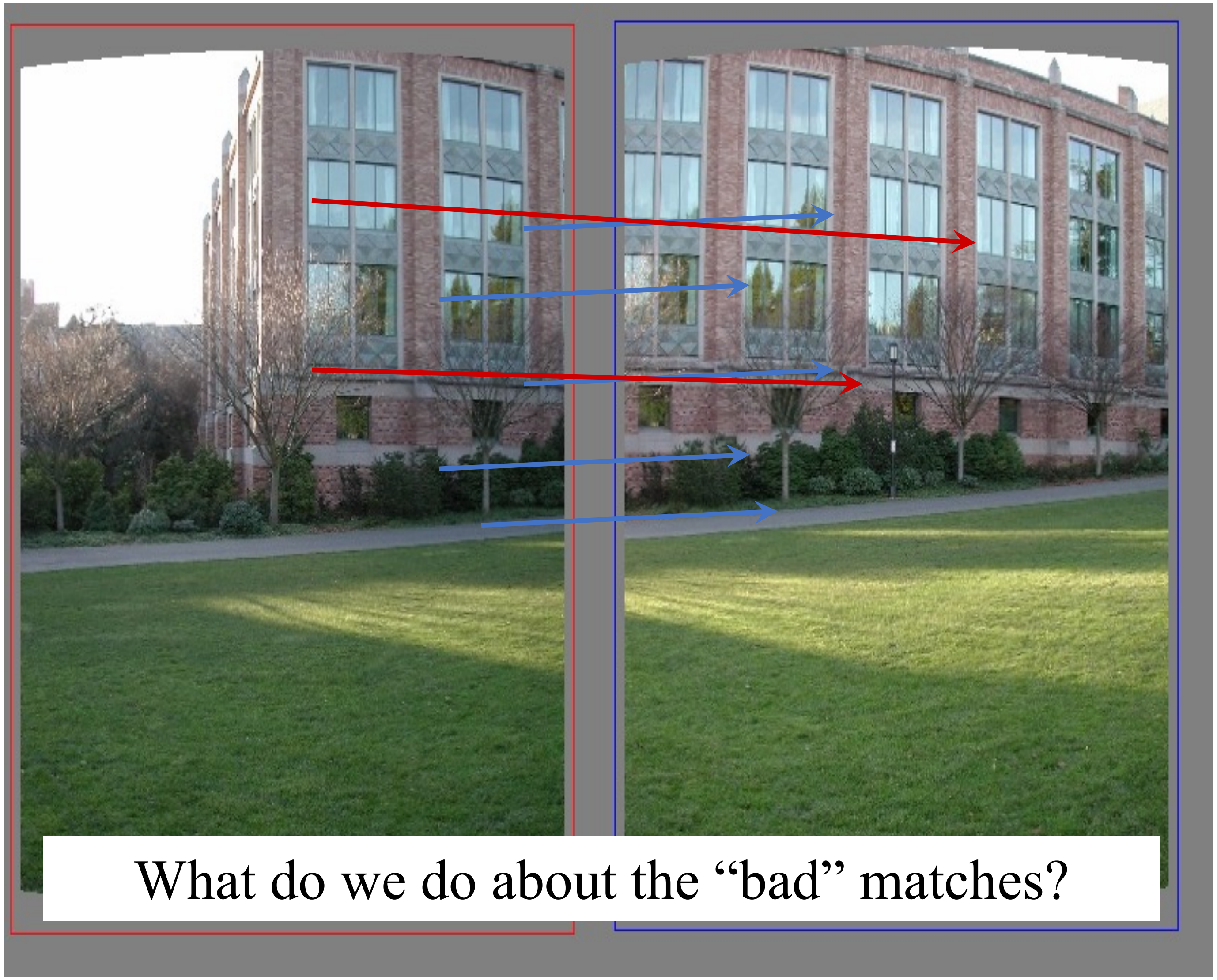
- Fitting with outliers – RANSAC
- Warping
- Blending
- HW3 Motivation

Outliers



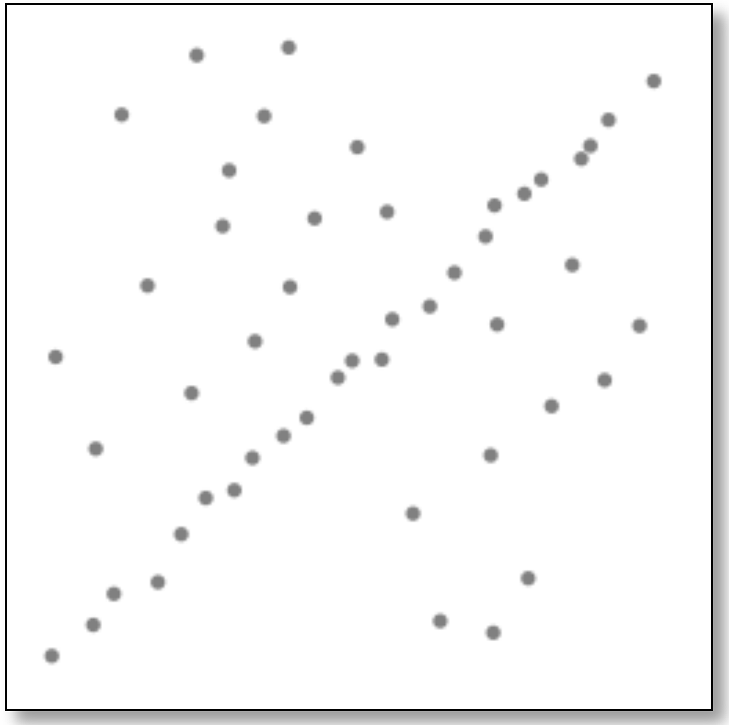
outliers
MAKING
COMICS
STORYTELLING SECRETS OF COMICS, MANGA, AND GRAPHIC NOVELS
FROM THE AUTHOR OF UNDERSTANDING COMICS
SCOTT McCLOUD
DVD
THE COMPLETE
BLACK ADDER
BBC
SOUTH WEST
inliers

Matching features

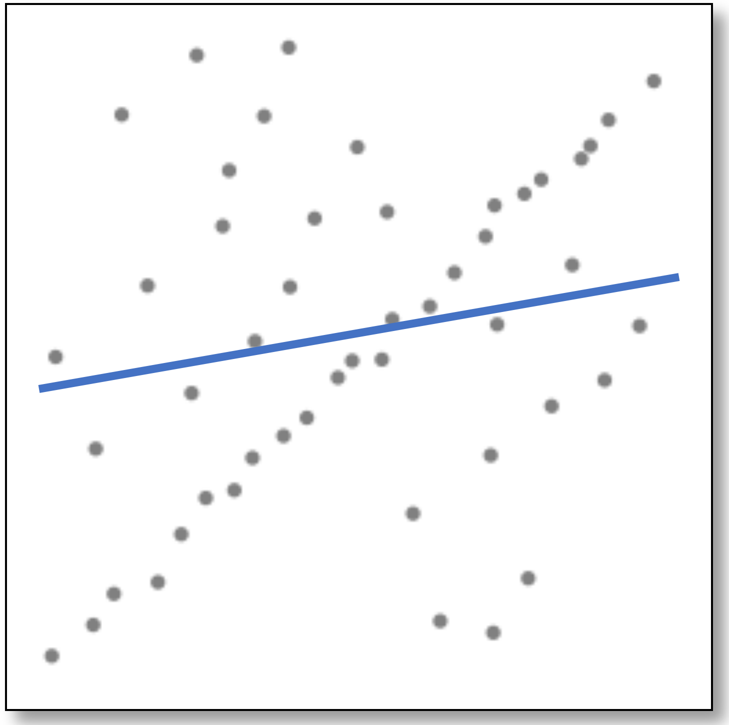


Robustness

- Let’s consider the problem of linear regression



Problem: Fit a line to these datapoints



Least squares fit

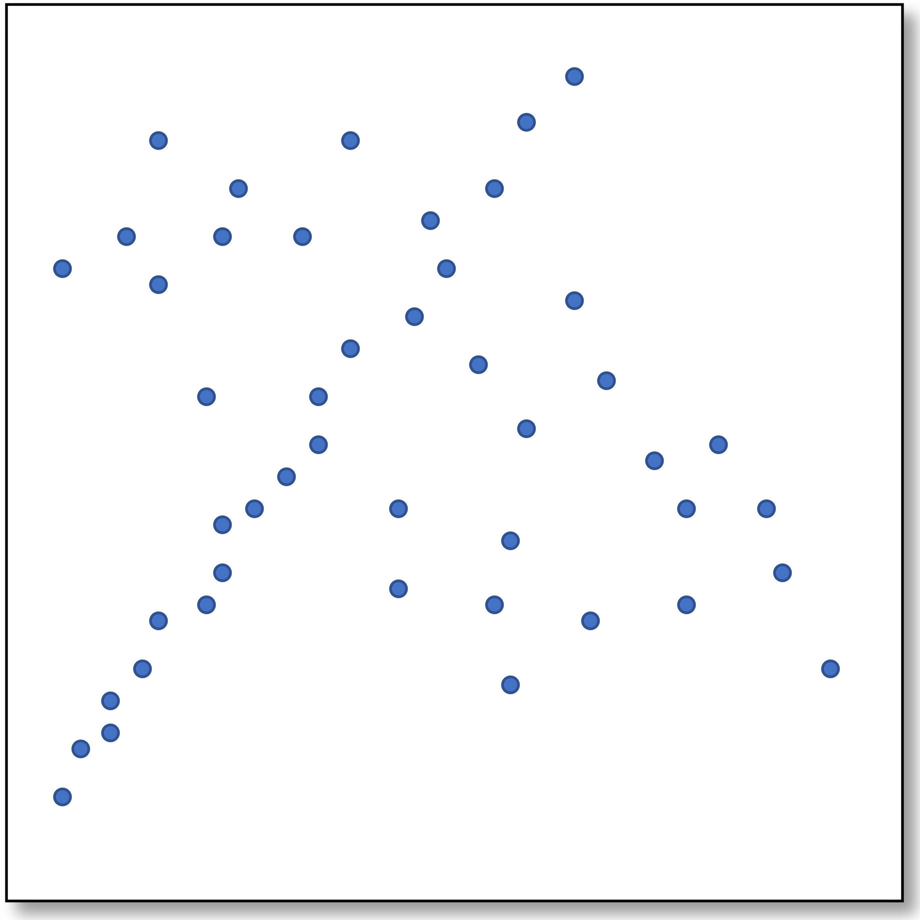
- How can we fix this?

