# Lecture 8

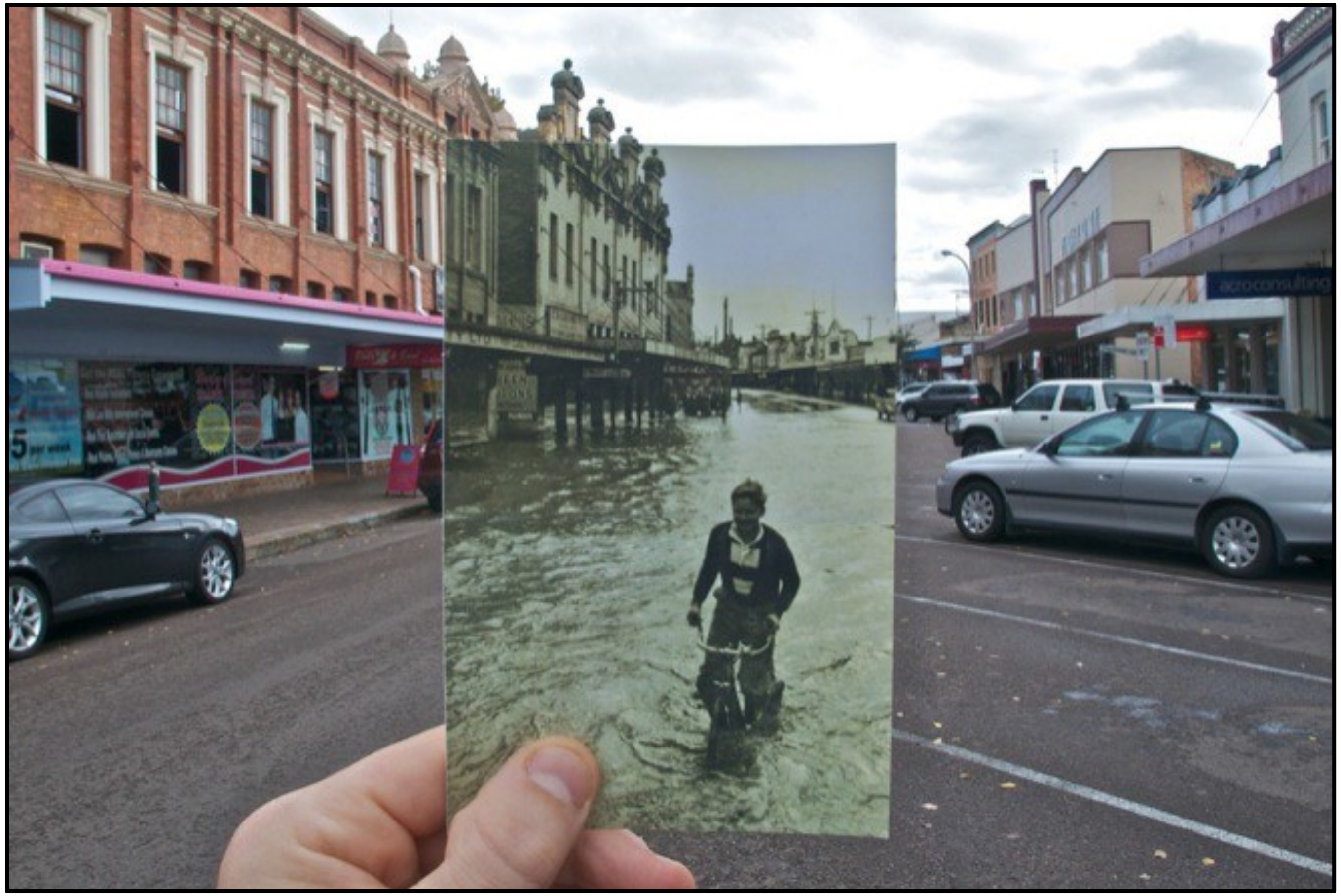
## Homographies and Global Descriptors

# Today’s agenda

- Local descriptors (SIFT)
  - Making keypoints rotation invariant
  - Designing a descriptor
  - Designing a matching function
- Image Homography
- Global descriptors (HoG)

# Image homographies

a geometric transformation that maps points from one image plane to another



# How do you create a panorama?

Panorama: an image of (near) 360o field of view.

# How do you create a panorama?

Could Use a very wide-angle lens.
**Pros**: Everything is done optically, single capture.
**Cons**: Lens is super expensive and bulky, lots of distortion (can be dealt-with in post).

# Or you can capture multiple photos and combine them

# How do we stitch images from different viewpoints?

# How do we stitch images from different viewpoints?

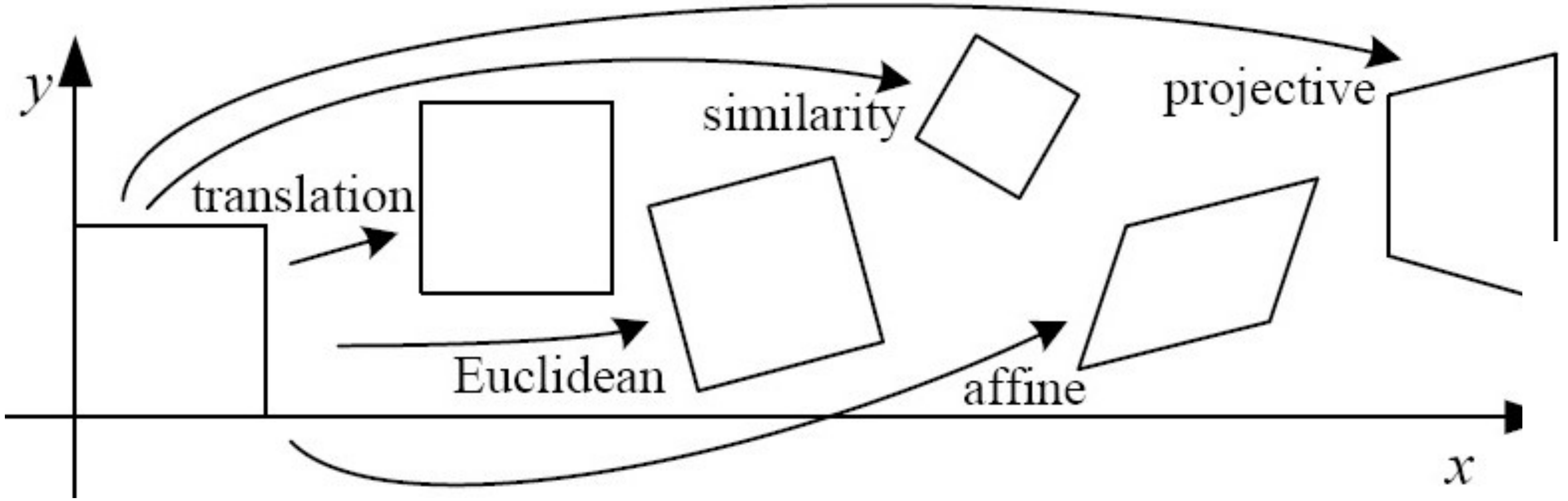
We can’t simply place on on top of another.

