— CM50248 — 2017/2018 —

Visual Understanding 1

Dr Christian Richardt

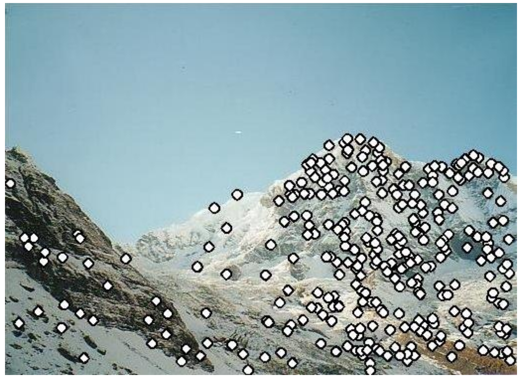
Today's lecture

- Recap:
  - features, descriptors, matching
- 2D planar geometry:
  - 2D transformations
  - Homographies
- RANSAC
- Image warping & interpolation
- Image stitching

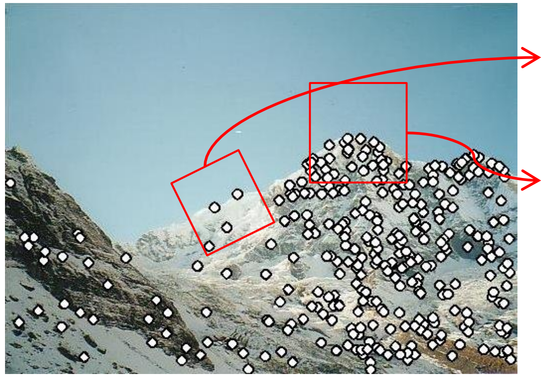
- Recommended reading:
  - CM50248 course notes:
    - 4.3.1 Matching: RANSAC
  - CM20219 course notes:
    - 4.1 2D rigid body transformations
    - 4.4 Homographies
    - 4.5 Digital image warping
  - FoCG textbook:
    - 2.6 Linear Interpolation
    - 6.1 2D Linear Transformations
    - 6.3 Translation and Affine Transformations

Recap: Local invariant features

1. **Detection:**
   Identify the interest points



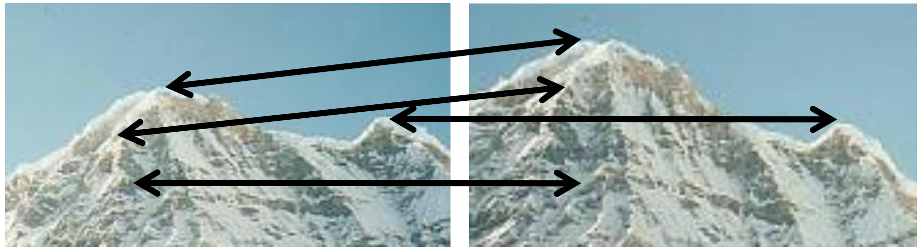
2. **Description:**
   Extract vector feature descriptor surrounding each interest point.