Idea

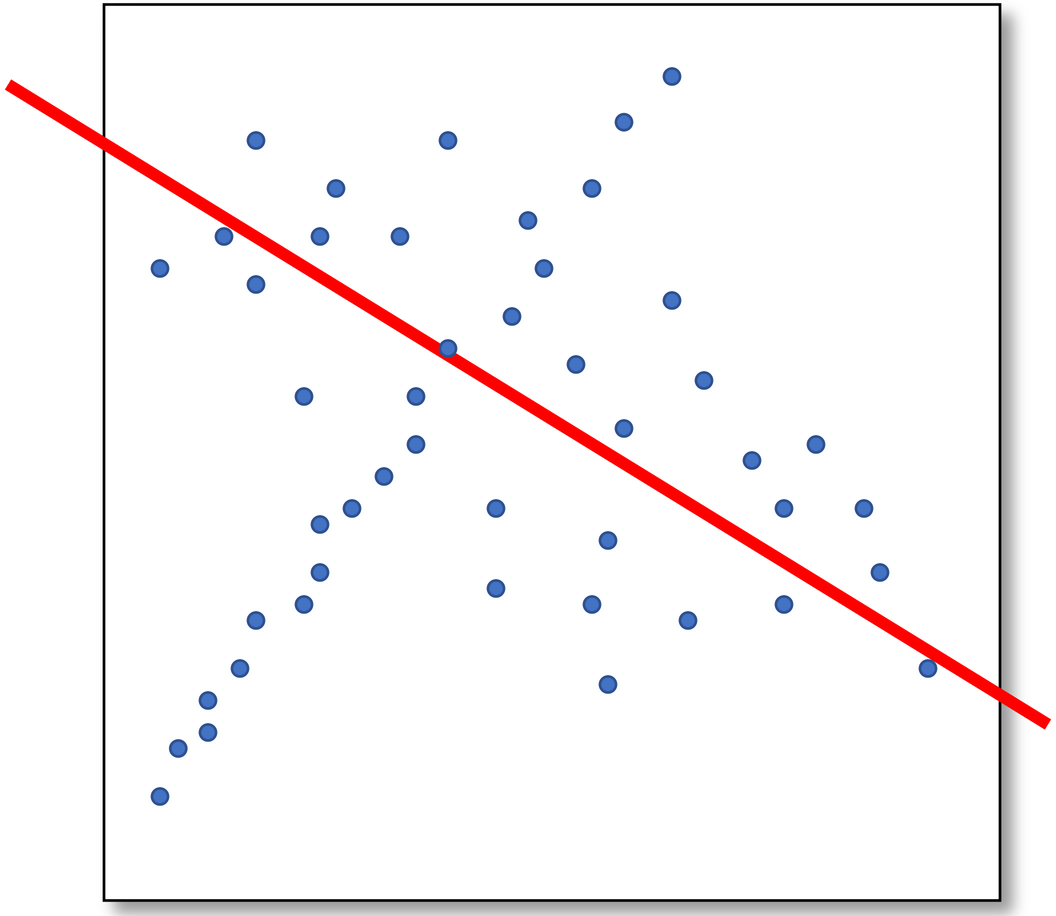
- Given a hypothesized line
- Count the number of points that “agree” with the line
  - “Agree” = within a small distance of the line
  - I.e., the **inliers** to that line