left on top

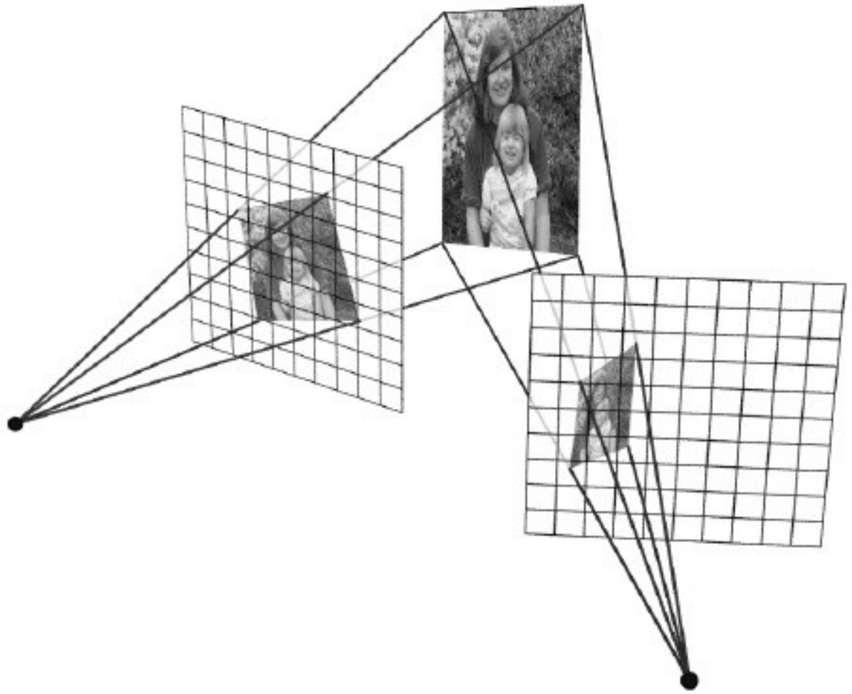
right on top

# This is where homographies come in

# Homographies explain how one image needs to be transformers

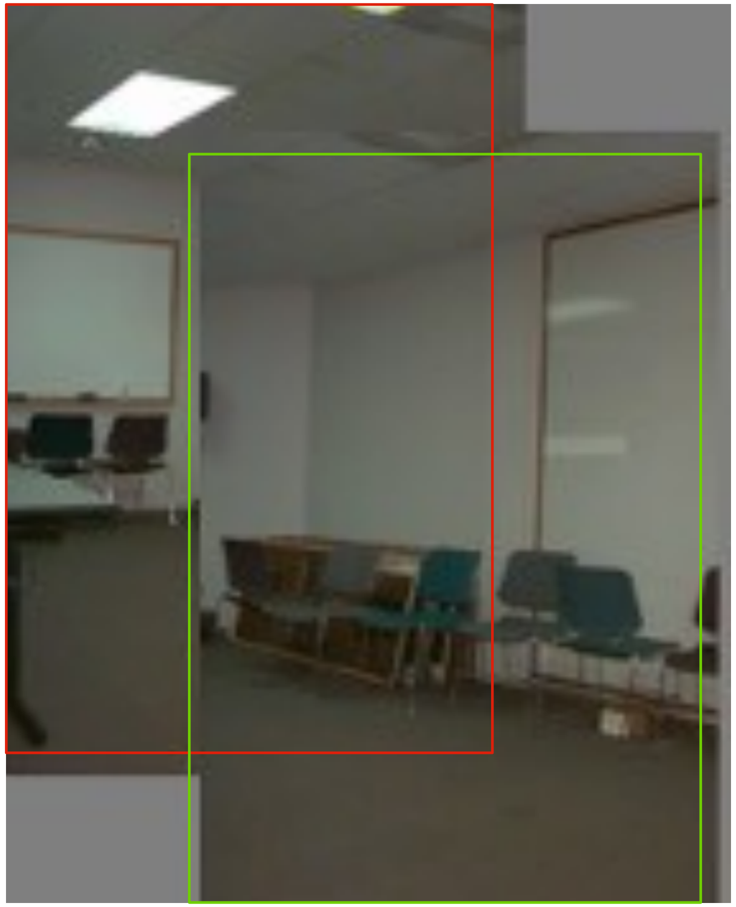



Q. Which kind of transformation is needed to warp projective plane 1 into projective plane 2?

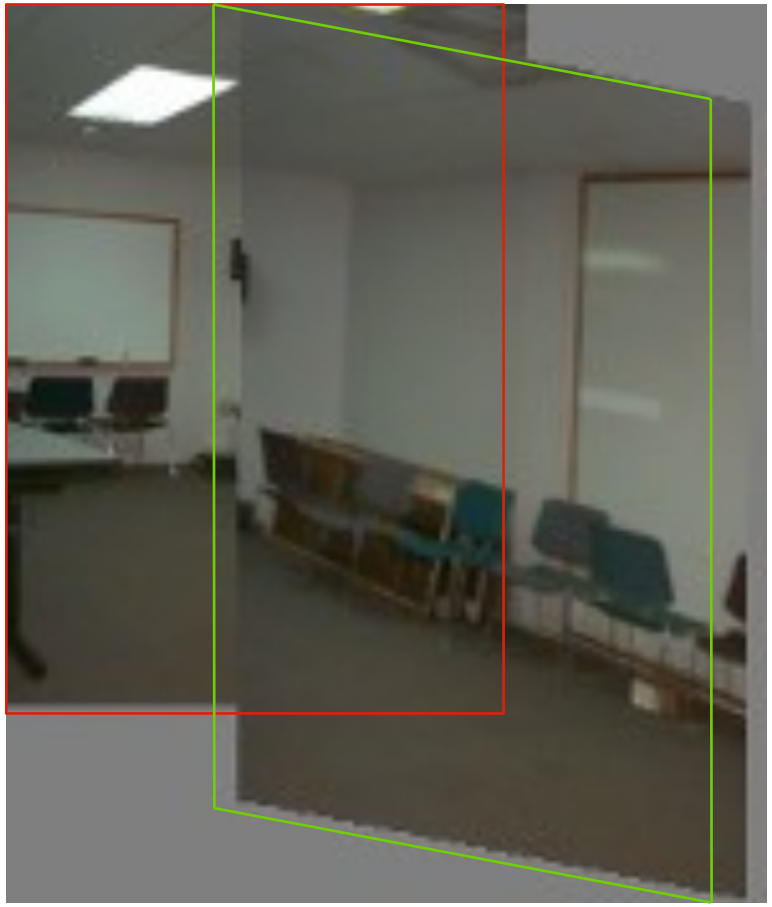


# Warping with different transformations

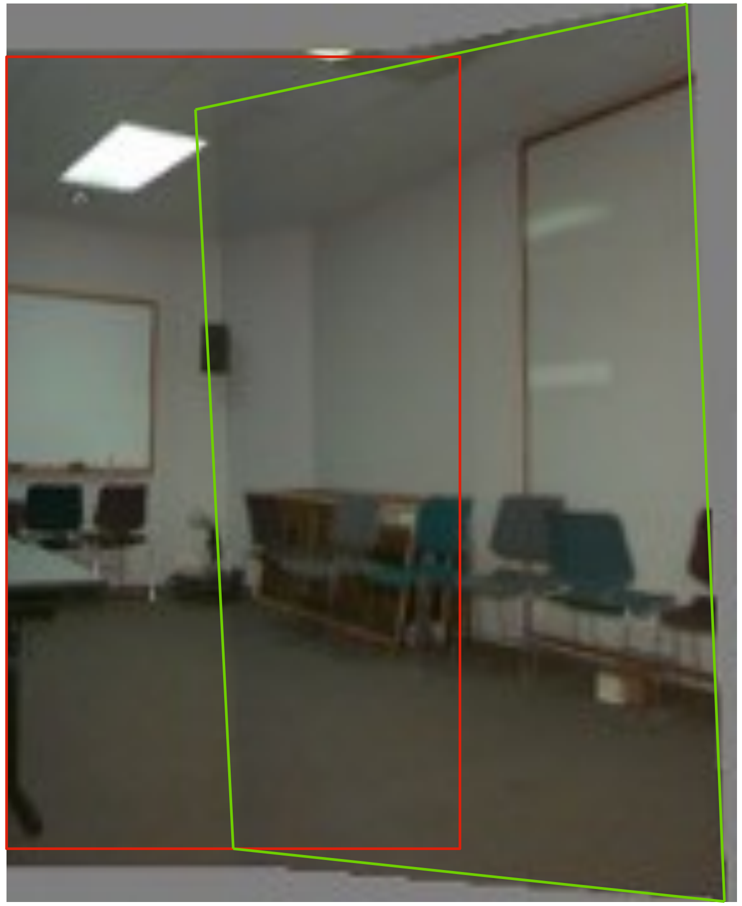
translation



affine



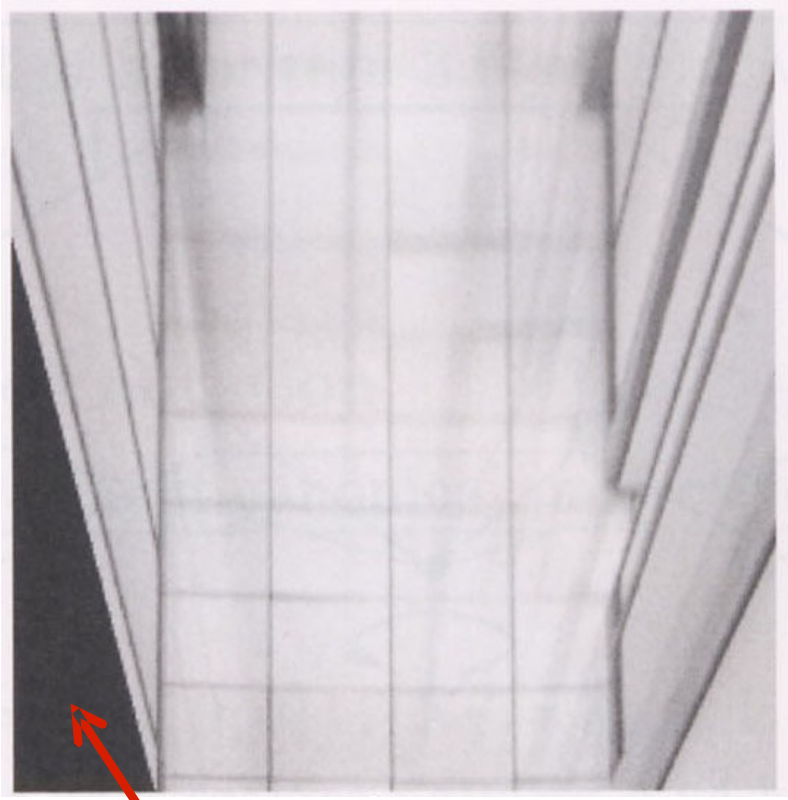
projective (homography)



# What happens when you transform one image to another view?

original view

synthetic top view



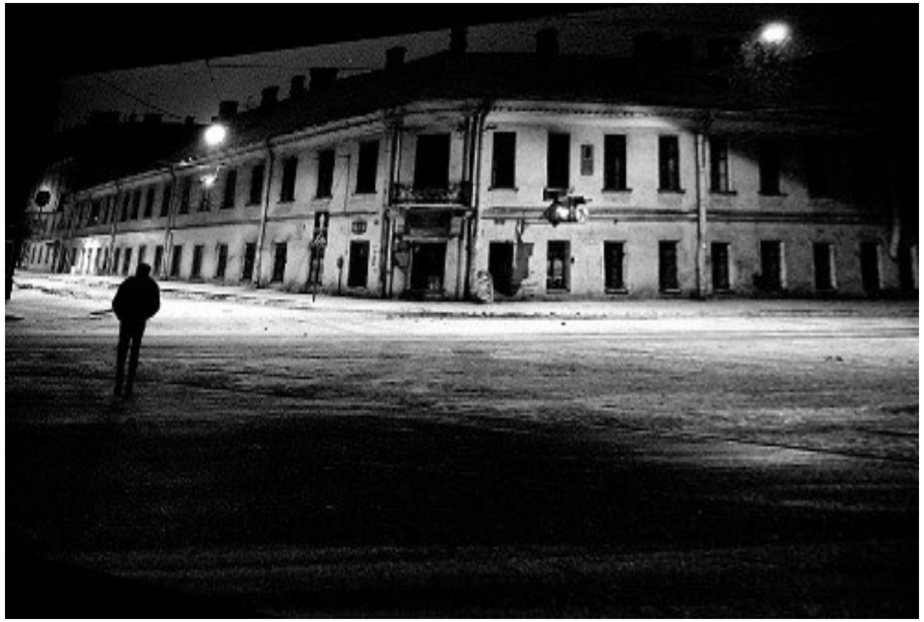
synthetic side view

What are these black areas near the boundaries?