$$\mathbf{x}_1 = [x_1^{(1)}, \ldots, x_d^{(1)}]$$

$$\mathbf{x}_2 = [x_1^{(2)}, \ldots, x_d^{(2)}]$$

3. **Matching:**
   Determine correspondence between descriptors in 2 views



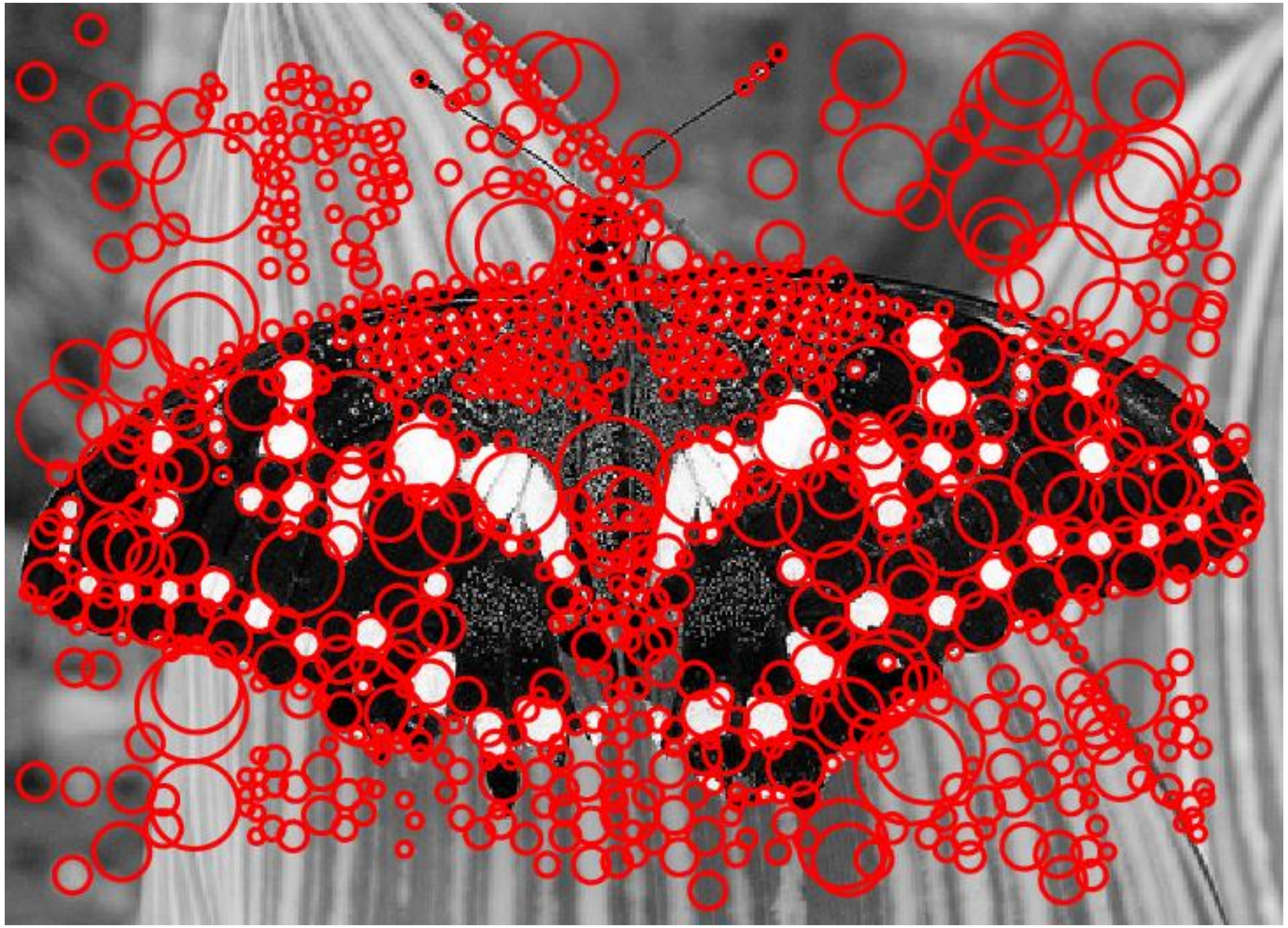
Slide by Kristen Grauman

Recap: Canny edges

Recap: Harris corners

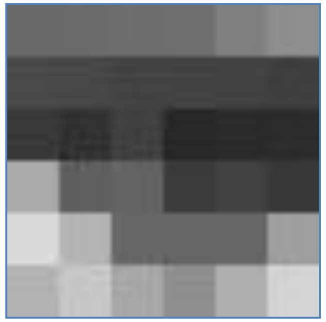
Recap: Scale-space blob detector



Slide by Noah Snavely

Recap: Descriptors

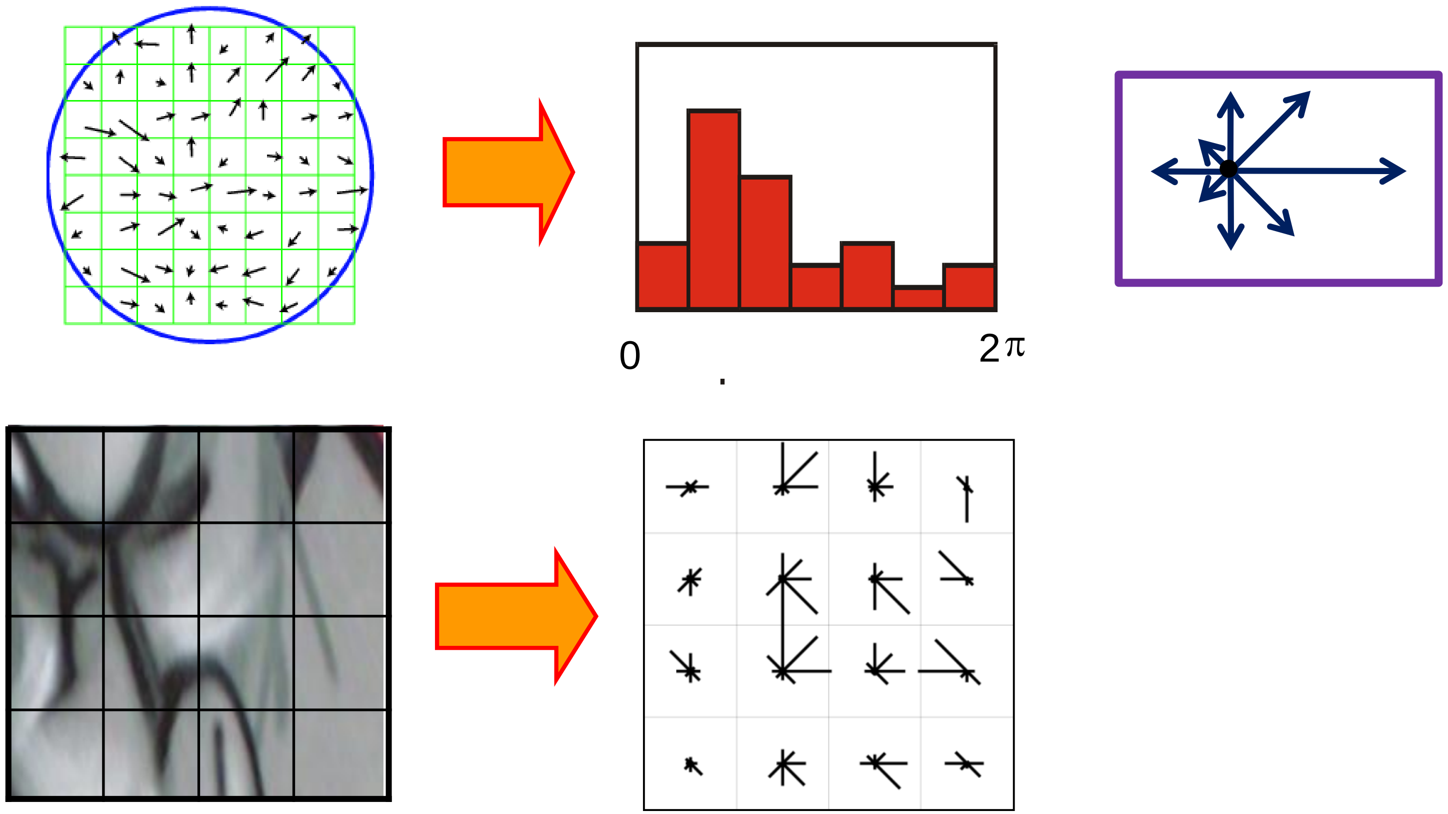
Compare two patches e.g. with sum of squared differences (SSD)



0.412	0.418	0.431	0.439	0.503	0.558
0.289	0.282	0.262	0.263	0.262	0.246
0.224	0.196	0.255	0.155	0.168	0.176
0.667	0.374	0.393	0.229	0.259	0.219
0.853	0.708	0.404	0.401	0.404	0.622
0.702	0.779	0.658	0.556	0.685	0.830

Recap: The SIFT descriptor

Use histograms to bin pixels within sub-patches according to their orientation.



Slide by James Hays

Recap: Ratio test (Lowe, 2004)

- Consider two best-matched features (m_1, m_2) to a feature f
- Ratio $r = \mathrm{d}(f, m_1) \,/\, \mathrm{d}(f, m_2)$
- If r is high, then f is
  - not unique (>1 match), or
  - not matched (random match)
- Rejects matches with $r > 0.8$
  - Eliminates 90% of false matches
  - Discards <5% of correct matches

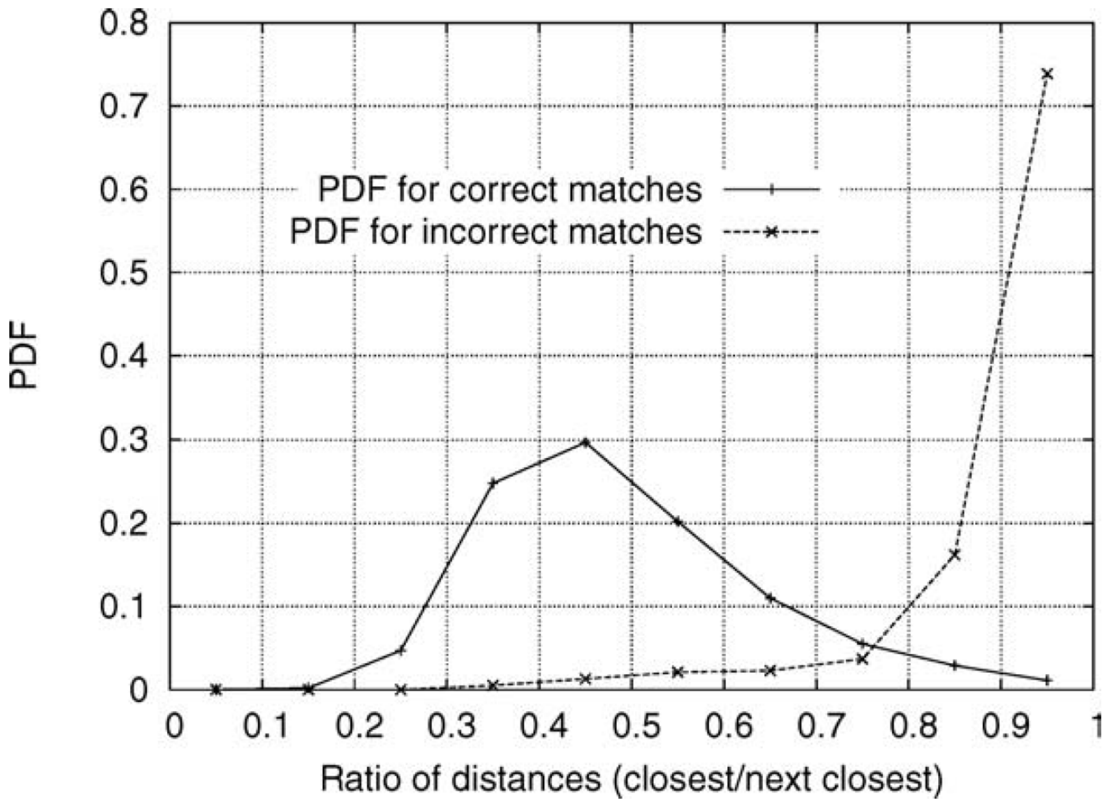
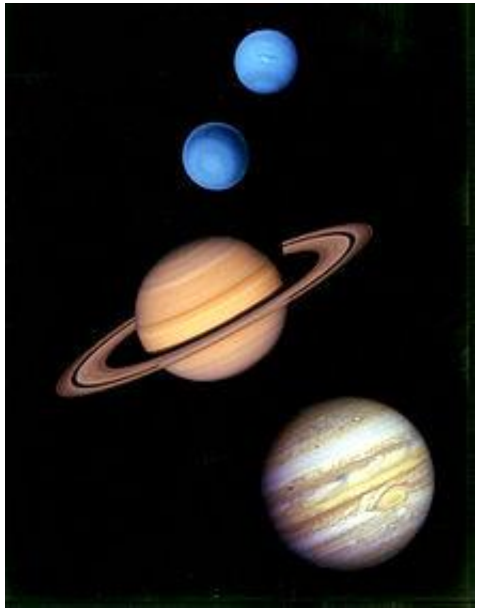


Figure 11. The probability that a match is correct can be determined by taking the ratio of distance from the closest neighbor to the distance of the second closest. Using a database of 40,000 keypoints, the solid line shows the PDF of this ratio for correct matches, while the dotted line is for matches that were incorrect.