- For all possible lines, select the one with the largest number of inliers

Counting inliers

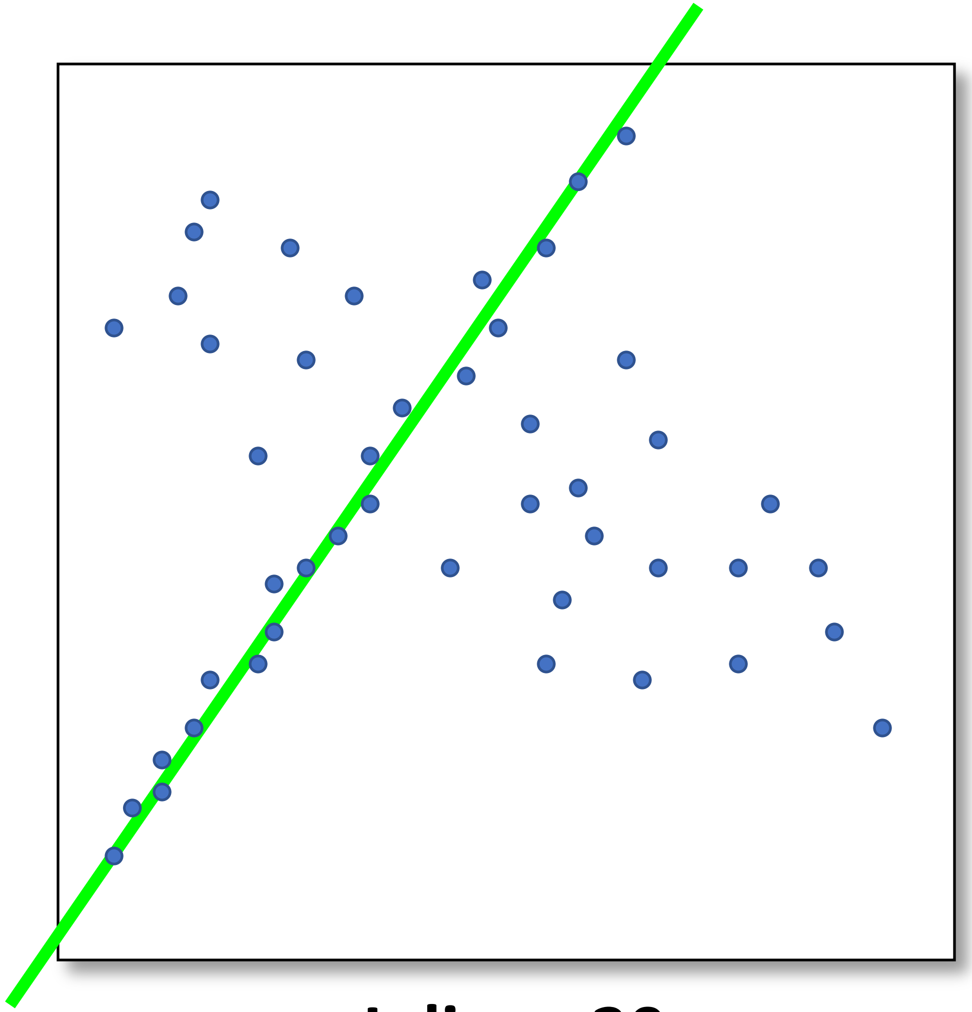


Counting inliers



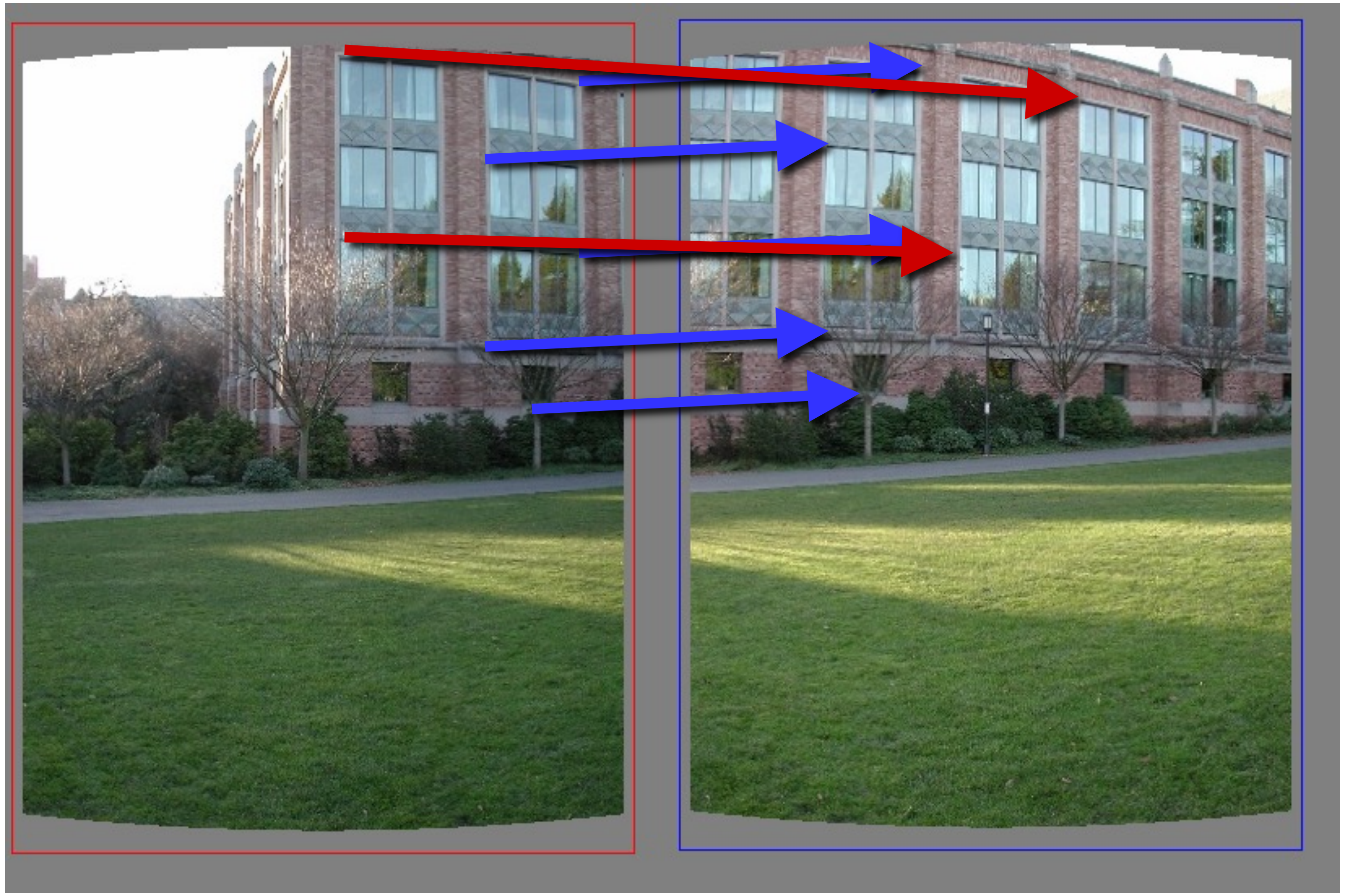
Inliers: 3

Counting inliers

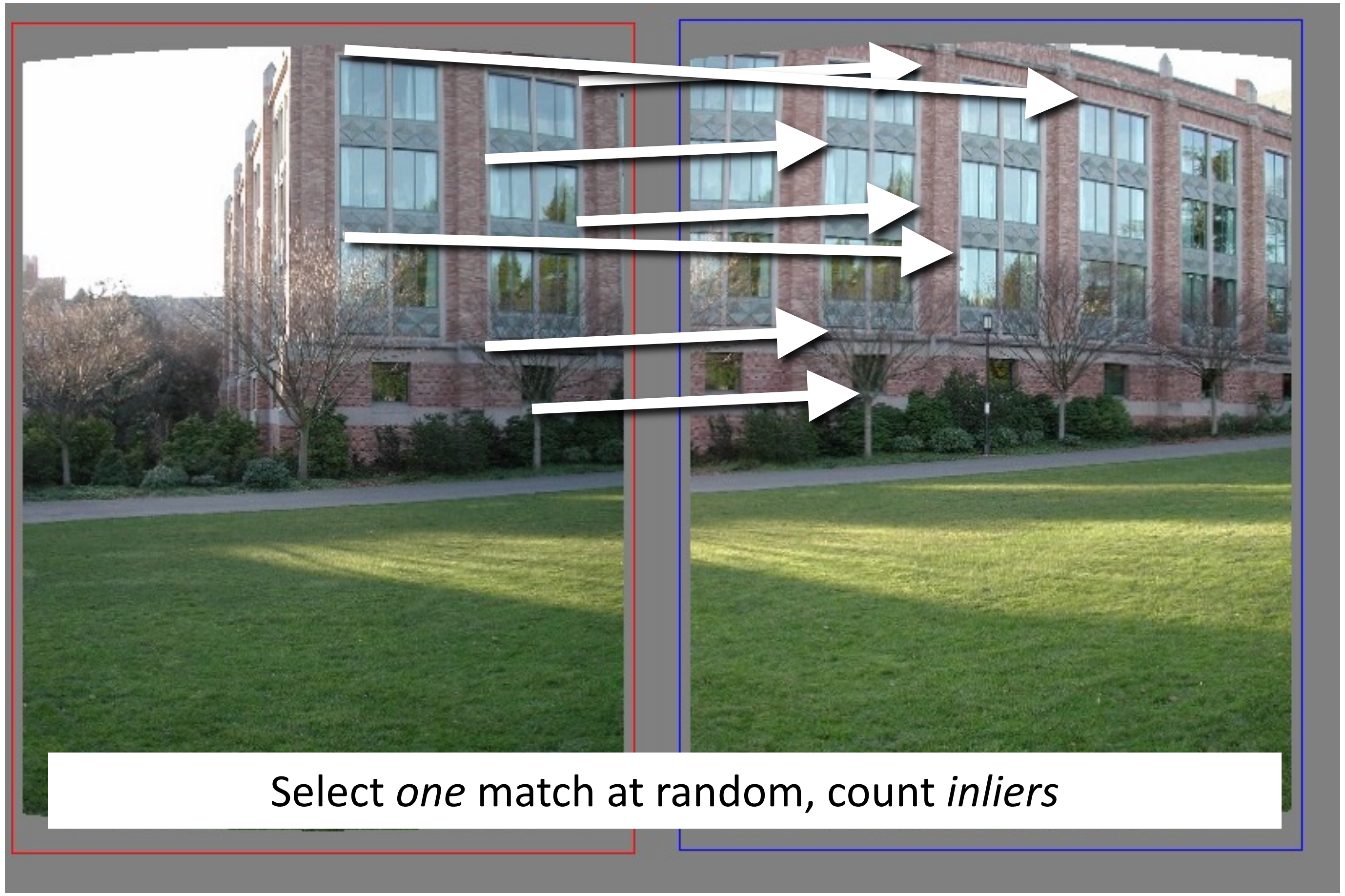


Inliers: 20

Translations

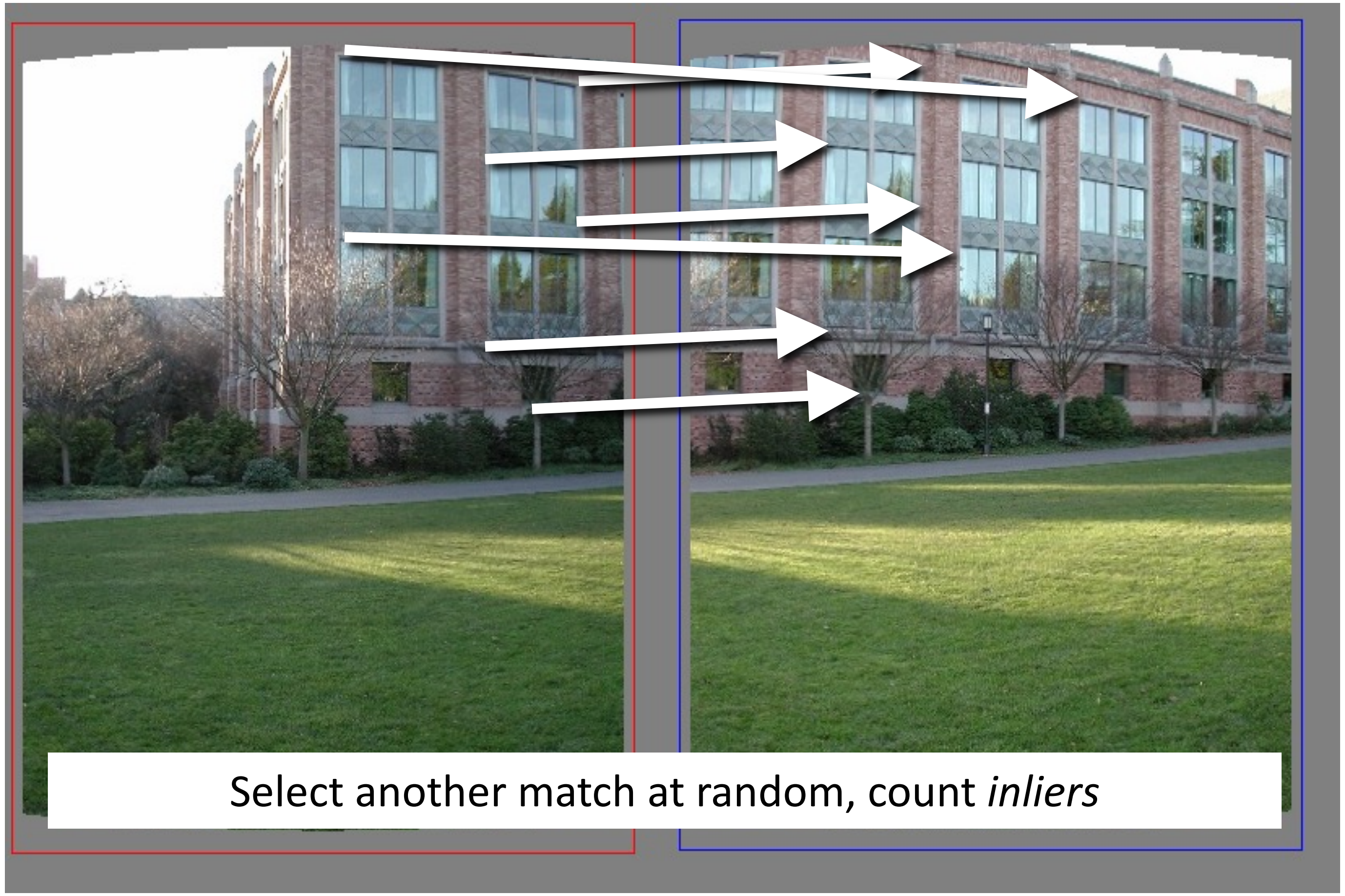


RAndom SAmple Consensus



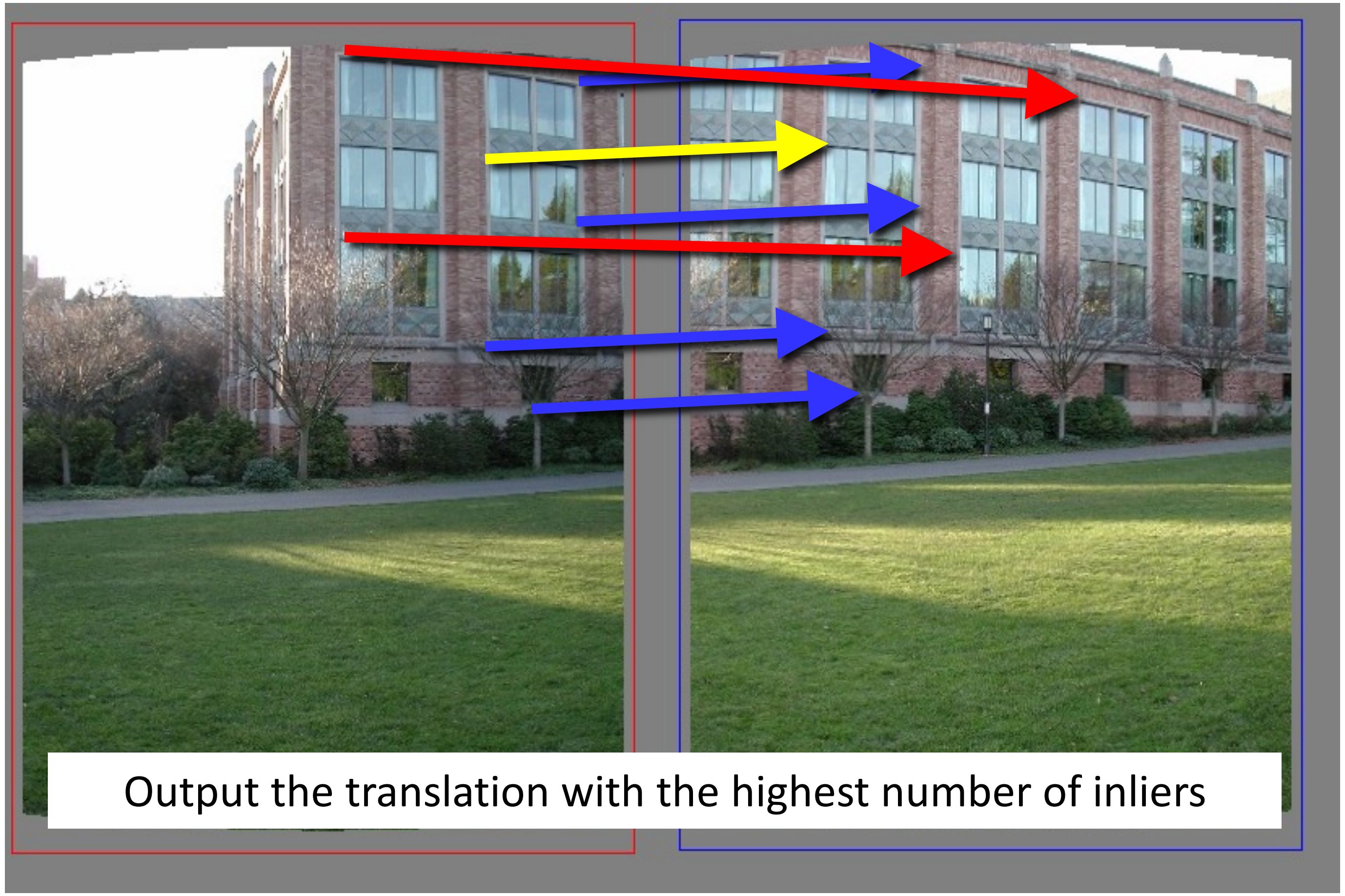
Select *one* match at random, count *inliers*