# Virtual camera rotations

original view

synthetic rotations



# Image rectification

o
nal
ges

rectified and stitched

# Street art


# Carpet illusion

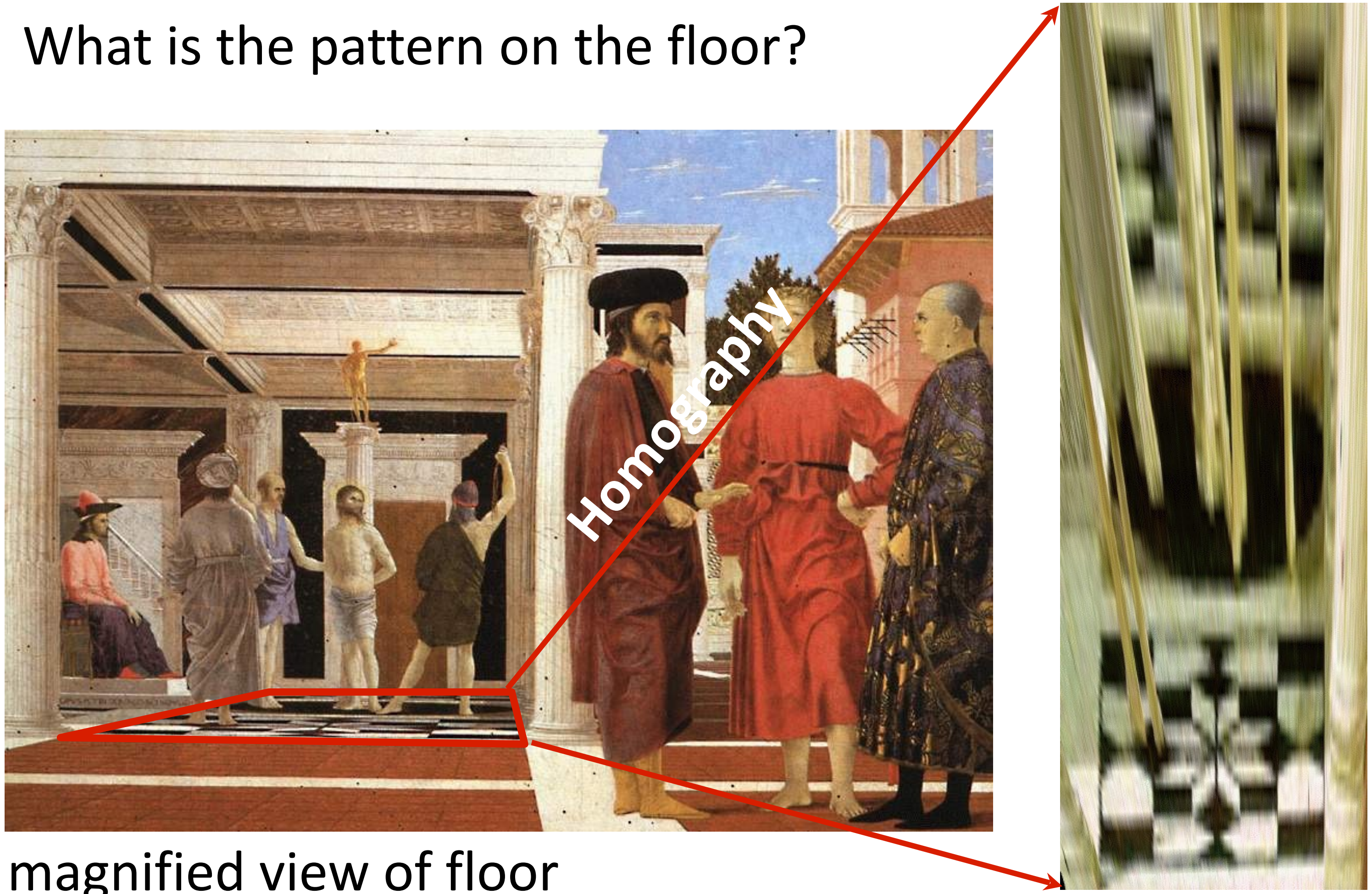
# Understanding geometric patterns

What is the pattern on the floor?

magnified view of floor

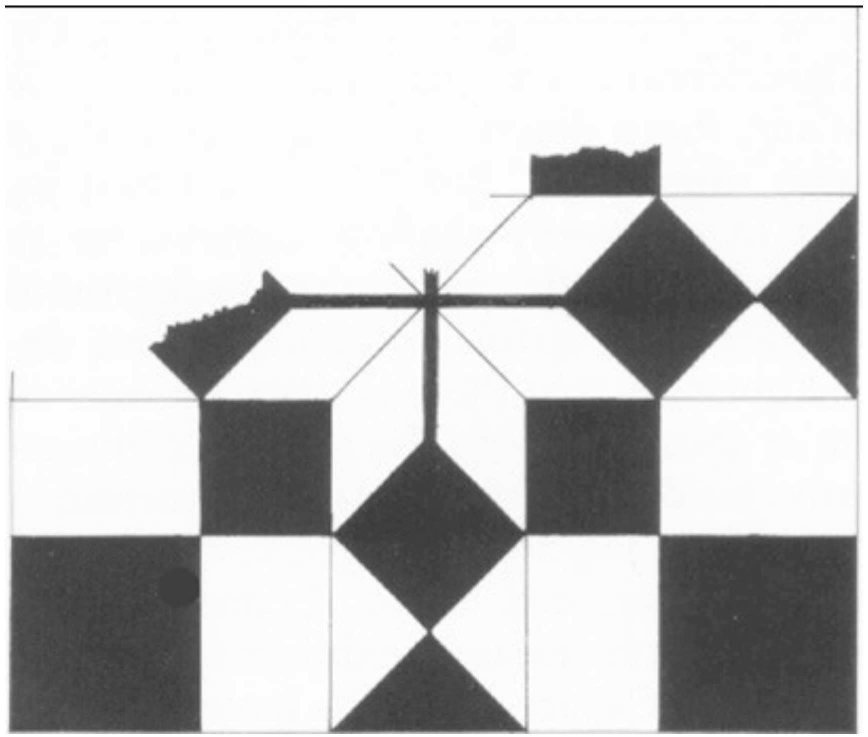
# Understanding geometric patterns

What is the pattern on the floor?




magnified view of floor

rectified view

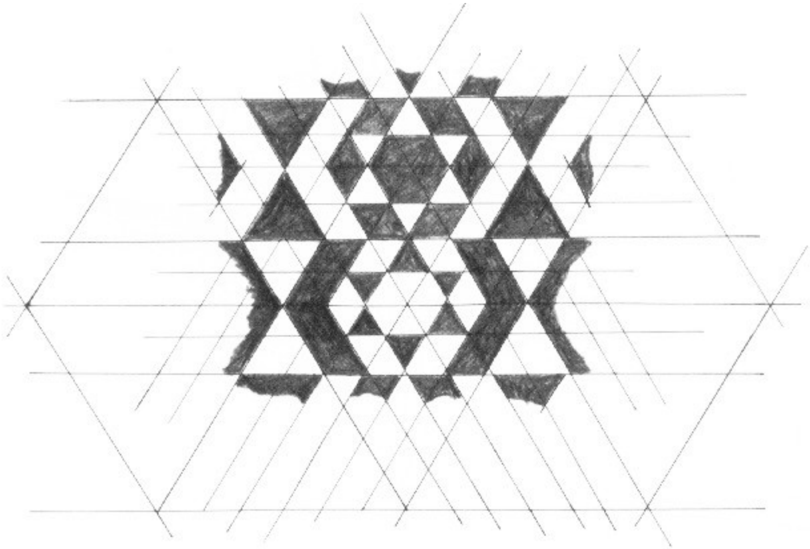


reconstruction from rectified view

# Understanding geometric patterns

What is the pattern on the floor?

rectified view of floor



reconstruction

# A weird painting

Holbein, “The Ambassadors”

# A weird painting

What’s this???

Holbein, “The Ambassadors”

# A weird painting

rectified view

rectified viewskull under anamorphic perspective

Holbein, “The Ambassadors”

# A weird painting

Holbein, “The Ambassadors”

DIY: use a polished spoon to see the skull

# What we will focus on: Panoramas

1. Capture multiple images from different viewpoints.

2. Stitch them together into a virtual wide-angle image.

# When can we calculate homographies?

when the scene is planar;
or

when the scene is very far or has small (relative) depth variation → scene is approximately planar

# When can we calculate homographies?

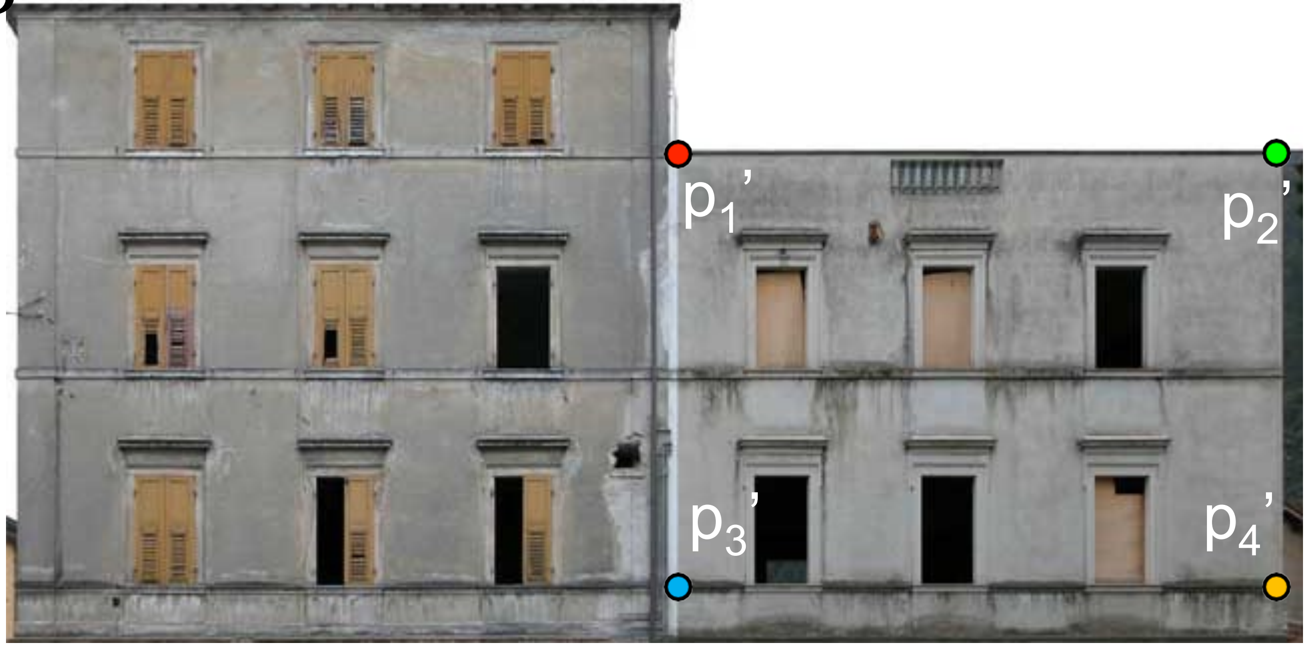
when the scene is captured under camera rotation only (no translation or pose change)

More on why this is the case in a later lecture.

# How do we do it? Keypoint matching!

The homography matrix H!

$P' = H \cdot P$


original image target image

How many correspondences do we need?

# Determining the homography matrix

$P!$ $= H \cdot$

$P$

# Determining the homography matrix

Write out linear equation for each correspondence:

$P! \quad = H \cdot P$ or $$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

# Determining the homography matrix

Write out linear equation for each correspondence:

$P' = H \cdot P$ or $\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$

Q. Why is there a 1 here?

# Determining the homography matrix

Write out linear equation for each correspondence:

$P! = H \cdot P$ or $$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Q. Why is there a 1 here?

Homogenous coordinates:

More common to use w.

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

# Determining the homography matrix

Write out linear equation for each correspondence:

$$P! = H \cdot P$$

or

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Q. Why is there a 1 here?

Homogenous coordinates:

More common to use w.

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

The output x' and y' in image space is found by: $x' = \frac{x'}{w'}, \quad y' = \frac{y'}{w'}$

# Determining the homography matrix

Write out linear equation for each correspondence:

$$P' = H \cdot P \quad \text{or} \quad \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Q. Why is there a 1 here?

Homogenous coordinates:

More common to use w'

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Q. What can you say about points where w' = 0?

# Determining the homography matrix

Write out linear equation for each correspondence:

$P! = H \cdot P$ or $$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Q. Why is there an alpha there?