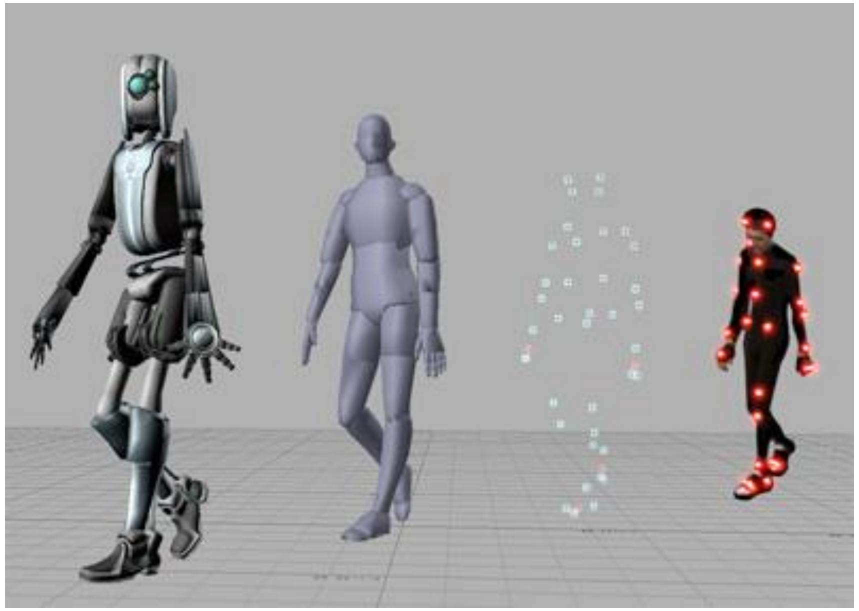
Lowe (IJCV 2004)

Planar geometry

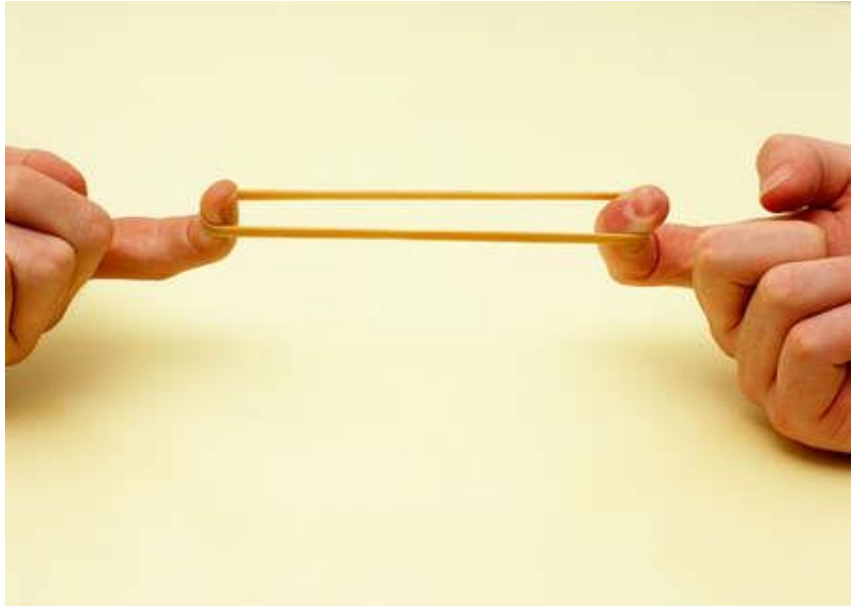
Types of transforms



Rigid



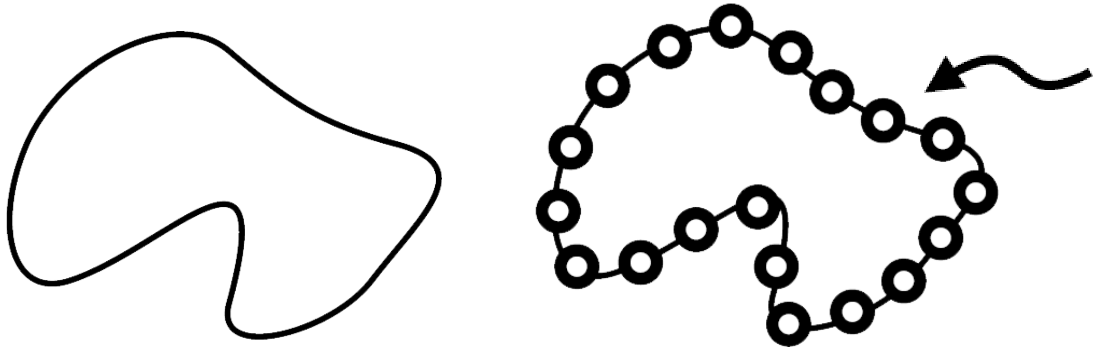
Articulated
(piecewise rigid)



Non-Rigid

2D transforms

- Consider a arbitrary shape in 2D
- We could sample this continuous shape with points:



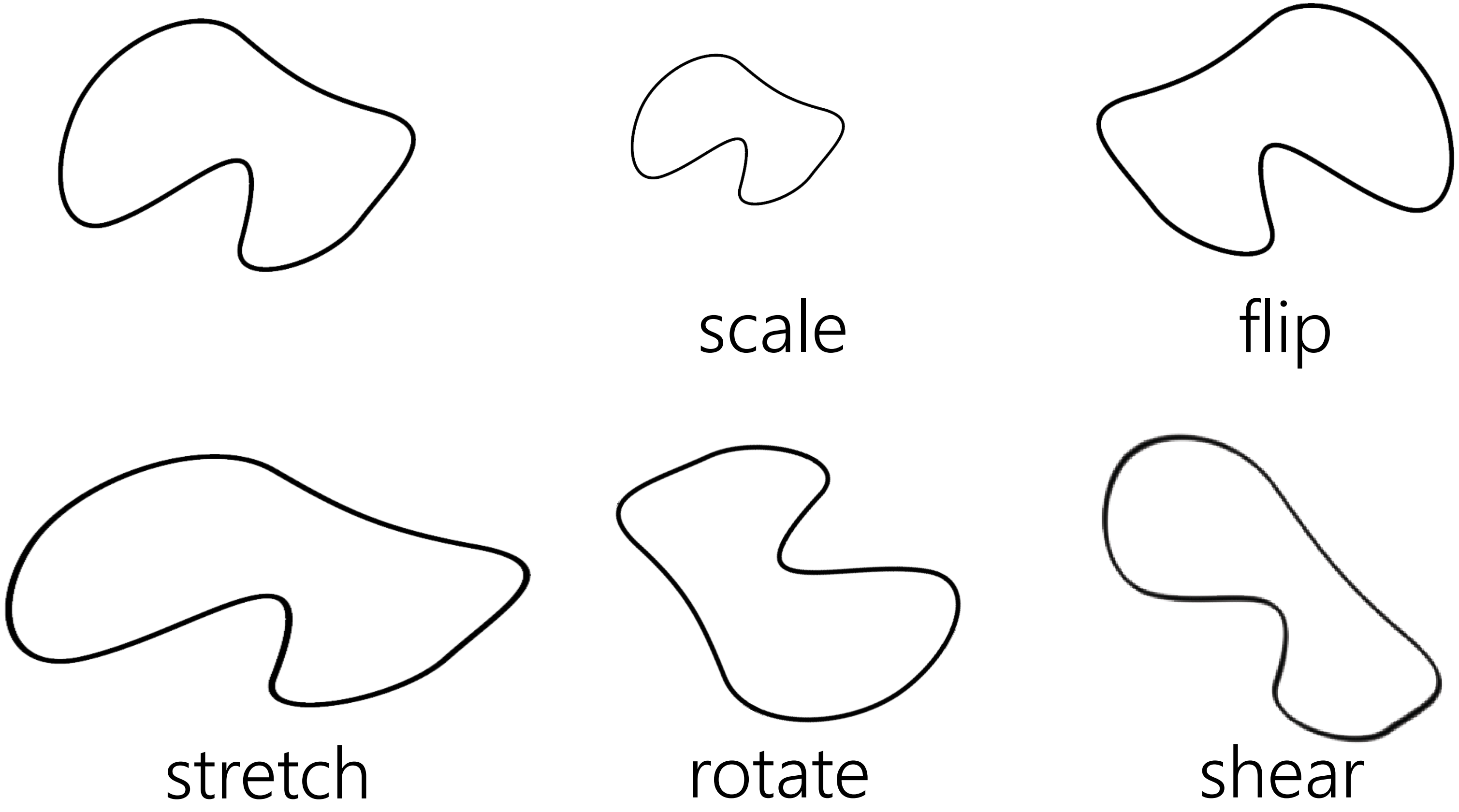
$$\mathbf{p}_1 = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_n \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \end{bmatrix}$$