RAndom SAmple Consensus



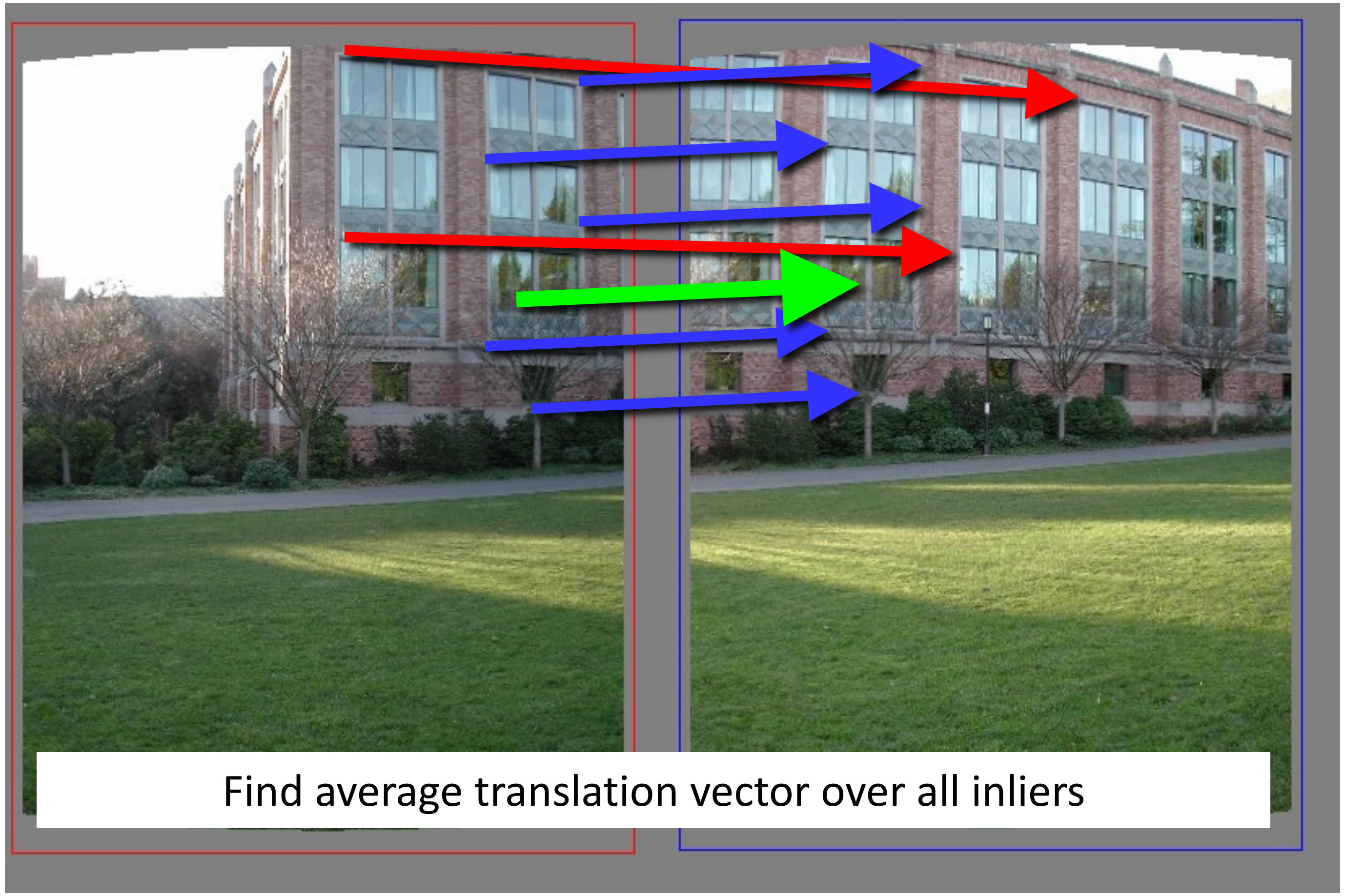
Select another match at random, count *inliers*

RAndom SAmple Consensus



Output the translation with the highest number of inliers

Final step: least squares fit



RANSAC

- Idea:
  - All the inliers will agree with each other on the translation vector; the (hopefully small) number of outliers will (hopefully) disagree with each other
    - RANSAC only has guarantees if there are < 50% outliers
  - "All good matches are alike; every bad match is bad in its own way."

    – Tolstoy via Alyosha Efros

RANSAC

- General version:
  1. Randomly choose *s* samples
     - Typically *s* = minimum sample size that lets you fit a model
  2. Fit a model (e.g., line) to those samples
  3. Count the number of inliers that approximately fit the model
  4. Repeat *N* times
  5. Choose the model that has the largest set of inliers

RANSAC for estimating homography

- RANSAC loop:

1. Select four feature pairs (at random)
2. Compute homography H (exact)
3. Compute *inliers* where $dist(p_i', \boldsymbol{H}\, p_i) < \varepsilon$
4. Keep largest set of inliers
5. Re-compute least-squares H estimate on all of the inliers

How many rounds?

- If we have to choose *s* samples each time
  - with an outlier ratio *e*
  - and we want the right answer with probability *p*

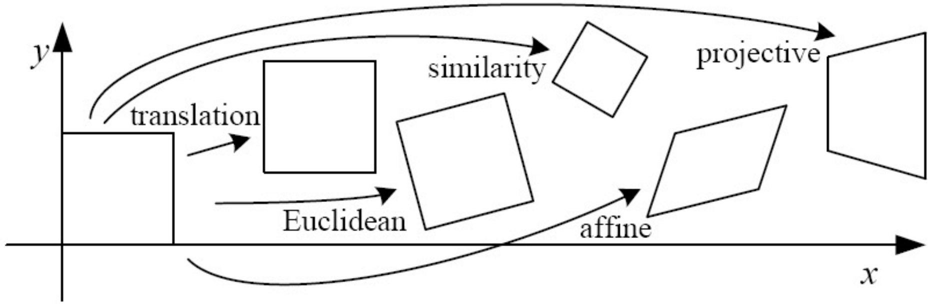
$$N \geq \frac{\log(1-p)}{\log(1-(1-e)^s)}$$

	proportion of outliers *e*						
s	5%	10%	20%	25%	30%	40%	50%
2	2	3	5	6	7	11	17
3	3	4	7	9	11	19	35
4	3	5	9	13	17	34	72
5	4	6	12	17	26	57	146
6	4	7	16	24	37	97	293
7	4	8	20	33	54	163	588
8	5	9	26	44	78	272	1177

$p = 0.99$

How big is *s*?

- For alignment, depends on the motion model
  - Here, each sample is a correspondence (pair of matching points)



Name	Matrix	# D.O.F.	Preserves:	Icon
translation	$\begin{bmatrix} \boldsymbol{I} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	2	orientation $+\cdots$	
rigid (Euclidean)	$\begin{bmatrix} \boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	3	lengths $+\cdots$	
similarity	$\begin{bmatrix} s\boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	4	angles $+\cdots$	
affine	$\begin{bmatrix} \boldsymbol{A} \end{bmatrix}_{2\times 3}$	6	parallelism $+\cdots$	
projective	$\begin{bmatrix} \tilde{\boldsymbol{H}} \end{bmatrix}_{3\times 3}$	8	straight lines	

RANSAC pros and cons

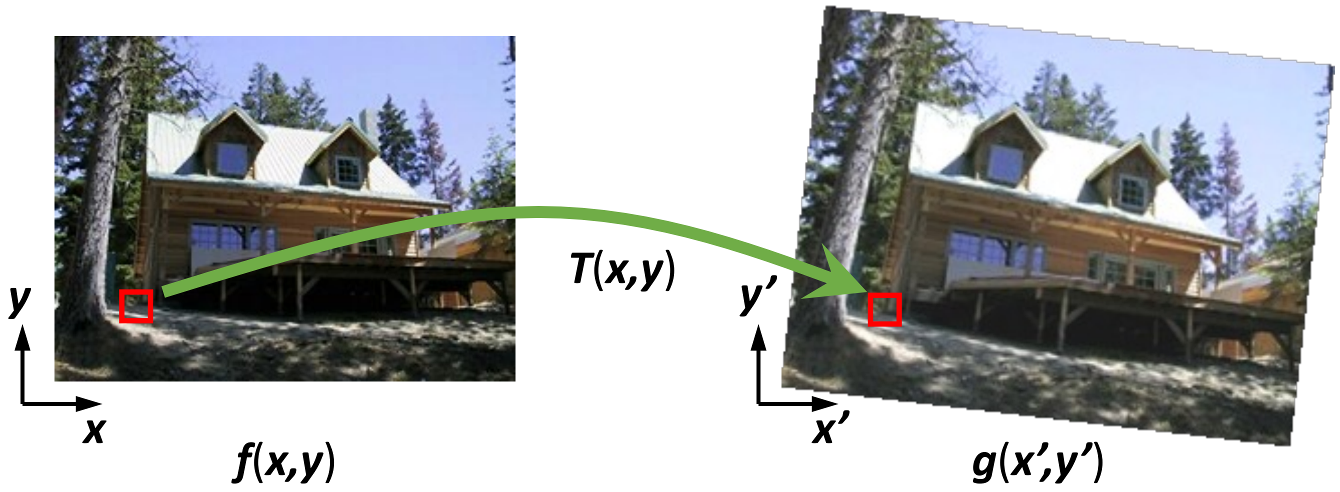
- Pros
  - Simple and general
  - Applicable to many different problems
  - Often works well in practice
- Cons
  - Parameters to tune
  - Sometimes too many iterations are required
  - Can fail for extremely low inlier ratios
  - We can often do better than brute-force sampling