# Determining the homography matrix

Write out linear equation for each correspondence:

$P! = H \cdot P$ or $\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$

Expand matrix multiplication:

$$x' = \alpha(h_1 x + h_2 y + h_3)$$
$$y' = \alpha(h_4 x + h_5 y + h_6)$$
$$1 = \alpha(h_7 x + h_8 y + h_9)$$

# Determining the homography matrix

Write out linear equation for each correspondence:

$P' = H \cdot P$ or

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Expand matrix multiplication:

$$x' = \alpha(h_1 x + h_2 y + h_3)$$
$$y' = \alpha(h_4 x + h_5 y + h_6)$$
$$1 = \alpha(h_7 x + h_8 y + h_9)$$

Divide out unknown scale factor:

$$x'(h_7 x + h_8 y + h_9) = (h_1 x + h_2 y + h_3)$$
$$y'(h_7 x + h_8 y + h_9) = (h_4 x + h_5 y + h_6)$$

Ok so we have 2 equations and 9 unknowns!

$$x'(h_7x + h_8y + h_9) = (h_1x + h_2y + h_3)$$
$$y'(h_7x + h_8y + h_9) = (h_4x + h_5y + h_6)$$

Let's rearrange the terms:

$$h_7xx' + h_8yx' + h_9x' - h_1x - h_2y - h_3 = 0$$
$$h_7xy' + h_8yy' + h_9y' - h_4x - h_5y - h_6 = 0$$

Same equations from previous slide:

$$h_7 x x' + h_8 y x' + h_9 x' - h_1 x - h_2 y - h_3 = 0$$
$$h_7 x y' + h_8 y y' + h_9 y' - h_4 x - h_5 y - h_6 = 0$$

Re-write in matrix form:

$$\mathbf{A}_i \boldsymbol{h} = \mathbf{0}$$

Same equations from previous slide:

$$h_7xx' + h_8yx' + h_9x' - h_1x - h_2y - h_3 = 0$$
$$h_7xy' + h_8yy' + h_9y' - h_4x - h_5y - h_6 = 0$$

Re-write in matrix form:

$$\mathbf{A}_i \boldsymbol{h} = \mathbf{0}$$

$$\mathbf{A}_i = \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix}$$

$$\boldsymbol{h} = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 \end{bmatrix}^\top$$

# What is this form useful?

$$\mathbf{A}\boldsymbol{h} = \mathbf{0}$$

$$\begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \\ h_9 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

# We get 2 rows per matching keypoint

Stack together constraints from multiple point correspondences:

$$\mathbf{A}\boldsymbol{h} = \mathbf{0}$$

$$\begin{bmatrix} \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix} \\ \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix} \\ \vdots \\ \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \\ h_9 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

This is called the *Homogeneous* linear least squares problem

# Q. Do you remember this equation from your linear algebra course?

$$\mathbf{A}\boldsymbol{h} = \mathbf{0}$$

$$\begin{bmatrix} \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix} \\ \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix} \\ \vdots \\ \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \\ 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{bmatrix} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \\ h_9 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

This is called the *Homogeneous* linear least squares problem

# We can solve this using SVD

SVD decomposition: $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$

h parameters are the eigenvector in V associated with the smallest eigenvalue in $\mathbf{\Sigma}$

$$\boldsymbol{h} = \boldsymbol{v}_{\hat{i}}$$

# Putting it all together to create a panorama

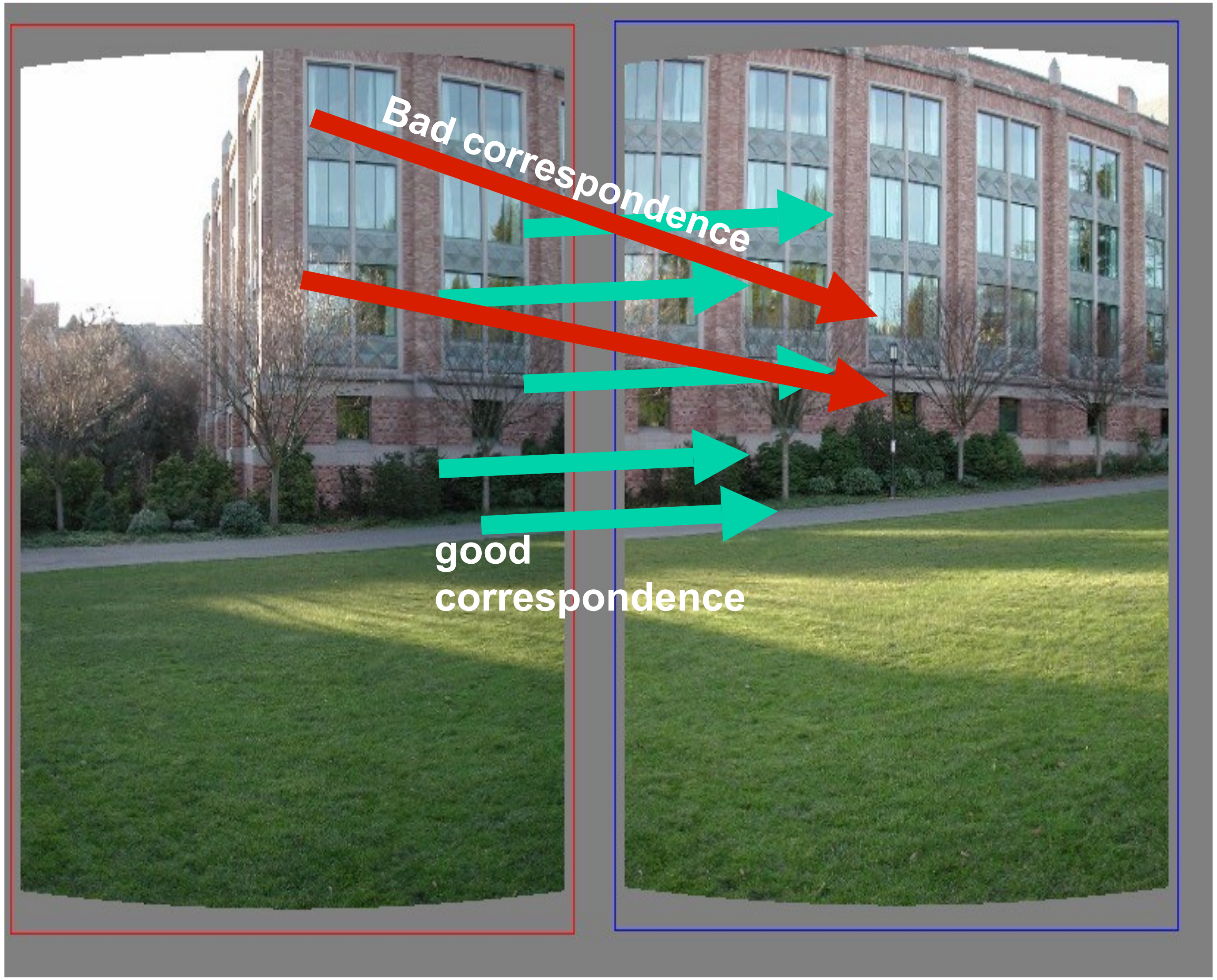
1. Find keypoints using SIFT or Harris corner
2. Find matches using local feature descriptors
3. Put all the matching points in the matrix form in the previous slide
4. Use SVD to solve for homography matrix h

# Putting it all together to create a panorama

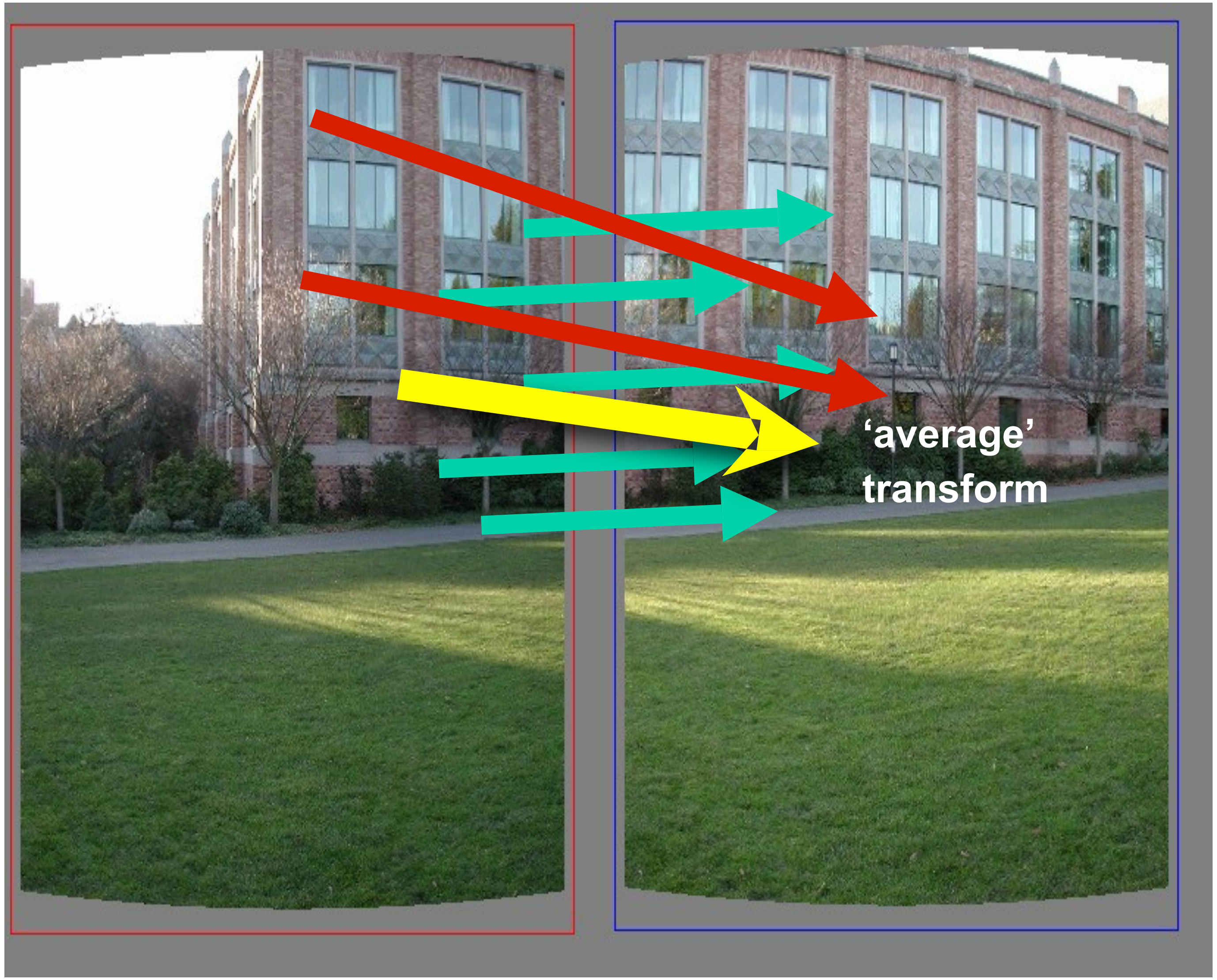
1. Find keypoints using SIFT or Harris corner
2. Find matches using local feature descriptors
3. Put all the matching points in the matrix form in the previous slide
4. Use SVD to solve for homography matrix h

Q. But wait, what if the keypoints are noisy and you have some bad matches?

Won’t that give you a bad homography???