- We can transform the shape using matrix operations, e.g. $\mathbf{p}' = \mathbf{M}\mathbf{p}$
- **M** is a 2×2 matrix with 4 degrees of freedom

2D matrix transforms

- What transformations can be represented by: $\mathbf{p}' = \mathbf{M}\mathbf{p}$



2D matrix transforms

Scale

$$x' = \lambda x \quad y' = \lambda y \quad \Longrightarrow \quad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

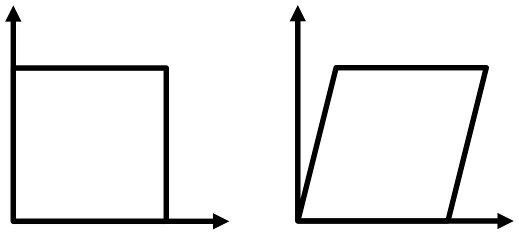
Stretch

$$x' = \alpha x \quad y' = \beta y \quad \Longrightarrow \quad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Flip horizontally

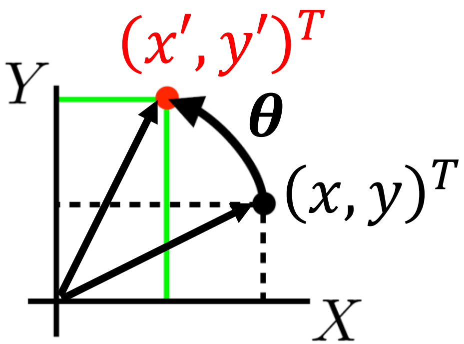
$$x' = -x \quad y' = y \quad \Longrightarrow \quad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

2D matrix transforms



Shear / Skew

$$x' = x + ky \quad y' = y \quad \Longrightarrow \quad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

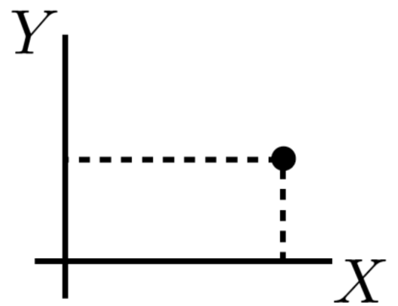


Rotation

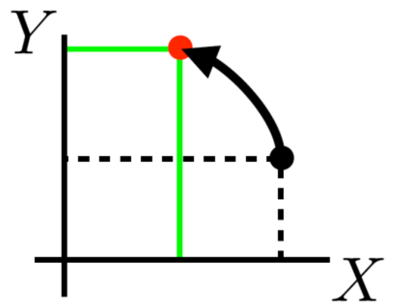
$$\begin{aligned} x' &= x\cos\theta - y\sin\theta \\ y' &= x\sin\theta + y\cos\theta \end{aligned} \quad \Longrightarrow \quad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \mathbf{p}$$

Active vs passive interpretation

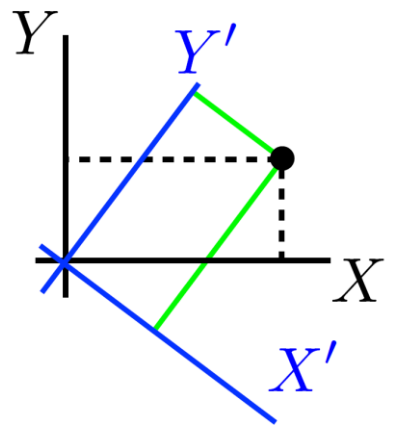
- We can think of a matrix transformation in 2 ways
  - 1) points are transformed in the original coordinate system
  - 2) points remain fixed and the coordinate system moves



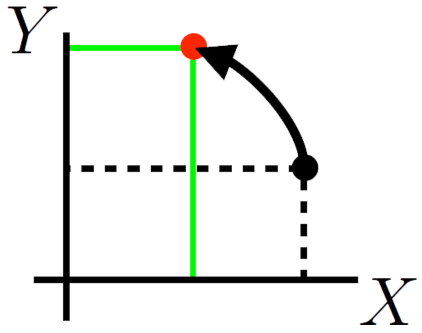
- e.g., for the matrix $\mathbf{M} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$

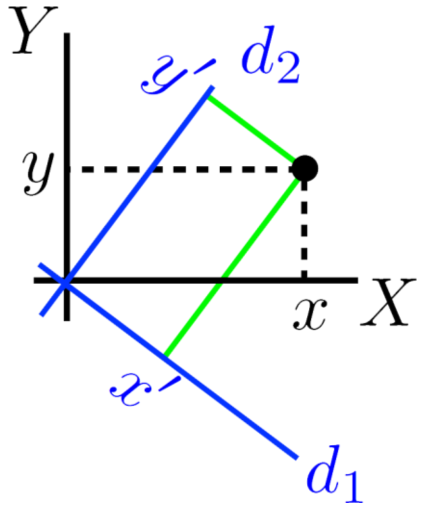


- we can think of transforming the points by **M**,
- or transforming the axes (basis vectors) by $\mathbf{M}^{-1}$



Active vs passive interpretation





$$\mathbf{M} = \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix}$$

$$\mathbf{M}^{-1} = \begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{M} \begin{bmatrix} x \\ y \end{bmatrix}$$
$$= x\mathbf{c}_1 + y\mathbf{c}_2$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{M}^{-1} \begin{bmatrix} x' \\ y' \end{bmatrix}$$
$$= x'\mathbf{d}_1 + y'\mathbf{d}_2$$

What about translation?

- Translation is vector addition

$$\begin{bmatrix} p'_1 \\ p'_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

- Can we represent translation as a linear operation? (matrix multiplication)

Homogeneous coordinates

- Yes, if we add a "1" to the vector

$$\begin{bmatrix} p_1' \\ p_2' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ 1 \end{bmatrix}$$

- These are called homogeneous coordinates

$$\begin{bmatrix} p_1' \\ p_2' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ 1 \end{bmatrix}$$

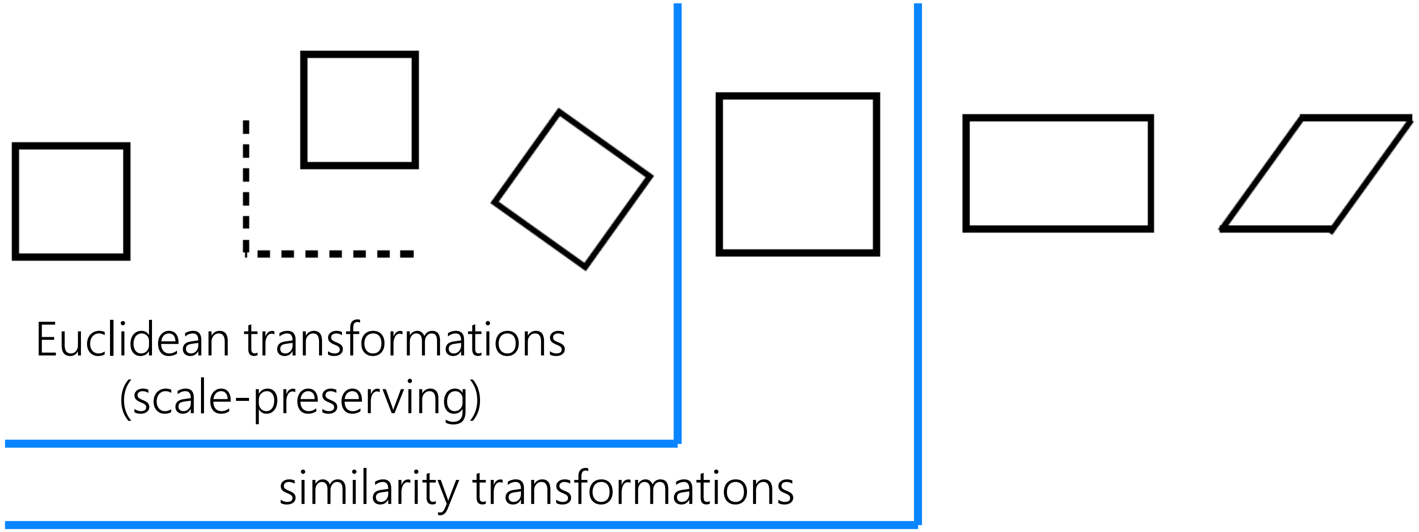
Translation is now a linear operation ...