Today’s class

- Fitting with outliers – RANSAC
- Warping
- Blending
- HW3 Motivation

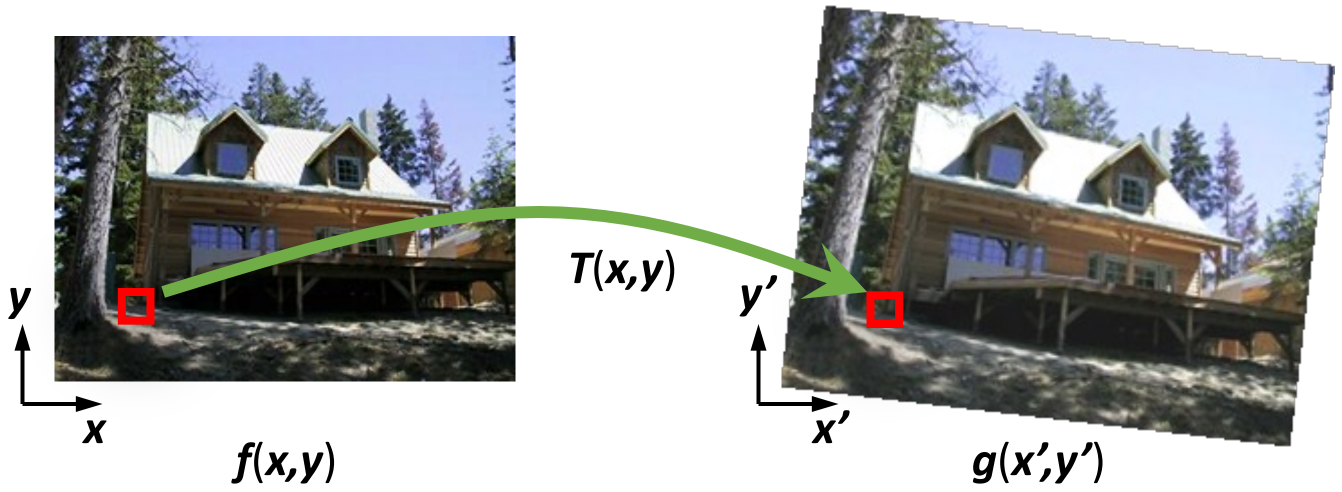
Implementing image warping

- Given a coordinate xform (***x'***,***y'***) = ***T***(***x***,***y***) and a source image ***f***(***x***,***y***), how do we compute a transformed image ***g***(***x'***,***y'***) = ***f***(***T***(***x***,***y***))?



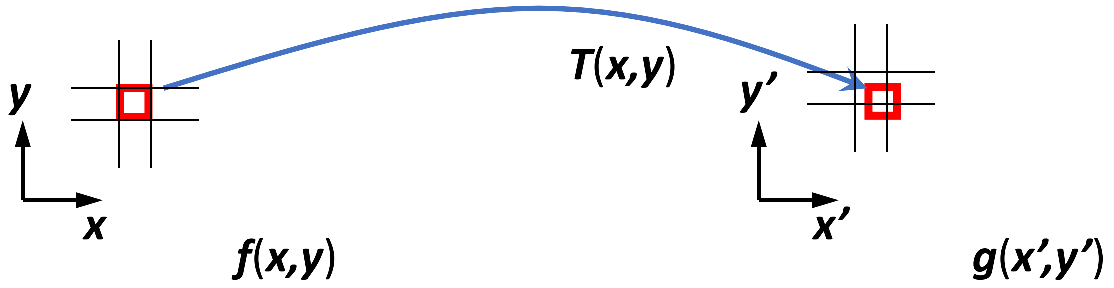
Forward Warping

- Send each pixel (***x,y***) to its corresponding location (***x',y'***) = ***T***(***x,y***) in ***g***(***x',y'***)
    - What if pixel lands "between" two pixels?



Forward Warping

- Send each pixel (***x,y***) to its corresponding location (***x',y'***) = ***T***(***x,y***) in ***g***(***x',y'***)
    - What if pixel lands "between" two pixels?
    - Answer: add "contribution" to several pixels, normalize later (*splatting*)
    - Can still result in holes



Today's class

- Fitting with outliers – RANSAC
- Warping
- Blending
- HW3 Motivation

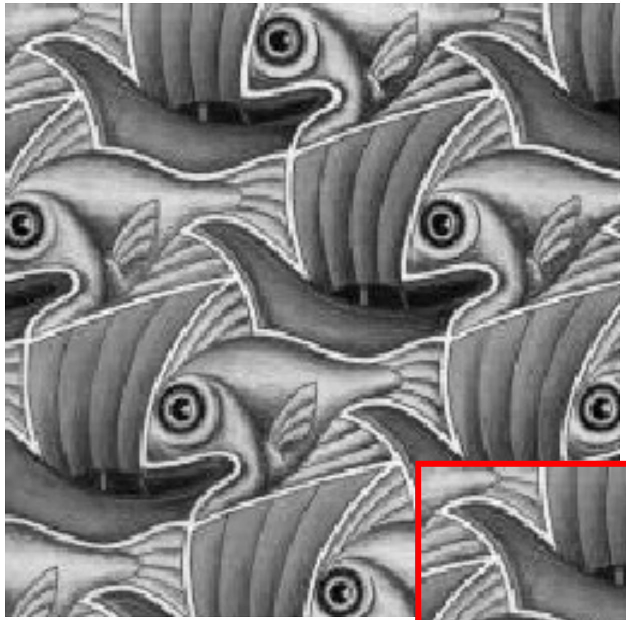
Blending

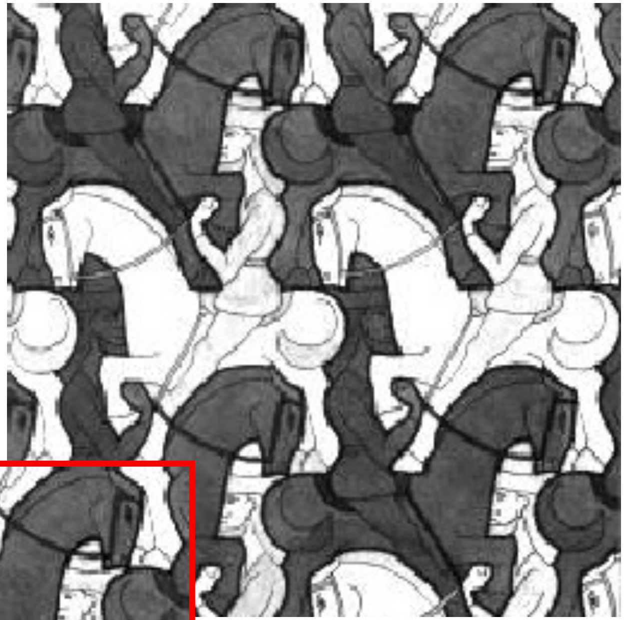
- We've aligned the images – now what?

Blending

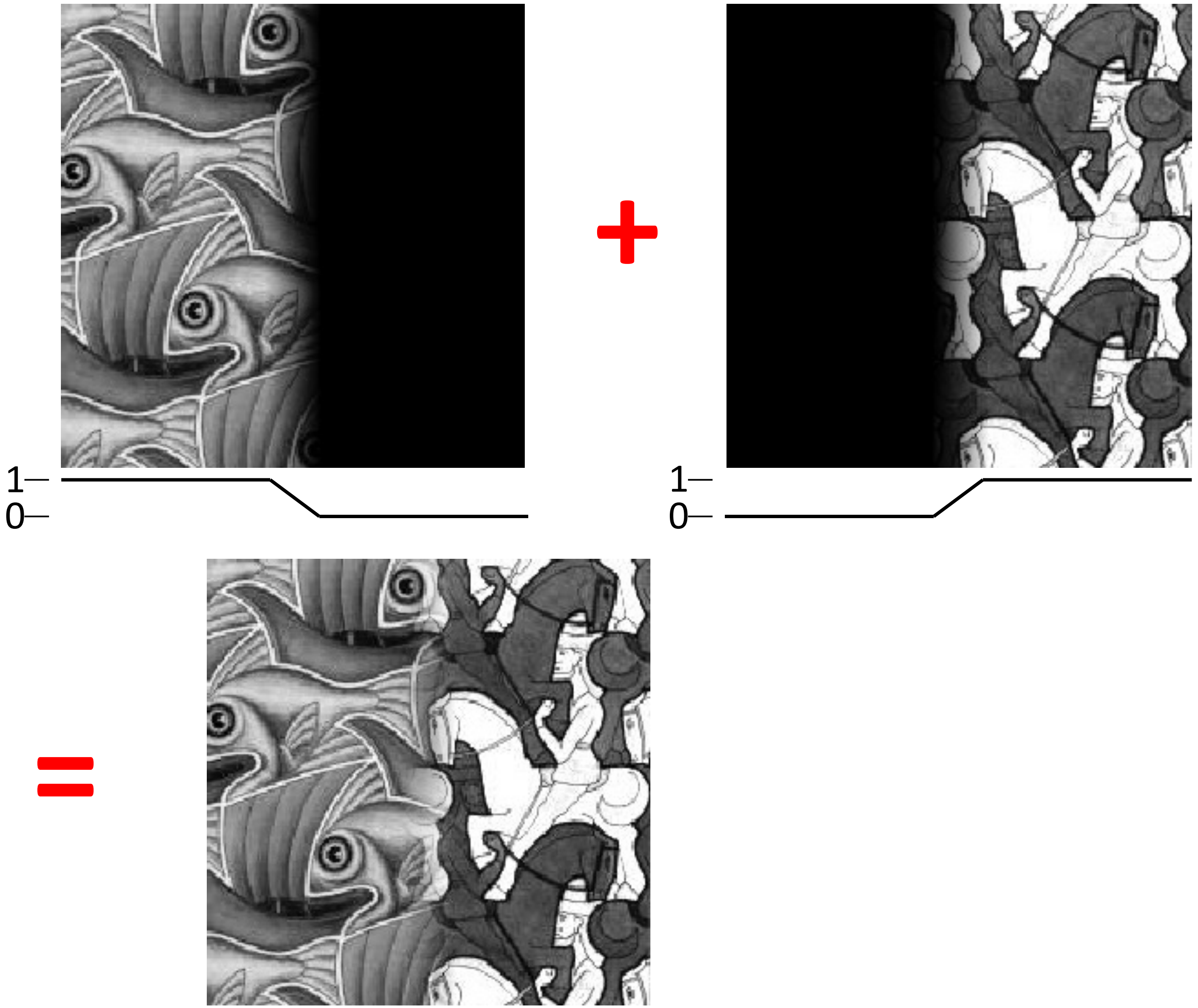
- Want to seamlessly blend them together