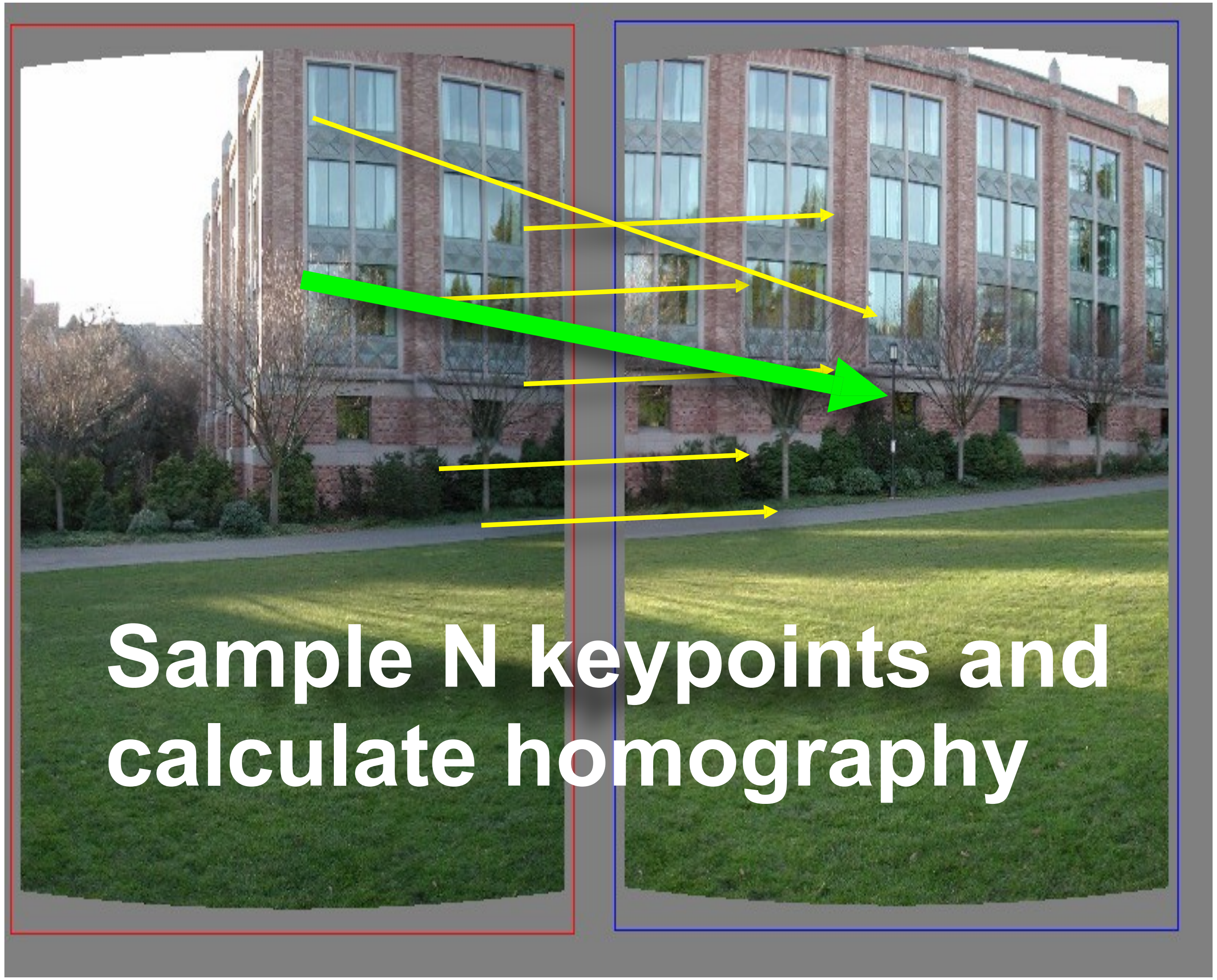

Bad correspondence
good
correspondence

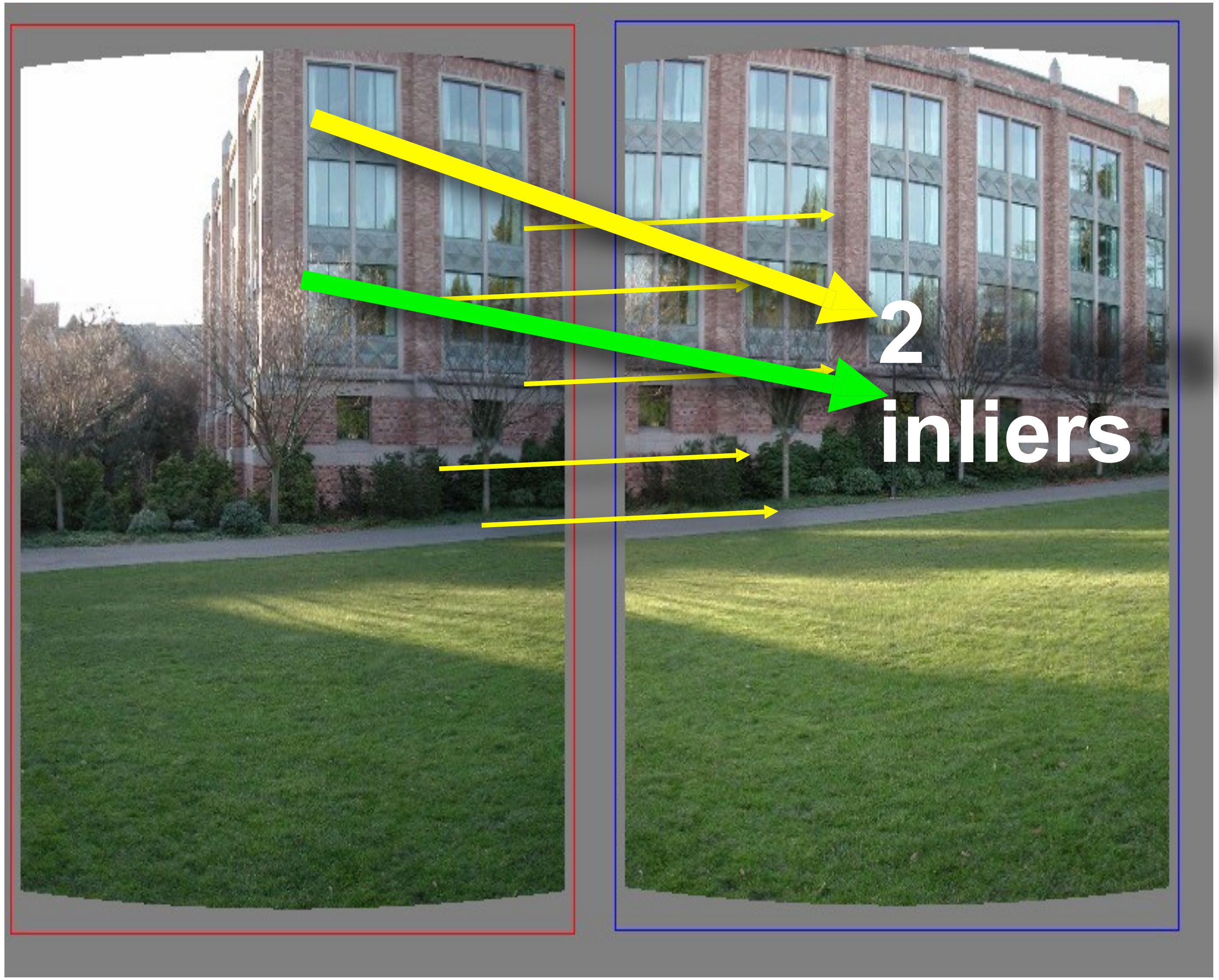
If we use noisy keypoints, we will get this bad transformation.