Affine transformations

- What range of transformations can we represent using this?

$$\begin{bmatrix} p'_1 \\ p'_2 \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ 1 \end{bmatrix}$$

- 6 degrees of freedom: translation (2), rotation, scale, stretch, shear



Example affine transformations

Stretch

$$\begin{bmatrix} \alpha & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Translation

$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Compound transformations

- Translation using $\mathbf{T}$: $\mathbf{p}' = \mathbf{T}\,\mathbf{p}$
- Scale using $\mathbf{S}$: $\mathbf{p}'' = \mathbf{S}\,\mathbf{p}'$

- Matrix multiplication is associative, we have $$\begin{aligned}\mathbf{p}'' &= \mathbf{S}\,\mathbf{p}' \\ &= \mathbf{S}(\mathbf{T}\,\mathbf{p}) \\ &= (\mathbf{S}\,\mathbf{T})\mathbf{p}\end{aligned}$$

- The compound transformation is $\mathbf{M} = \mathbf{S}\,\mathbf{T}$,
  which transforms points using $\mathbf{p}'' = \mathbf{M}\,\mathbf{p}$

Compound transformations

- Homogeneous transformations are easily concatenated

$$\mathbf{p}' = \mathbf{T}_3\mathbf{T}_2\mathbf{T}_1\mathbf{p}$$

- Note that the order of operations matters, in general

$$\mathbf{T}_1\mathbf{T}_2 \neq \mathbf{T}_2\mathbf{T}_1$$

- e.g., rotate then translate versus translate then rotate

$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & t_x \\ -\sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & t_x\cos\theta + t_y\sin\theta \\ -\sin\theta & \cos\theta & -t_x\sin\theta + t_y\cos\theta \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation about an arbitrary point

1. Translate centre of rotation to origin:

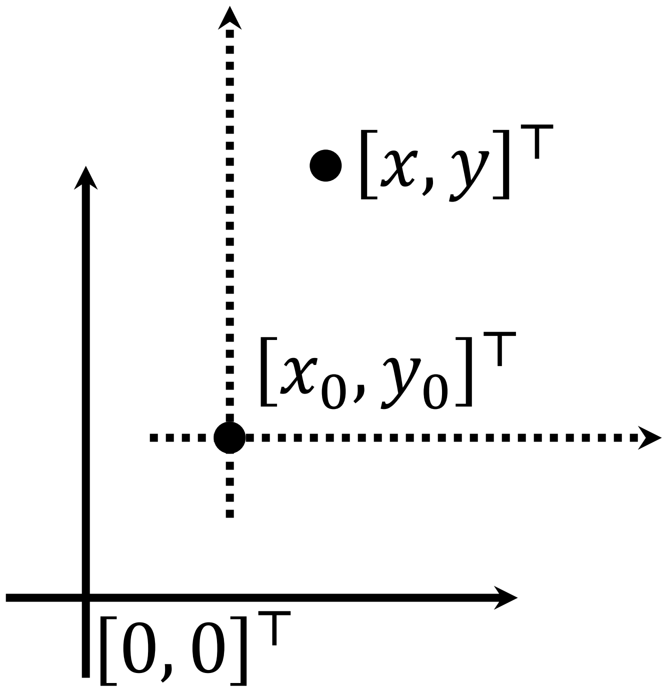
$$\mathbf{T} = \begin{bmatrix} 1 & 0 & -x_o \\ 0 & 1 & -y_o \\ 0 & 0 & 1 \end{bmatrix}$$

2. Rotate about origin:

$$\mathbf{R} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3. Translate centre of rotation back:

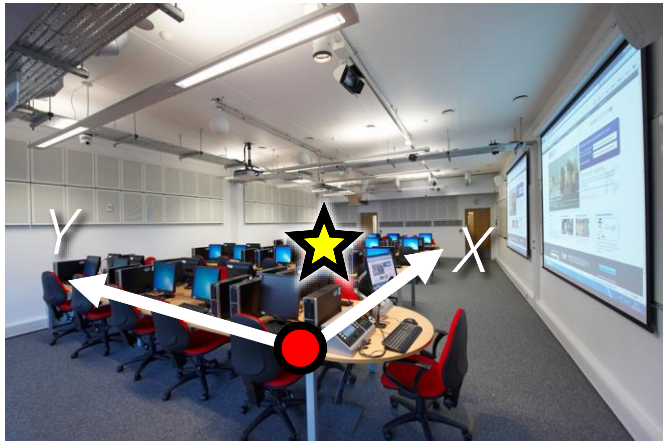
$$\mathbf{T}^{-1} = \begin{bmatrix} 1 & 0 & x_o \\ 0 & 1 & y_o \\ 0 & 0 & 1 \end{bmatrix}$$



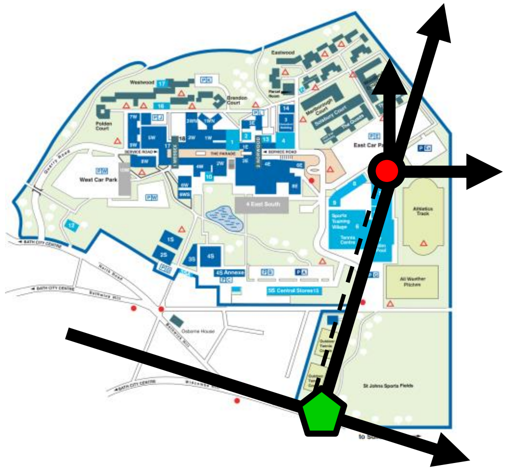
Compound transform:

$$\mathbf{p}' = (\mathbf{T}^{-1}\,\mathbf{R}\,\mathbf{T})\,\mathbf{p}$$

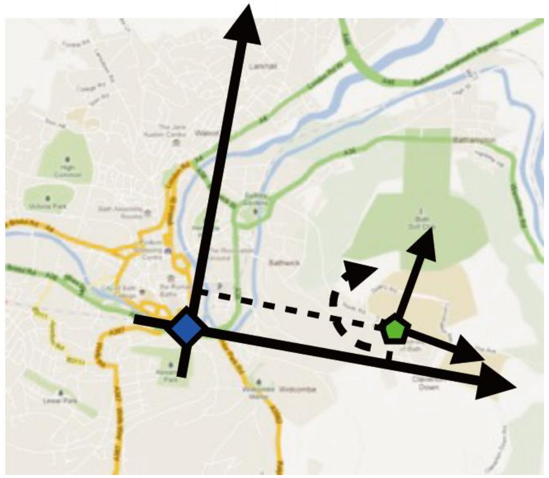
Local/global coordinates



row 3, seat 4



EB 0.9



University of Bath

$$\mathbf{p}_r = \begin{bmatrix} 4 \\ 3 \end{bmatrix}$$

(room coordinate)

$$\mathbf{p}_c = \mathbf{R}_1 \mathbf{p}_r + \mathbf{t}_1$$

$$\tilde{\mathbf{p}}_c = \mathbf{T}_1 \tilde{\mathbf{p}}_r$$

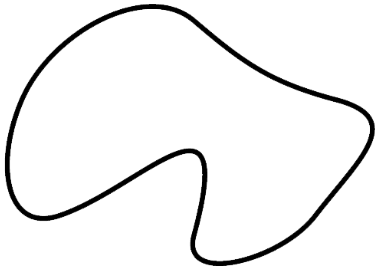
(campus coordinate)