Image Blending

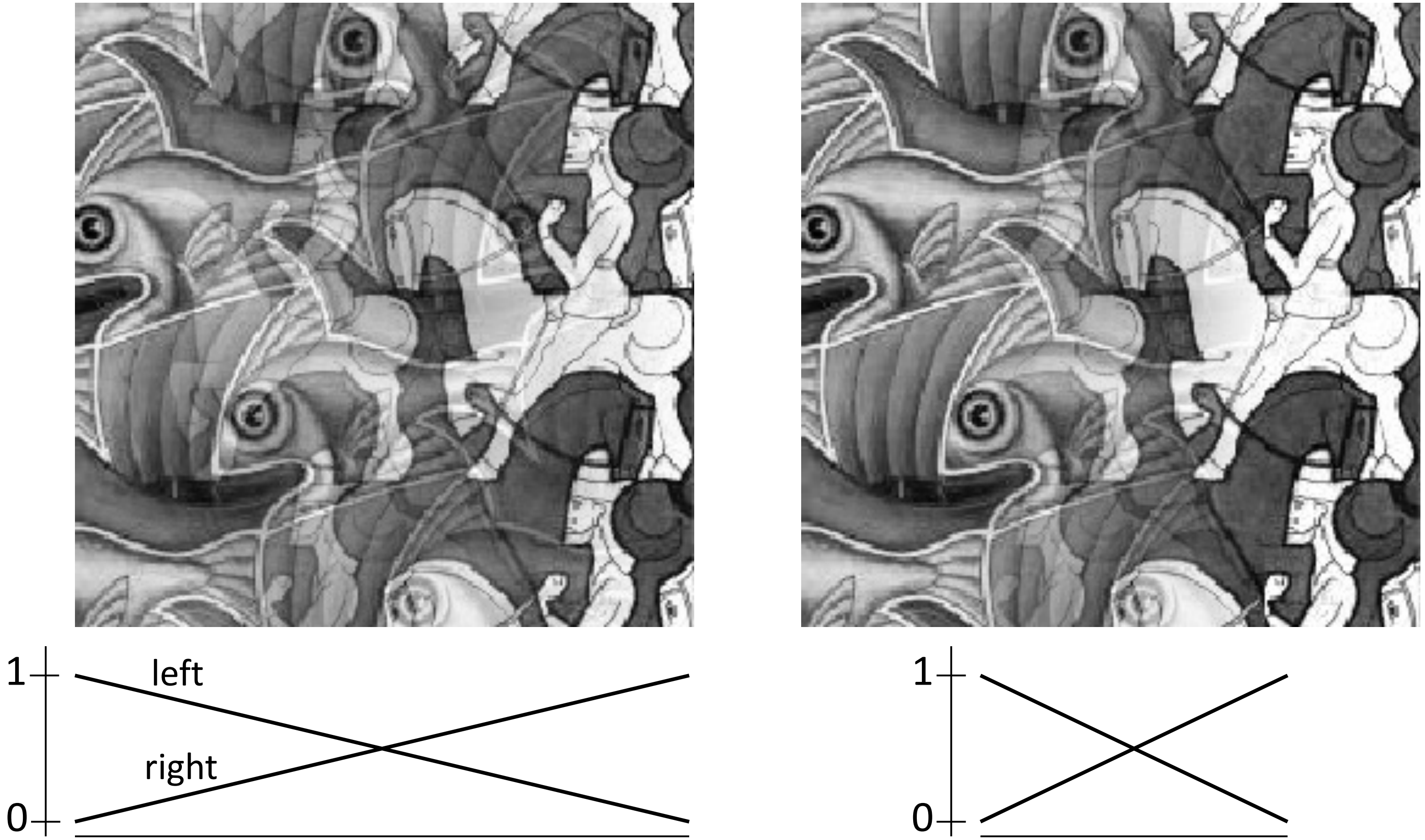




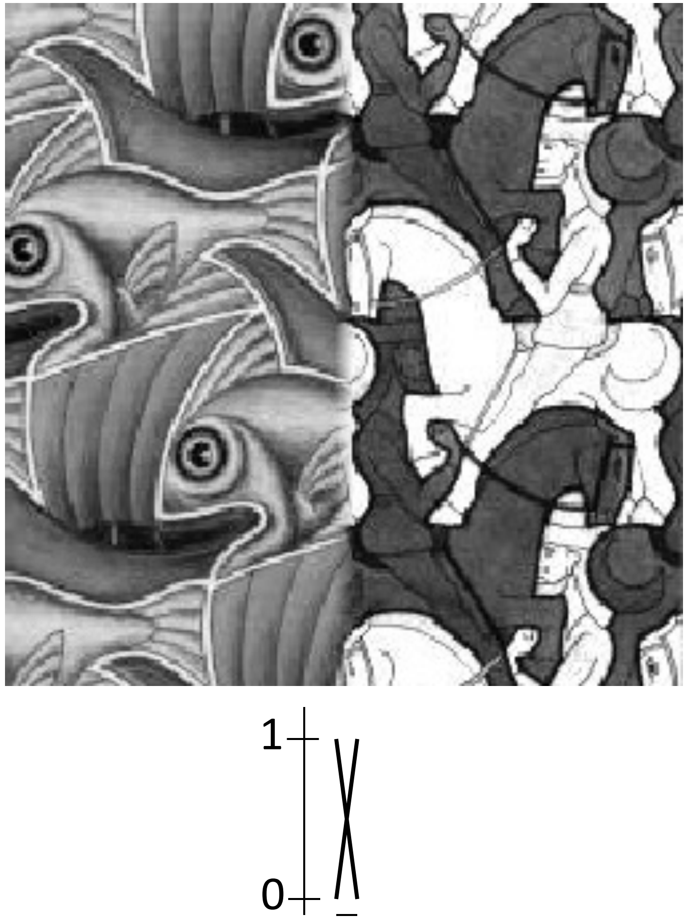
Feathering

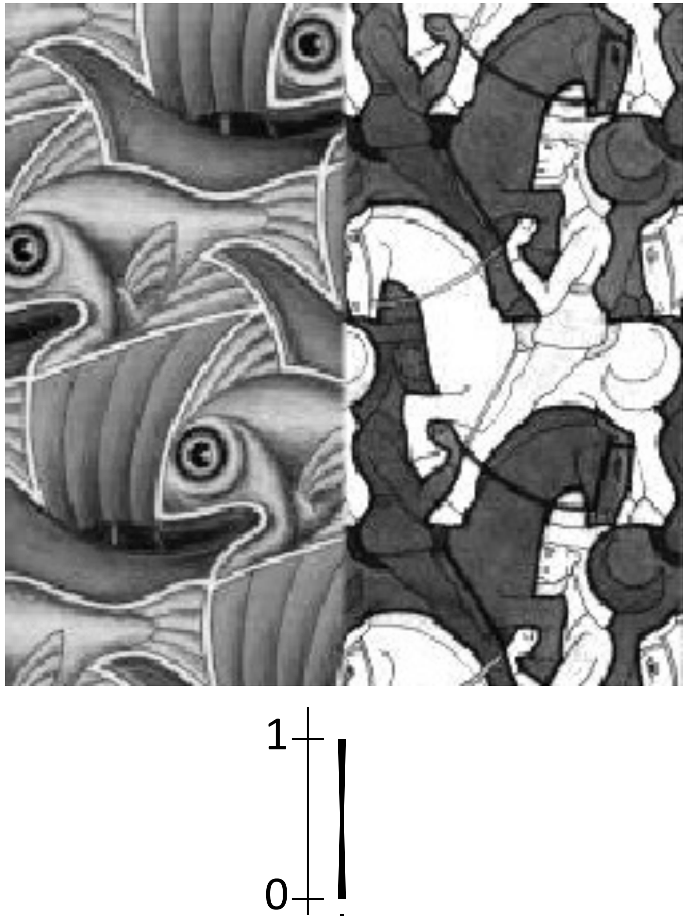


Effect of window size

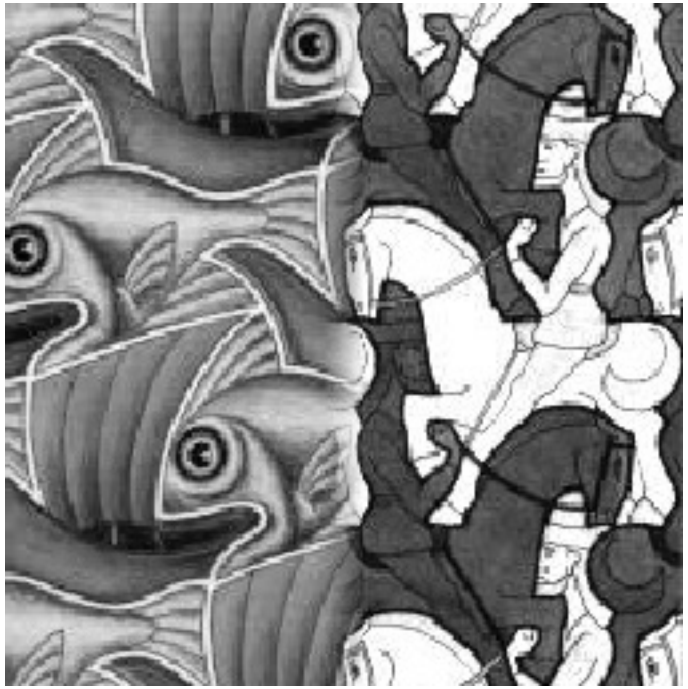


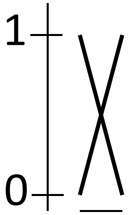
Effect of window size





Good window size

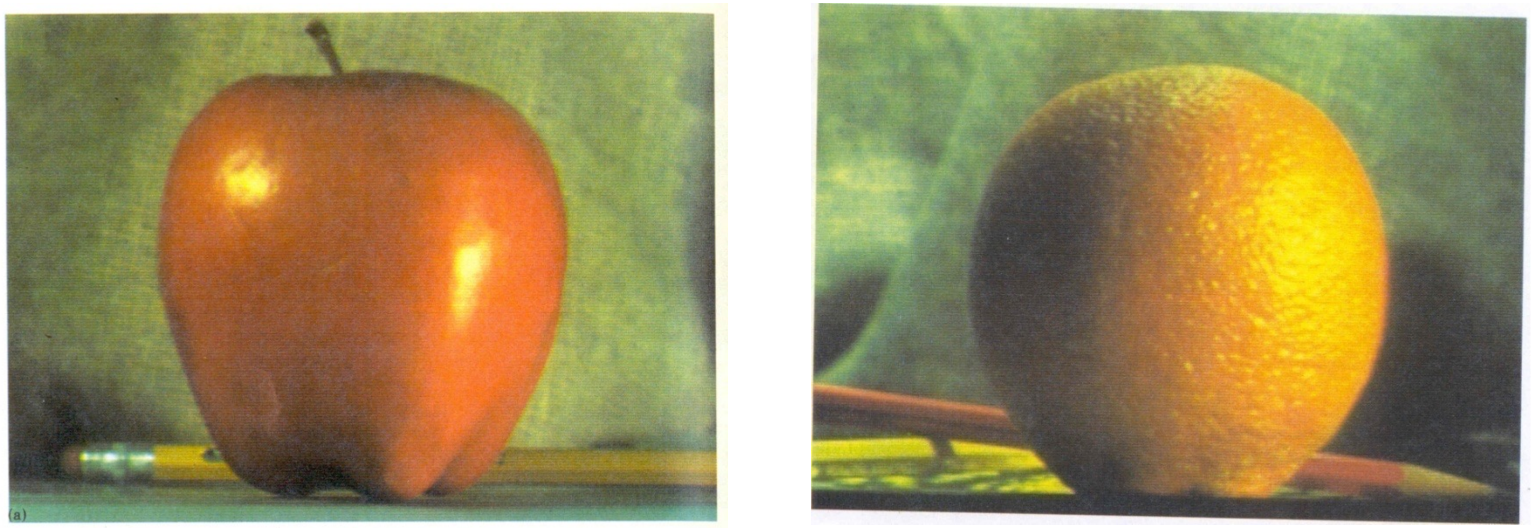


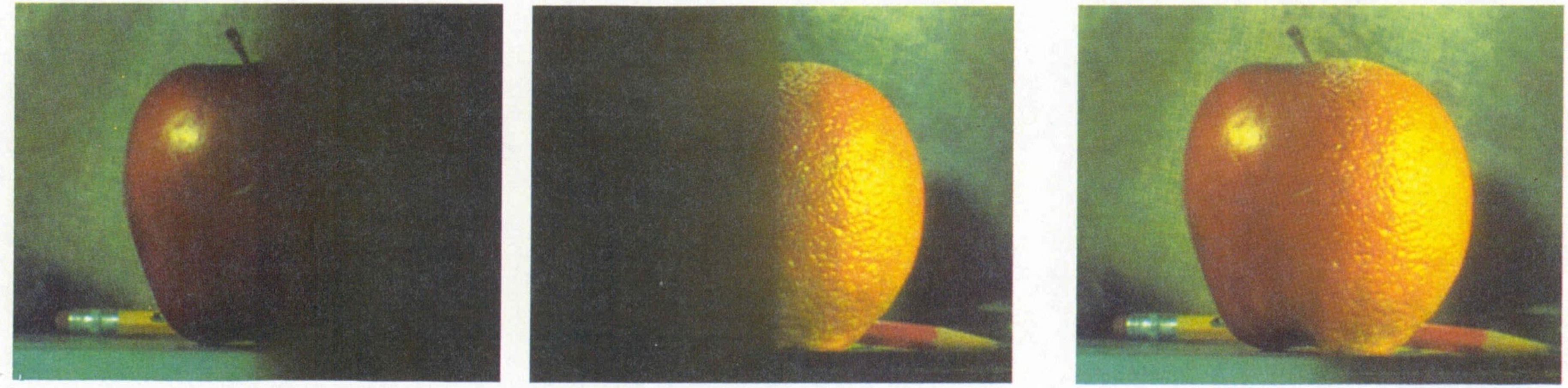