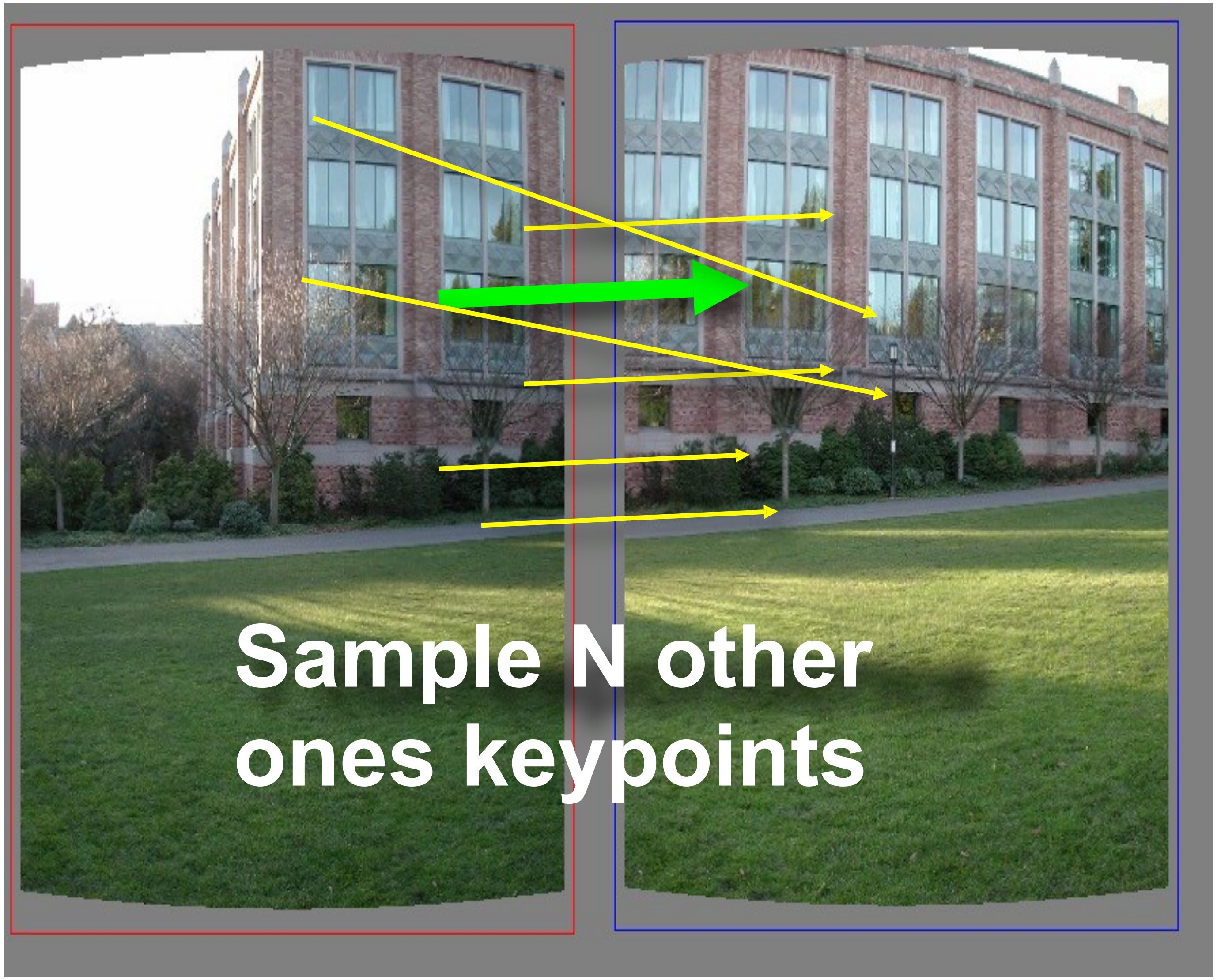


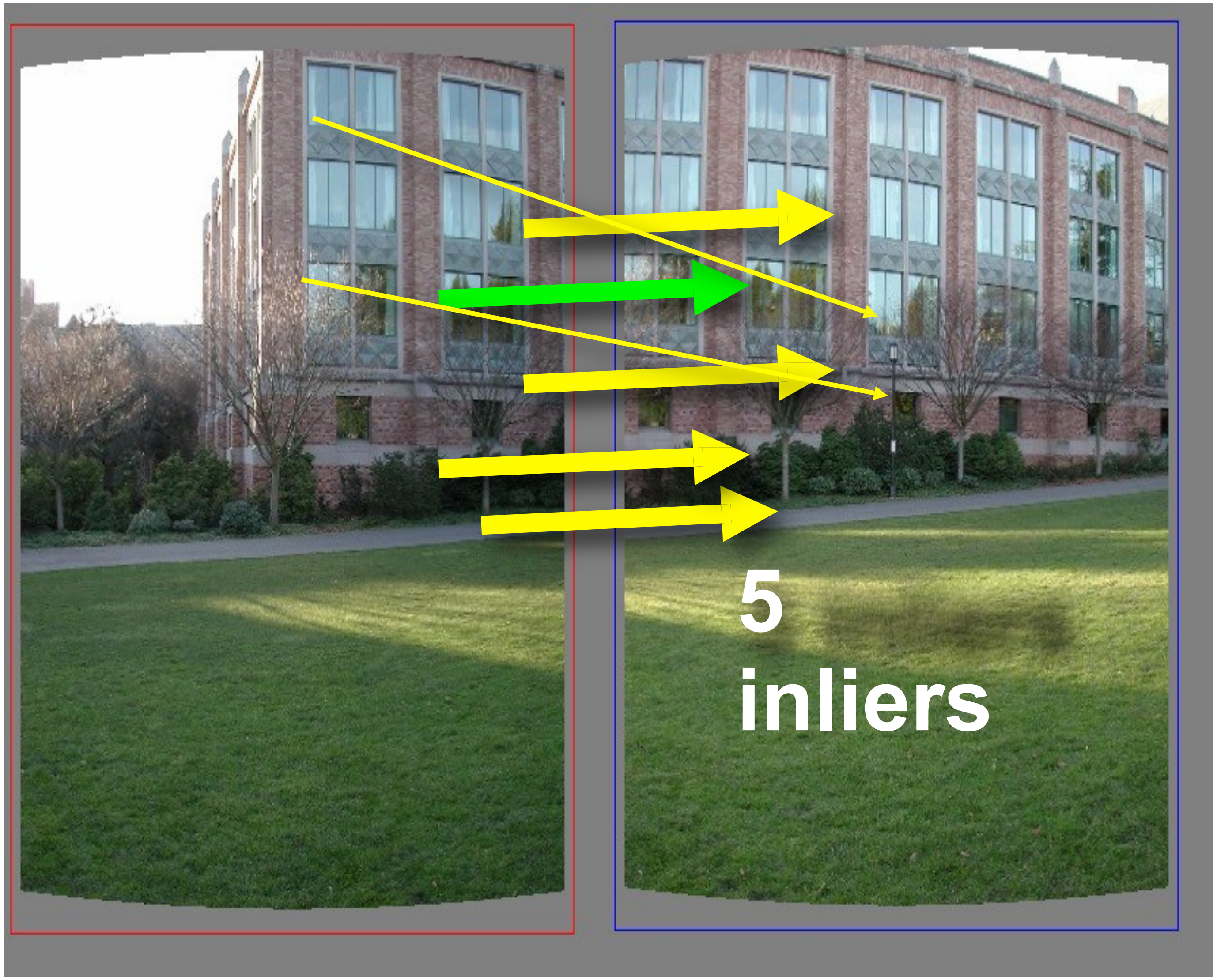

Q. Can you think of an algorithm we have learned that can fix this problem?

# RANSAC!!!!




2
inliers

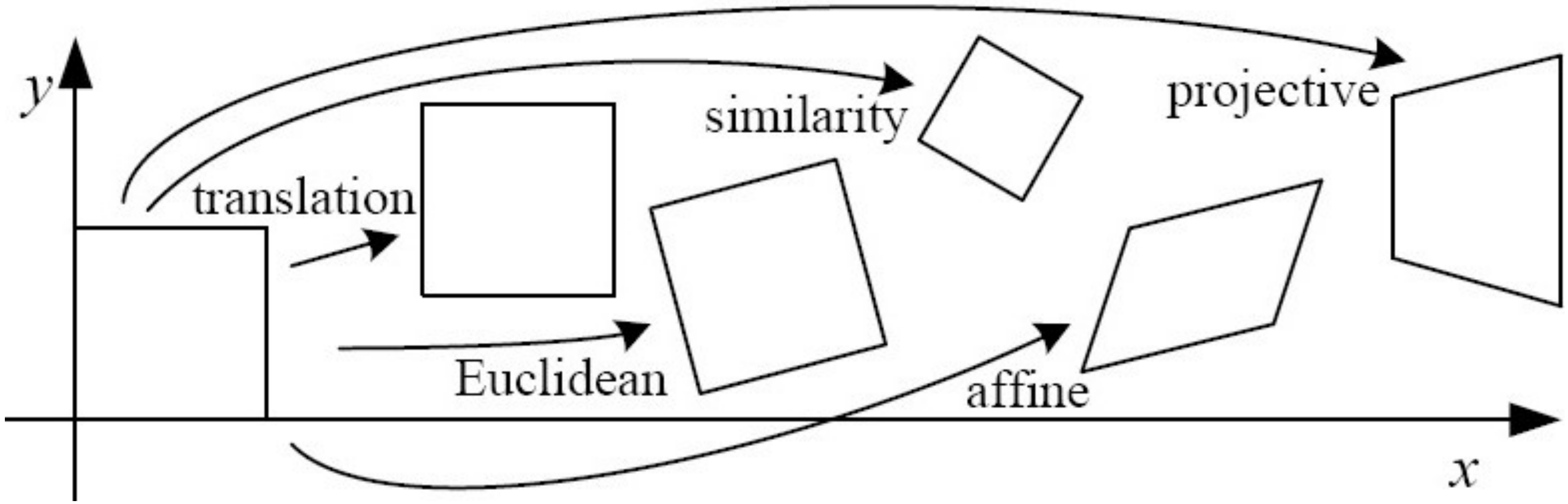

Sample N other
ones keypoints


5
inliers

# Putting it all together to create a panorama

1. Find keypoints using SIFT or Harris corner
2. Find matches using local feature descriptors
3. Sample N keypoints
    a. Put the sampled points in the matrix form $Ah = 0$
    b. Use SVD to solve for homography matrix h
    c. Calculate inliers (reprojection error + threshold)
    d. Repeat
4. Re-calculate h using the inliers from best homography

# Aside: Remember that we are doing projective transformations.




If the transformation was affine, the homography matrix would be simpler. We would only have rotation, translation and scaling.

For affine transformations, the solution is simpler!

Affine transformation:

$$H_{\text{affine}} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

# For affine transformations, the solution is simpler!

Affine transformation:

$$H_{\text{affine}} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

Vectorize transformation parameters:

Stack equations from point correspondences:

$$\begin{bmatrix} x' \\ y' \\ x' \\ y' \\ \vdots \\ x' \\ y' \end{bmatrix} = \begin{bmatrix} x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \\ x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \\ & & \vdots & & & \\ x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

# Today's agenda

- Local descriptors (SIFT)
  - Making keypoints rotation invariant
  - Designing a descriptor
  - Designing a matching function
- Image Homography
- Global descriptors (HoG)

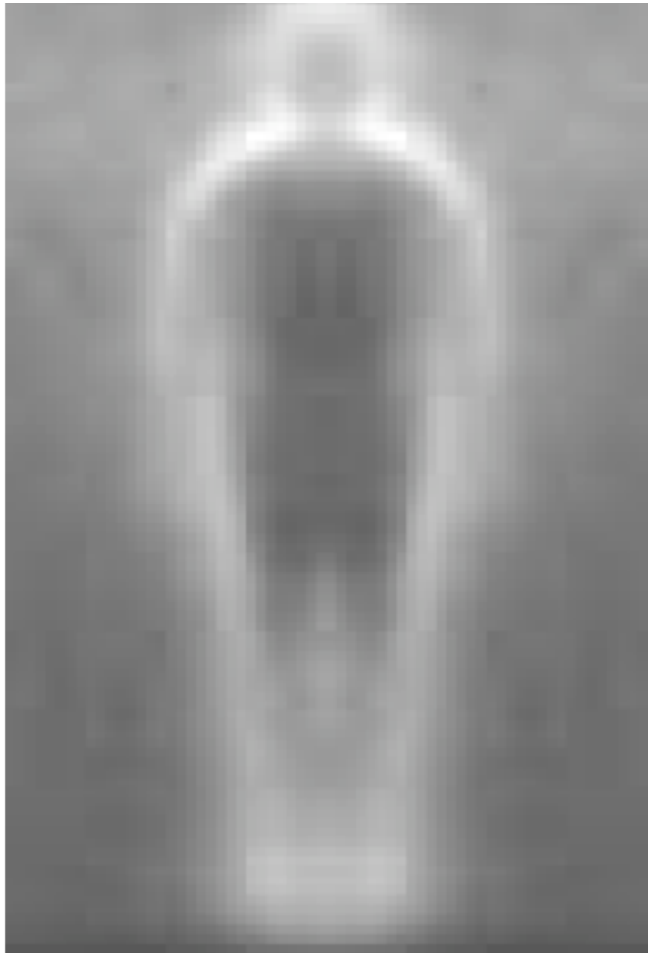
# Global Feature descriptors

- Find robust feature set that allows object shape to be recognized.
- **Challenges**
  - Wide range of pose and large variations in appearances
  - Cluttered backgrounds under different illumination
  - Computation speed
- **Histogram of Oriented Gradients (HoG)**

[1] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In CVPR, pages 886-893, 2005
[2] Chandrasekhar et al. CHoG: Compressed Histogram of Gradients - A low bit rate feature descriptor, CVPR 2009

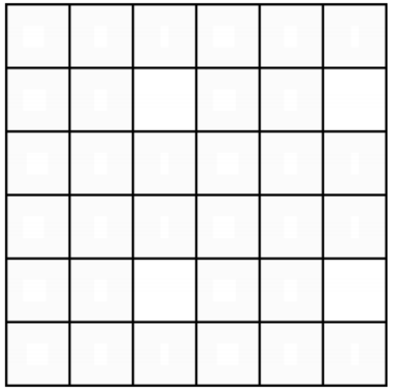
# Histogram of Oriented Gradients

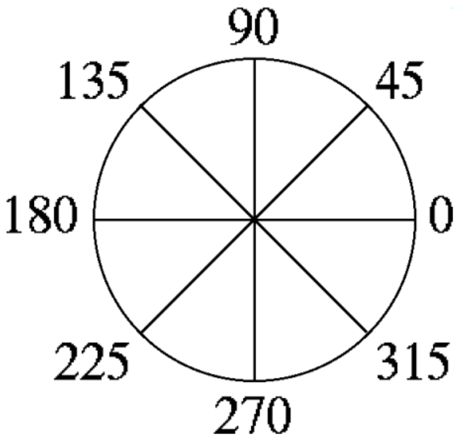
- Local object appearance and shape can often be characterized well using gradients.
- Specifically, the distribution of local intensity gradients or edge directions.