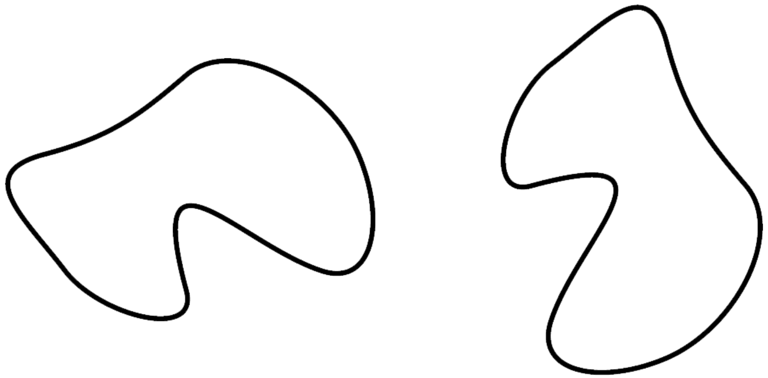
$$\begin{aligned} \mathbf{p}_b &= \mathbf{R}_2 \mathbf{p}_c + \mathbf{t}_2 \\ &= \mathbf{R}_2(\mathbf{R}_1 \mathbf{p}_r + \mathbf{t}_1) + \mathbf{t}_2 \end{aligned}$$

$$\boxed{\tilde{\mathbf{p}}_b = \mathbf{T}_2 \mathbf{T}_1 \tilde{\mathbf{p}}_r}$$

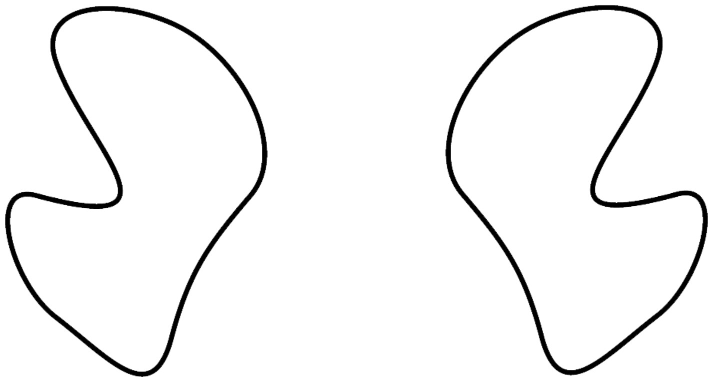
Order matters!

- Note that for matrices $\mathbf{AB} \neq \mathbf{BA}$ in general





Flip Vertical Rotate 90° CW



Rotate 90° CW Flip Vertical

2D Projective Transforms =

Homographies

Projective transformations

- What if we fill in the bottom row?

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ 0 & 0 & 1 \end{bmatrix} \longrightarrow \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}$$

- Transformation equations become

$$s \begin{bmatrix} p'_1 \\ p'_2 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ 1 \end{bmatrix}$$

- Note: scale factor *s* – bottom row is no longer = 1

Projective transformations

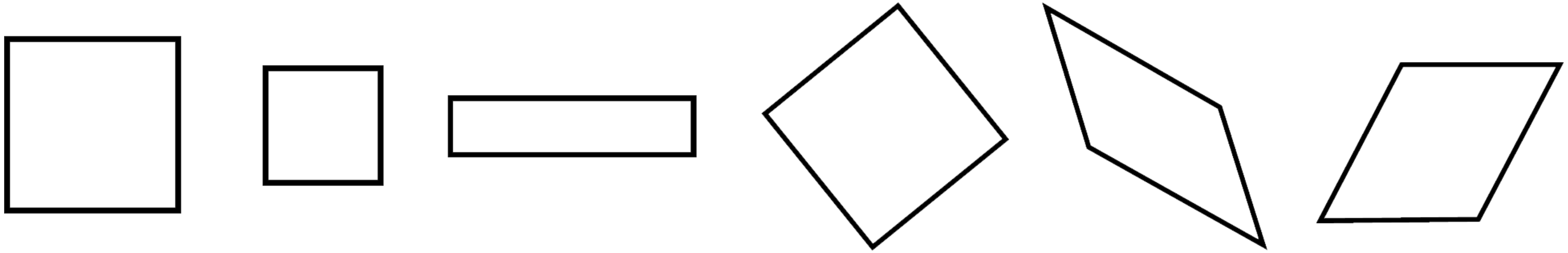
- 3 linear equations:

$$
\begin{aligned}
sp'_1 &= m_{11}p_1 + m_{12}p_2 + m_{13} \\
sp'_2 &= m_{21}p_1 + m_{22}p_2 + m_{23} \\
s &= m_{31}p_1 + m_{32}p_2 + m_{33}
\end{aligned}
$$

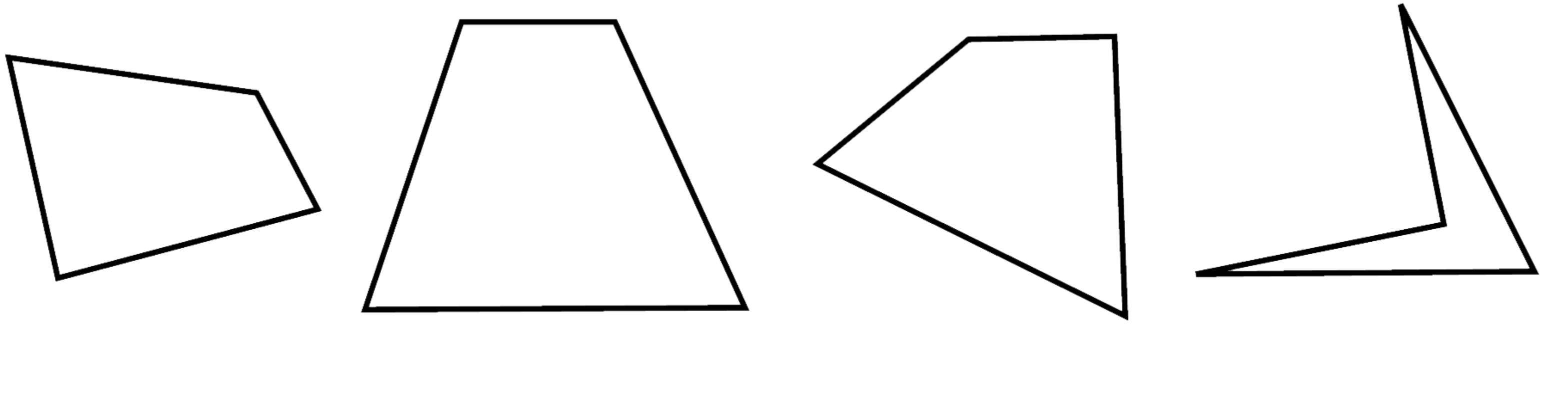
- Note: the overall scale of **M** does not affect *p*′

Examples of projective transformations

Includes all of the affine group (parallel lines preserved) ...



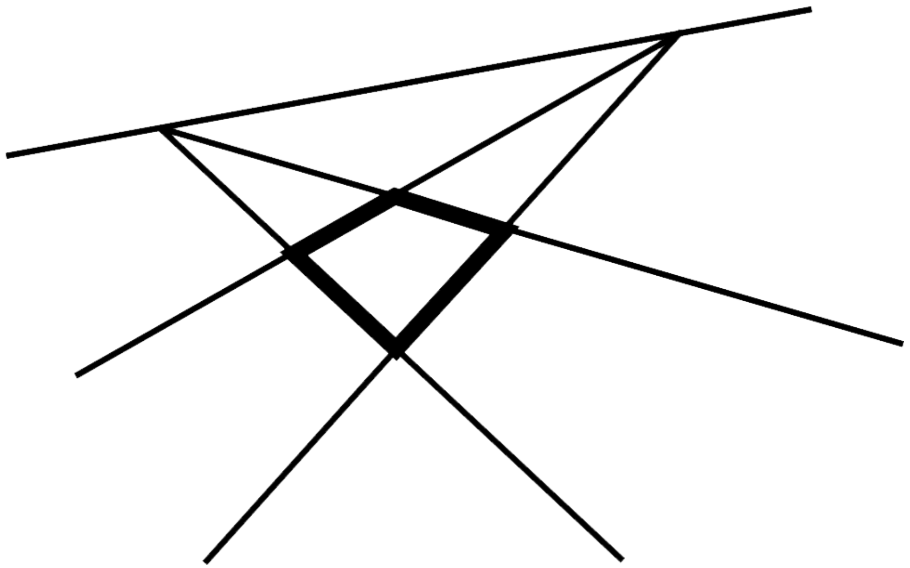
... plus non-parallel quadrilaterals (some not physically plausible)