"Optimal" window: smooth but not ghosted

- Doesn't always work...

Pyramid blending



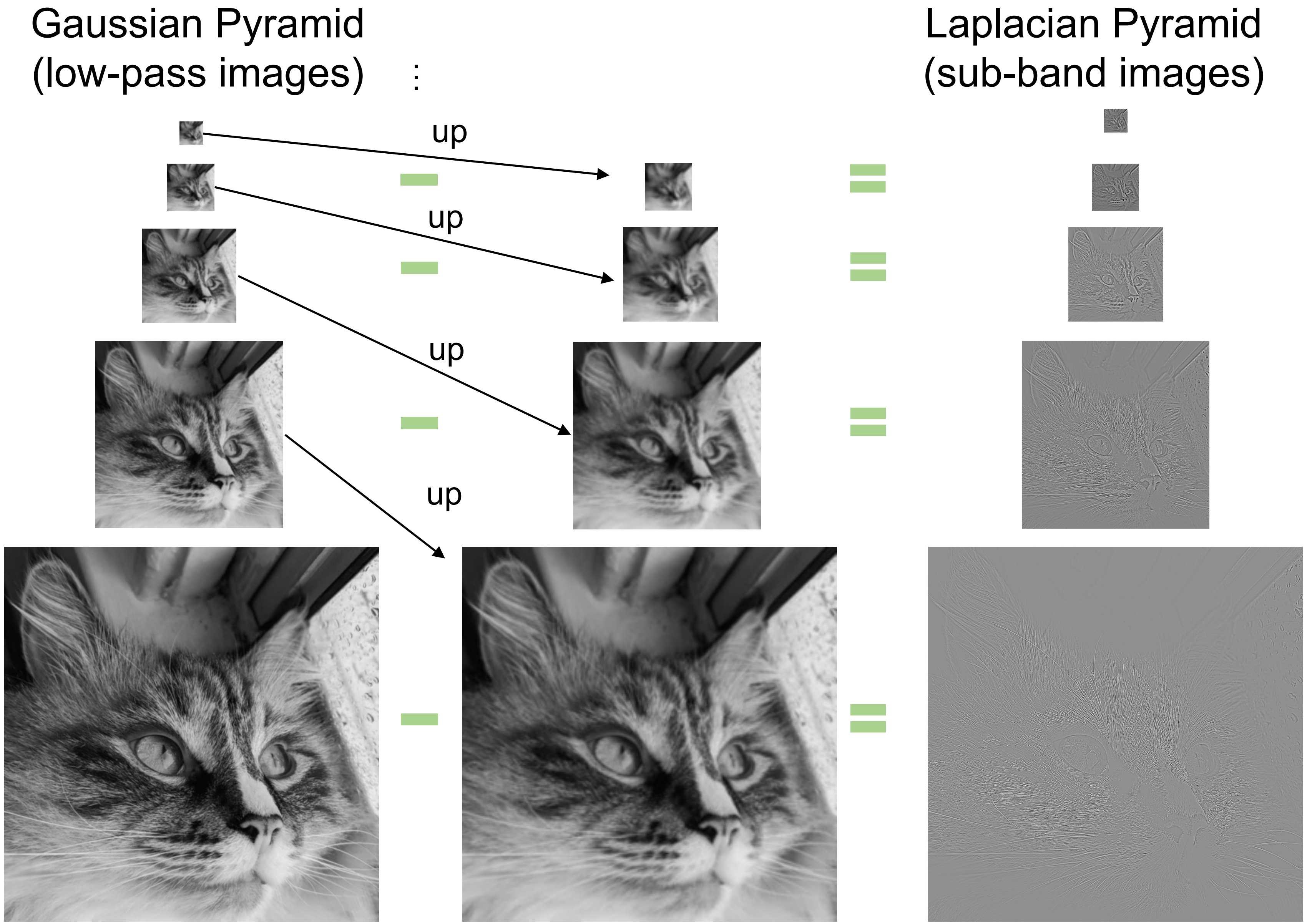


(d) (h) (l)

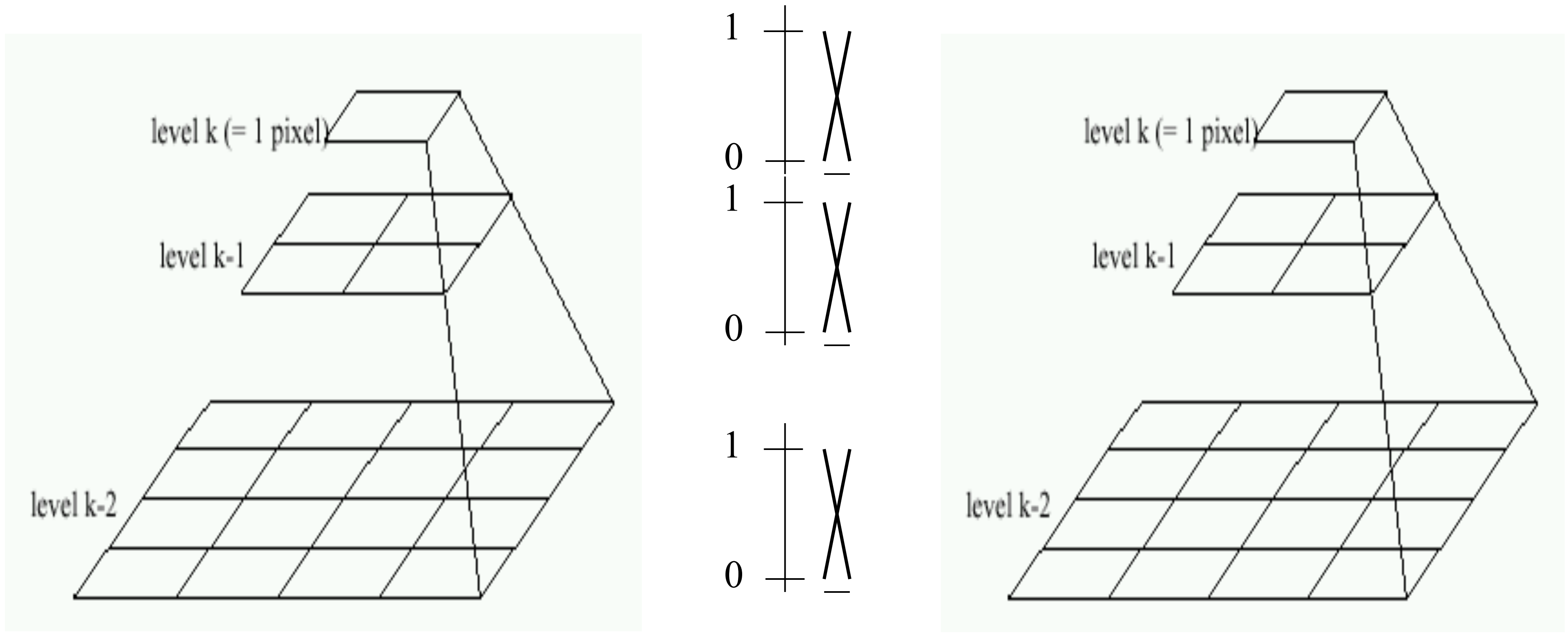
Create a Laplacian pyramid, blend each level

- Burt, P. J. and Adelson, E. H., A multiresolution spline with applications to image mosaics, ACM Transactions on Graphics, 42(4), October 1983, 217-236.