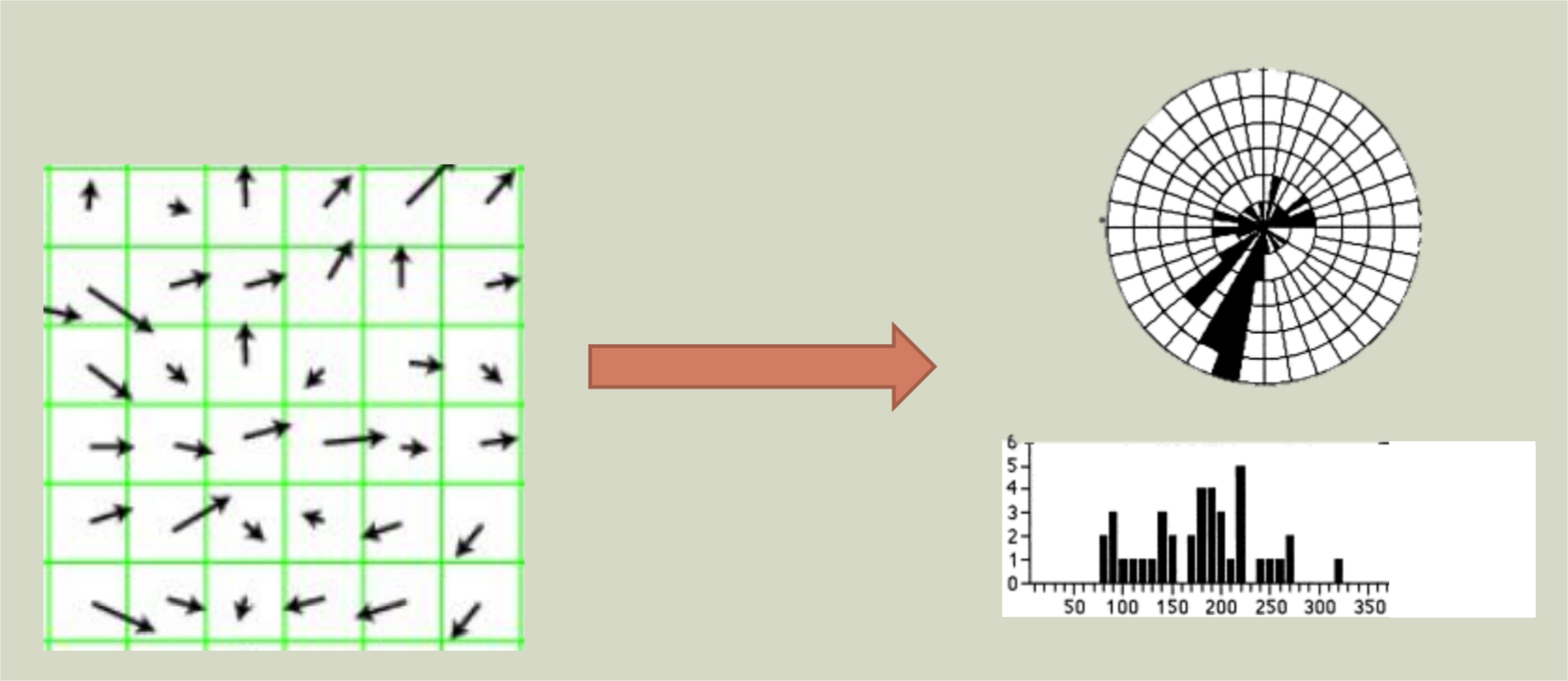
# Histogram of Oriented Gradients

- Dividing the image window into small spatial regions (cells)
- Cells can be either rectangle or radial.
- Each window sums up local 1-D histogram of gradient directions over the pixels of the cell.