Homographies

- Linear transform in homogeneous coordinates
- Represented by 3x3 matrix:

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$



- 8 degrees of freedom (scale of **H** arbitrary)
- 4 points can map to any 4 points
- Includes translation (2), rotation, scale, stretch, shear

Image alignment

- Aim: warp our images together using a 2D transformation

source image

target image

Image alignment

- Aim: warp our images together using a 2D transformation

Image alignment

- Find corresponding (matching) points between the images

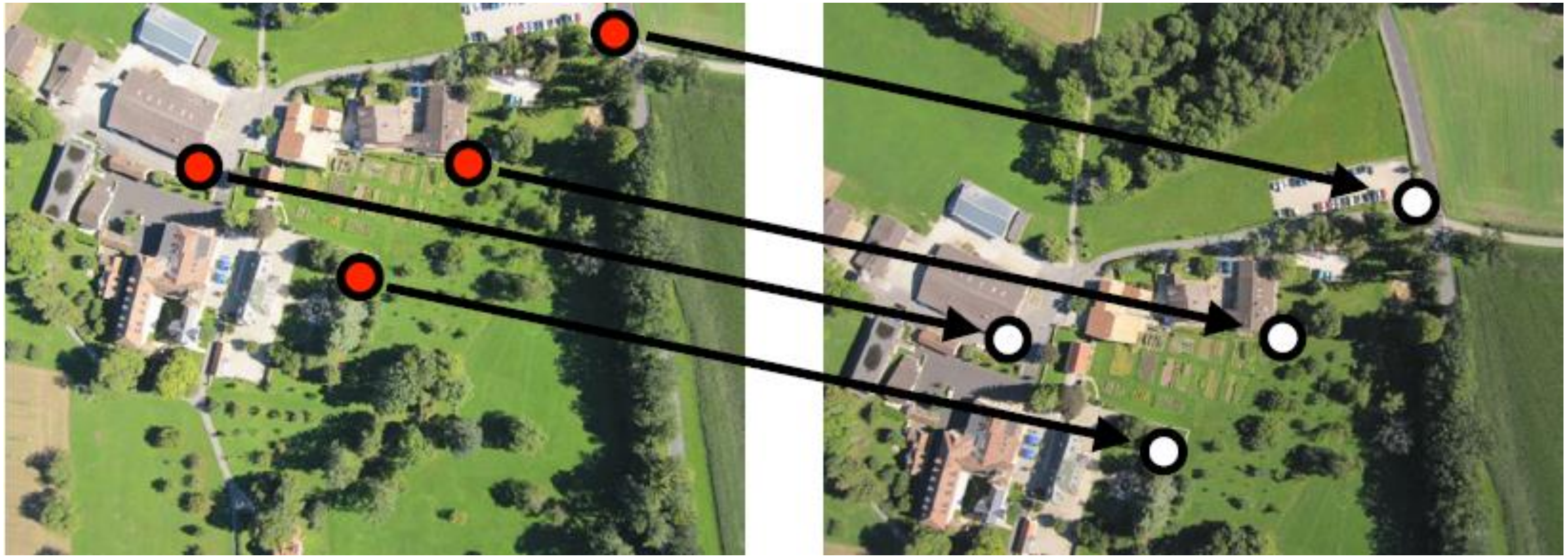
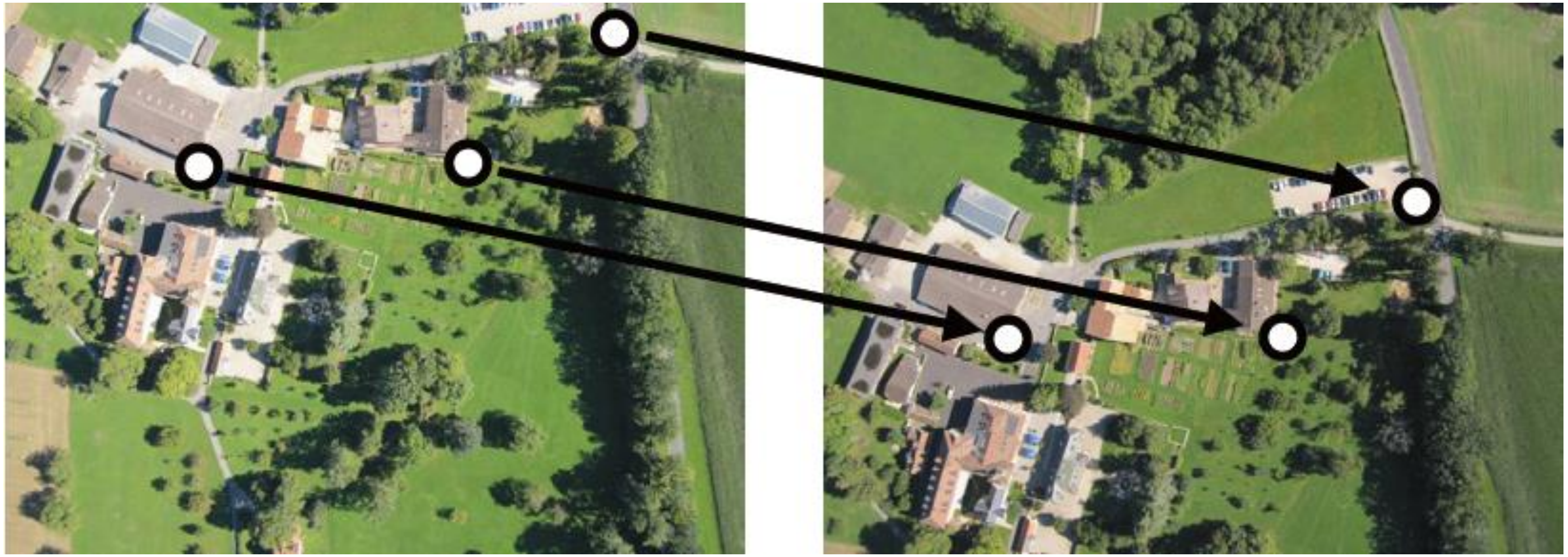


Image alignment

- Compute the transformation to align the points

Computing 2D transforms

Q: How many points are needed?

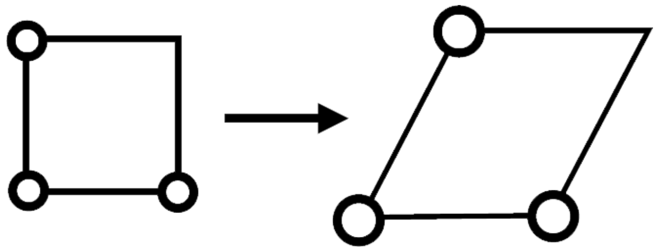


Computing 2D transforms

- Q: How many points are needed?