Band-pass filtering in spatial domain



Pyramid Blending



Left pyramid

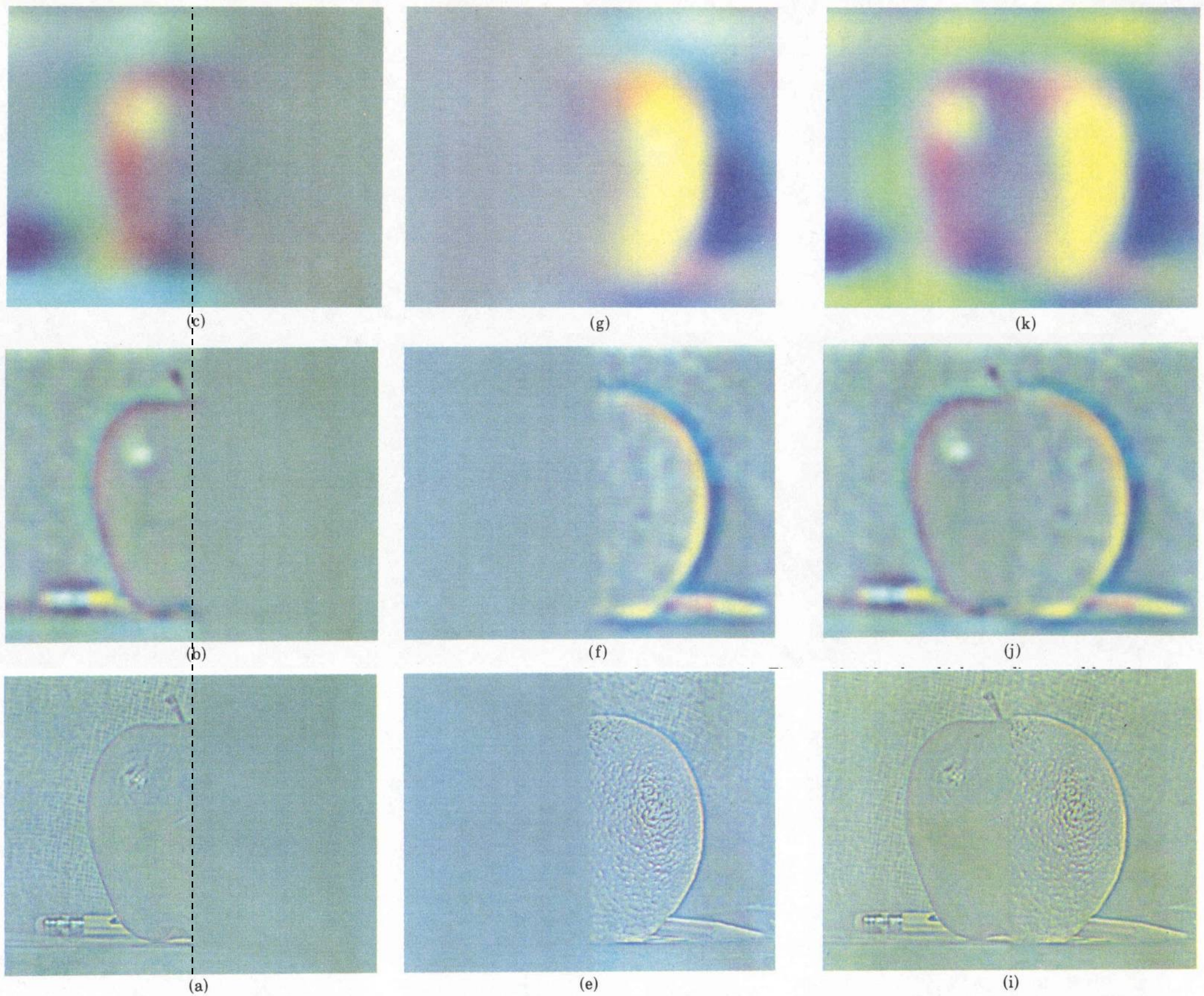
blend

Right pyramid

laplacian
level
4

laplacian
level
2

laplacian
level
0



left pyramid

right pyramid

blended pyramid

Poisson Image Editing

For more info: [Perez et al, SIGGRAPH 2003]

Today’s class

- Fitting with outliers – RANSAC
- Warping
- Blending
- HW3 Motivation

Fun with homographies

Original image

St.Petersburg
photo by A. Tikhonov

Virtual camera rotations

Analysing patterns and shapes

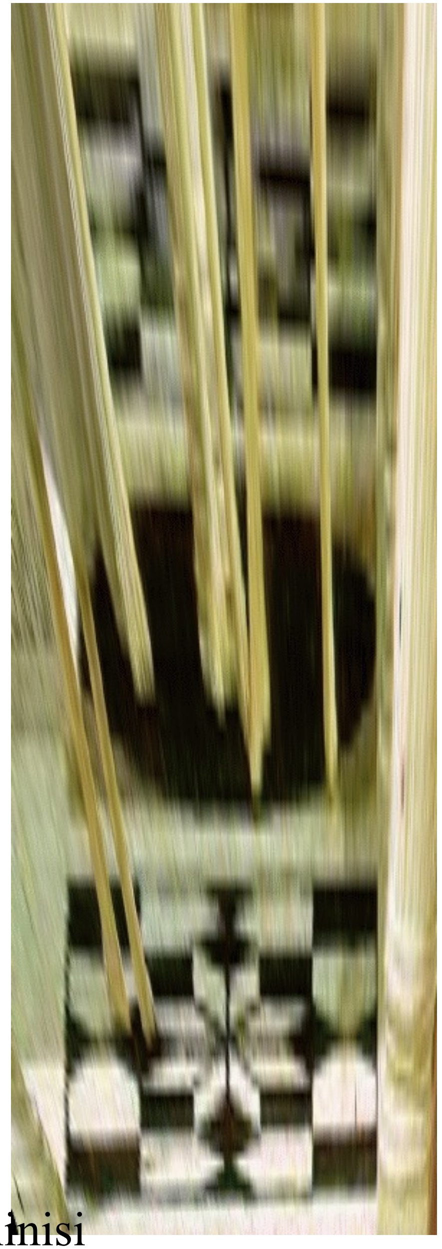
What is the shape of the b/w floor pattern?

The floor (enlarged)

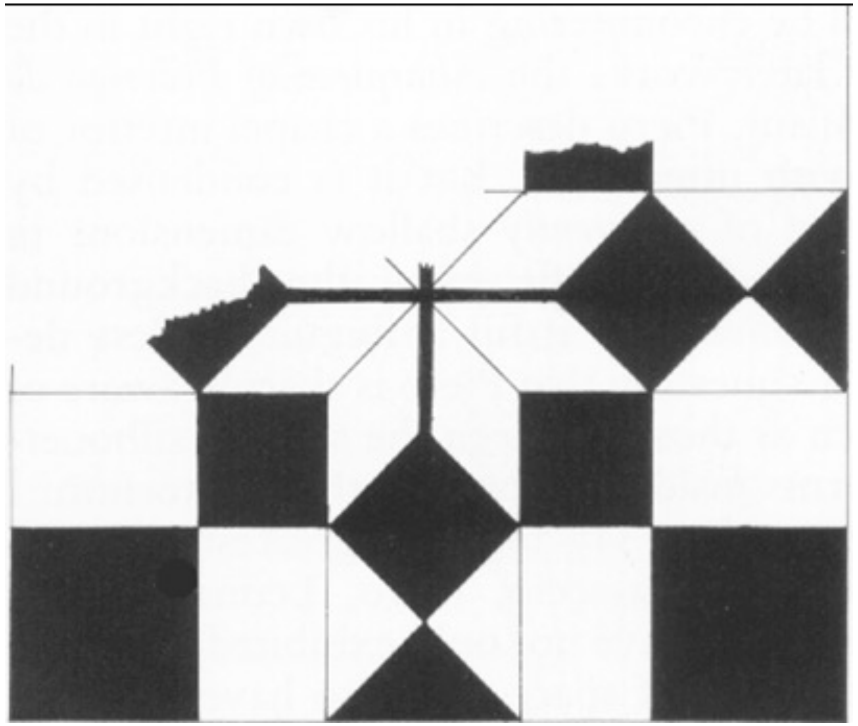
Automatically rectified floor

Slide from Criminisi

Analysing patterns and shapes



Automatic rectification



From Martin Kemp *The Science of Art*
(manual reconstruction)

2 patterns have been discovered !

Slide from Criminisi

Analysing patterns and shapes

St. Lucy Altarpiece, D. Veneziano

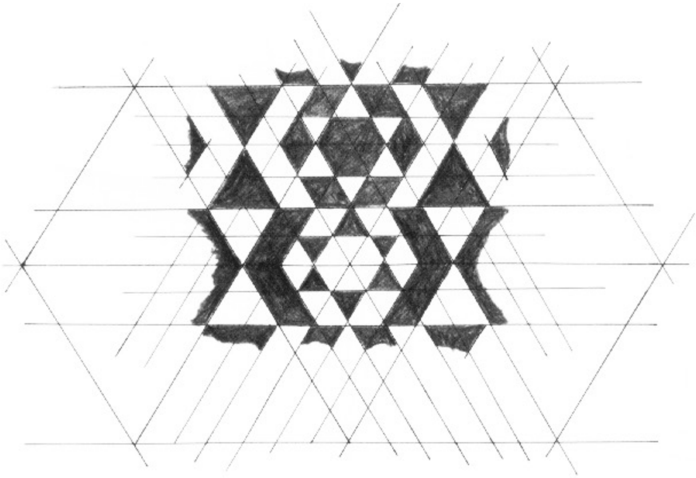
Slide from Criminisi

What is the (complicated) shape of the floor pattern?

Automatically rectified floor

Analysing patterns and shapes

Automatic rectification



From Martin Kemp, *The Science of Art* *(manual reconstruction)*

Slide from Criminisi

virtual wide-angle camera

Some panorama examples

“Before SIGGRAPH Deadline” Photo credit: Doug Zongker

Some panorama examples

- Every image on Google Streetview

Slide Credits

- CS5670, Introduction to Computer Vision, **Cornell Tech, by Noah Snavely.**
- CS 194-26/294-26: Intro to Computer Vision and Computational Photography, **UC Berkeley, by Alyosha Efros.**