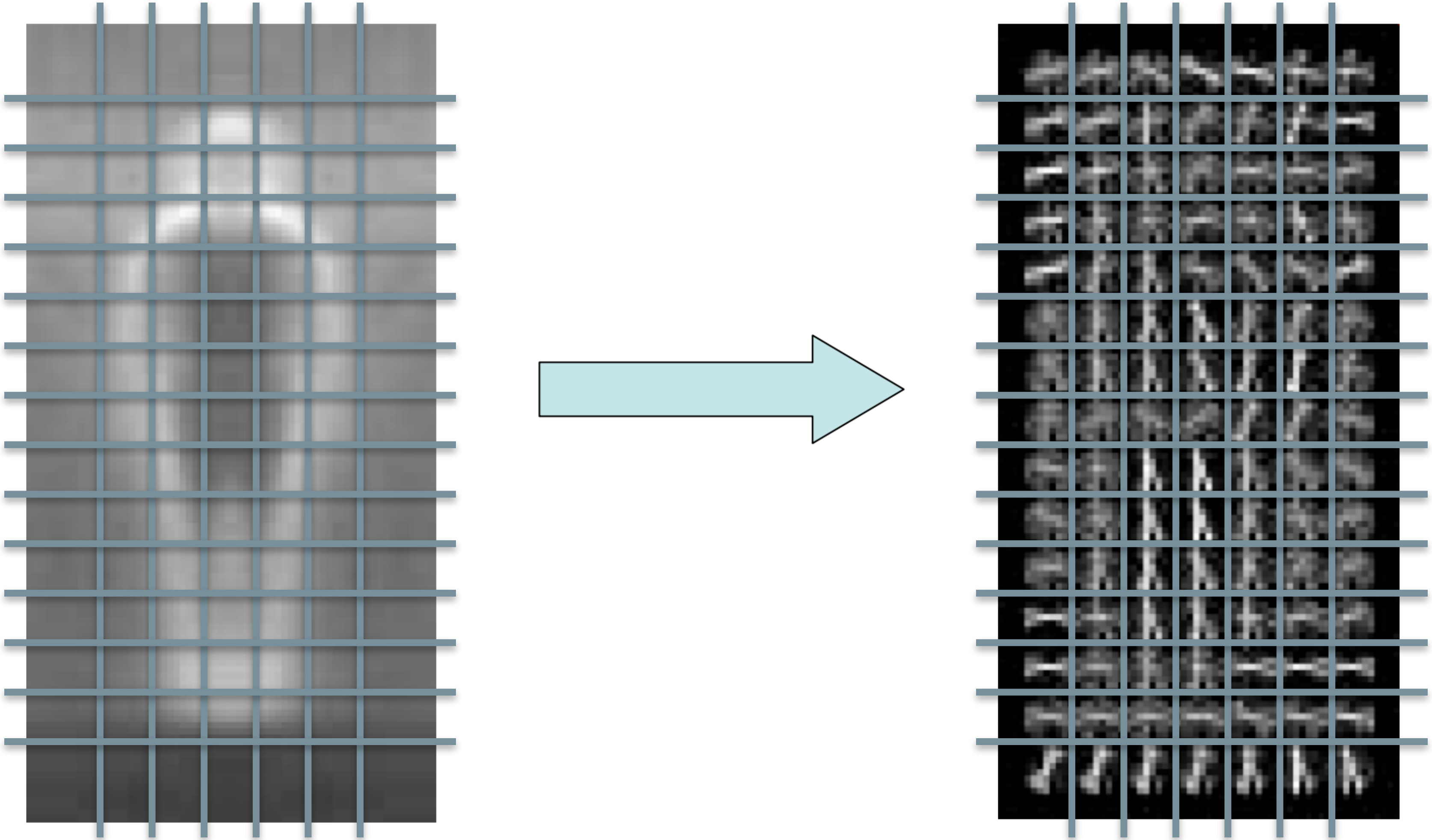




# Histogram of Oriented Gradients

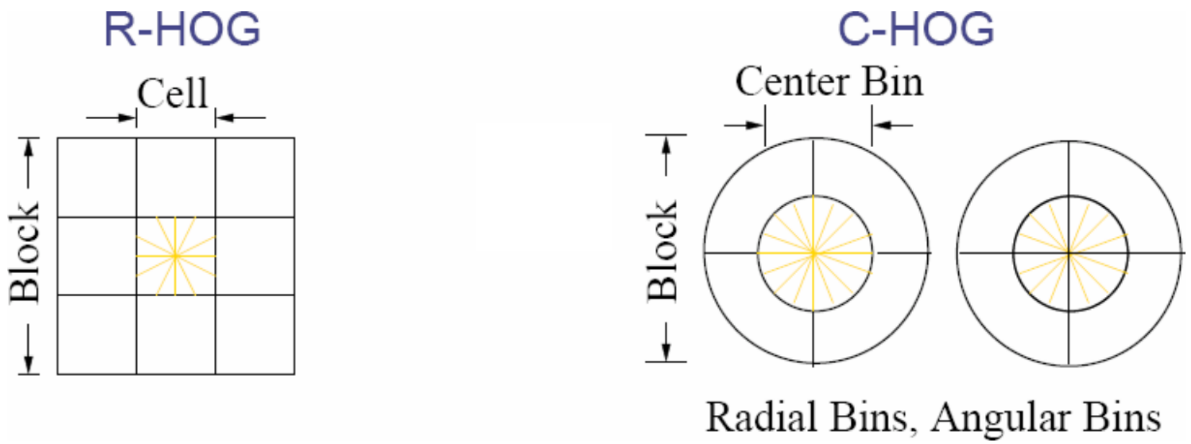


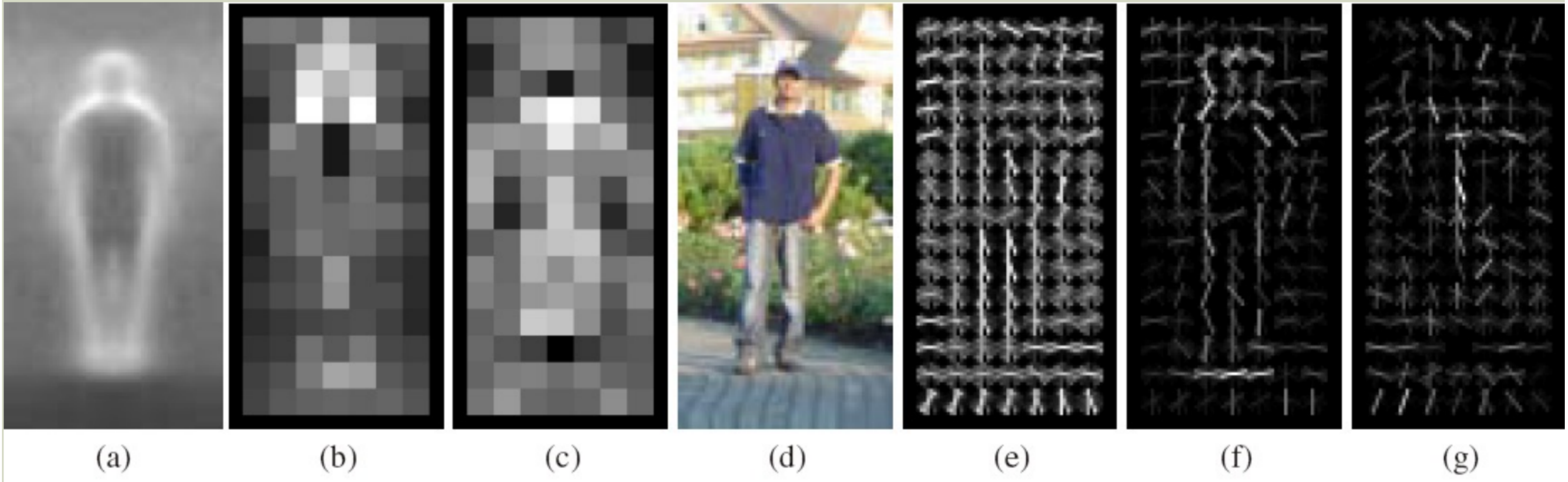
# Histogram of Oriented Gradients



# Normalization

- To make HoG invariant to illumination and shadows, it is useful to normalize the local responses
- Normalize each cell's histogram using histogram over a larger regions (“blocks”).

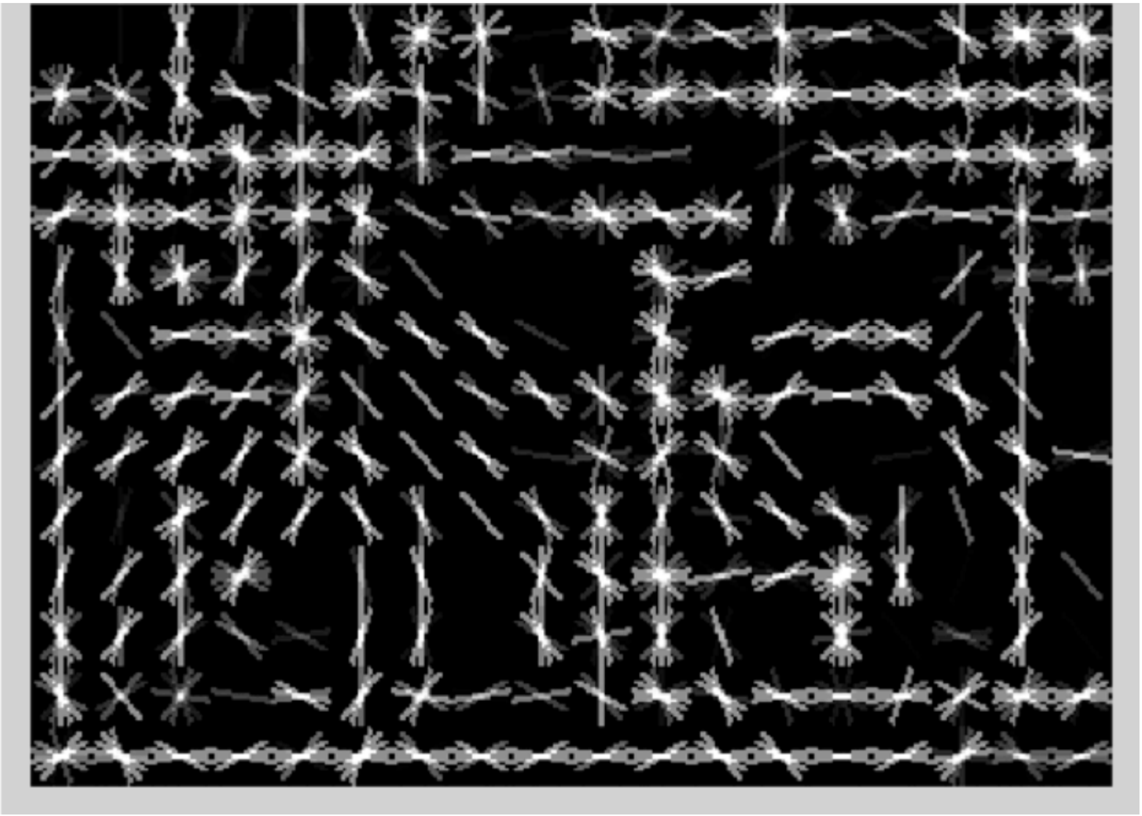




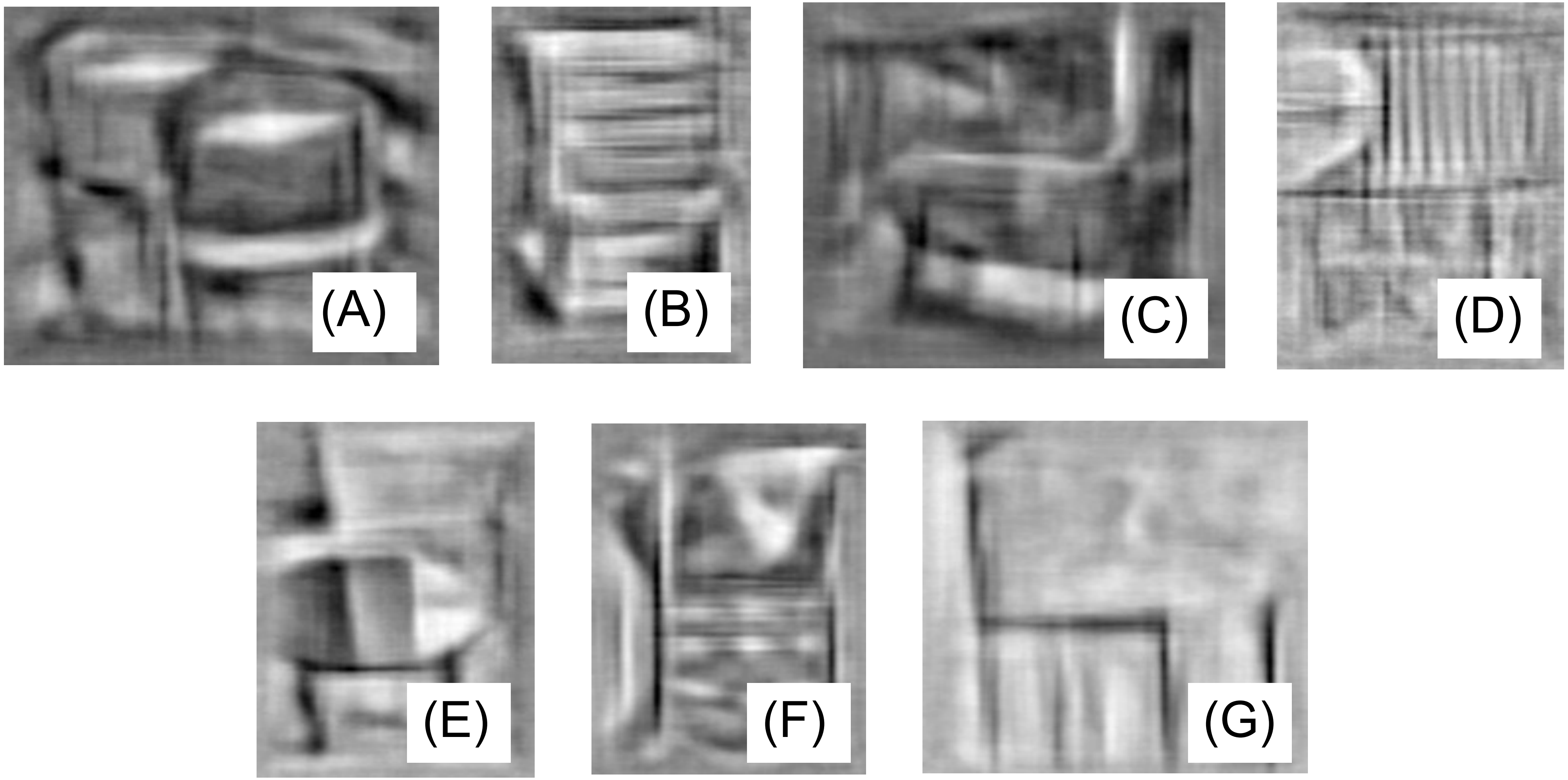

a. Average gradient over example photo of a person
b. “Positive” blocks that help match to other photos of people
c. “Negative” blocks that do not match to photos of other people
d. A test image
e. It’s HOG descriptor visualized
f. HOG descriptor weighted by positive weights
g. HOG descriptor weighted by negative weights

# Visualizing HoG

# Visualizing HoG



# Chair Detections

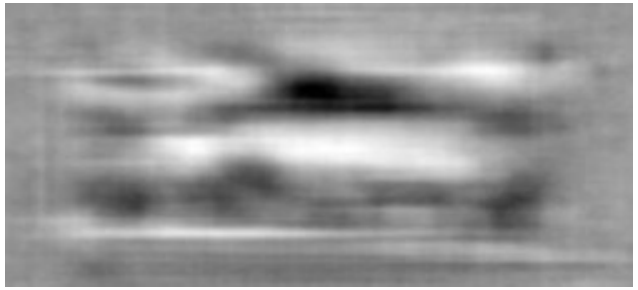


# Chair Detections


# Car Detections







# Car Detections





# Difference between HoG and SIFT

- HoG is usually used to describe larger image regions.
- SIFT is used for key point matching

- SIFT histograms are normalized with respect to the dominant gradient.
- HoG gradients are normalized using neighborhood blocks.

Q1: Is HoG rotation invariant?
Q2: Is HoG illumination invariant?

# Next time

Resizing image content