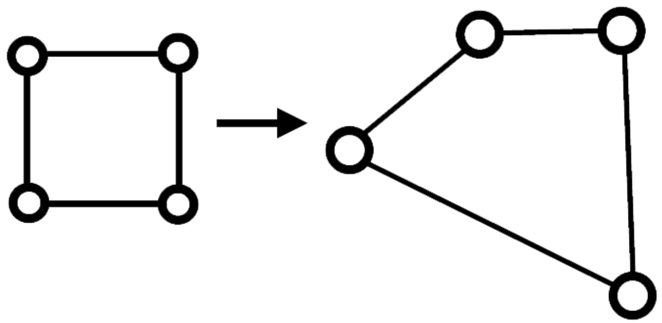
$$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

Affine transform
6 degrees of freedom
3 points to specify



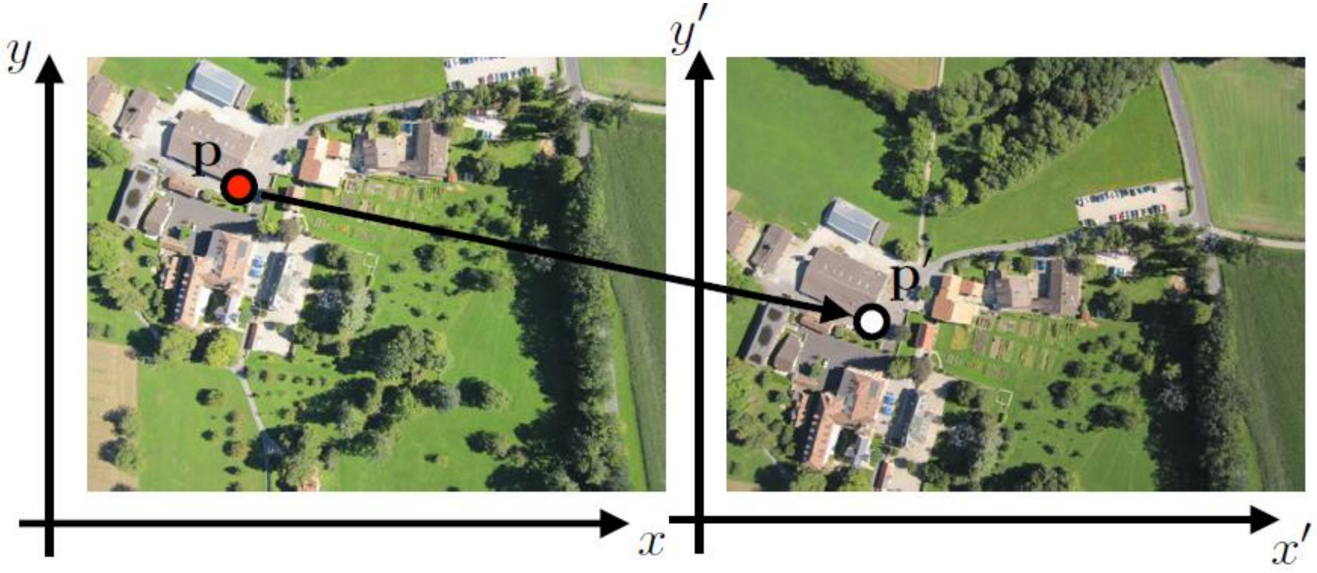
$$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}$$

Homography
8 degrees of freedom
4 points to specify



Computing affine transforms

- Consider a single point correspondence



$$\begin{bmatrix} x_1' \\ y_1' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

Each correspondence gives 2 equations in 6 unknowns

Computing affine transforms

- Affine transform equation

$$\begin{bmatrix} x_1' \\ y_1' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

- Re-arrange unknowns into a vector

Computing affine transforms

- Linear system in the unknown parameters **a**:

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ x_3 & y_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_3 & y_3 & 1 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \end{bmatrix}$$

- Of the form $\mathbf{Ma} = \mathbf{y}$
- Solve for **a** using Gaussian elimination

Computing affine transforms

- We can now map any other points between the two images

$$\begin{bmatrix} x'_1 \\ y'_1 \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

Computing affine transforms

- Transform one image in the coordinate system of the other

This allows us to "stitch" the two images

Computing homographies (projective transforms)

- Inhomogenous solution:
  - Similar to affine approach (few slides earlier)
  - But set m_{33}=1 to fix scale of homography
  - Incorrect when m_{33}~0

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}$$

- Homogeneous solution using direct linear transform:
  - Cancel out homogenous scale using $\mathbf{x}'_i \times \mathbf{H}\mathbf{x}_i = 0$
  - Expand out, rearrange, drop an equation, stack equations: $\mathbf{A}\mathbf{h} = \mathbf{0}$
  - Minimising $||\mathbf{A}\mathbf{h}|| = 0$ subject to $||\mathbf{h}|| = 1$
  - Solution: unit singular value corresponding to smallest singular value of $\mathbf{A}$
  - (for details, see Hartley & Zisserman, 2004)

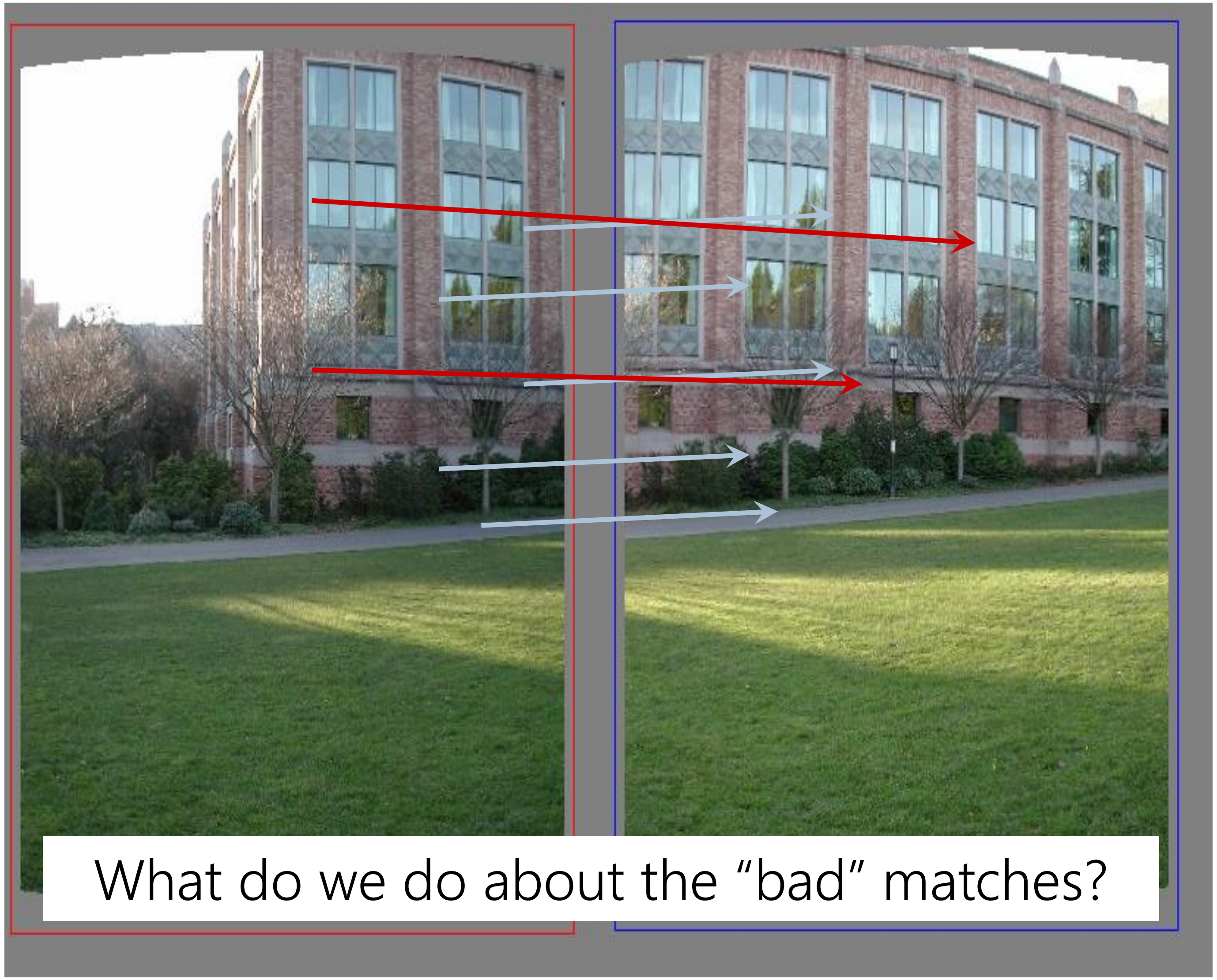
RANSAC

RANSAC

1. Select a random subset of the original data.
   Call this subset the *hypothetical inliers*.
2. A model is fitted to the set of hypothetical inliers.
3. All other data are tested against the fitted model.
   Points that fit the estimated model well (according to some model-specific loss function), are considered as part of the *consensus set*.
4. The estimated model is reasonably good if sufficiently many points have been classified as part of the consensus set.
5. Afterwards, the model may be improved by reestimating it using all members of the consensus set.

Source: https://en.wikipedia.org/wiki/Random_sample_consensus

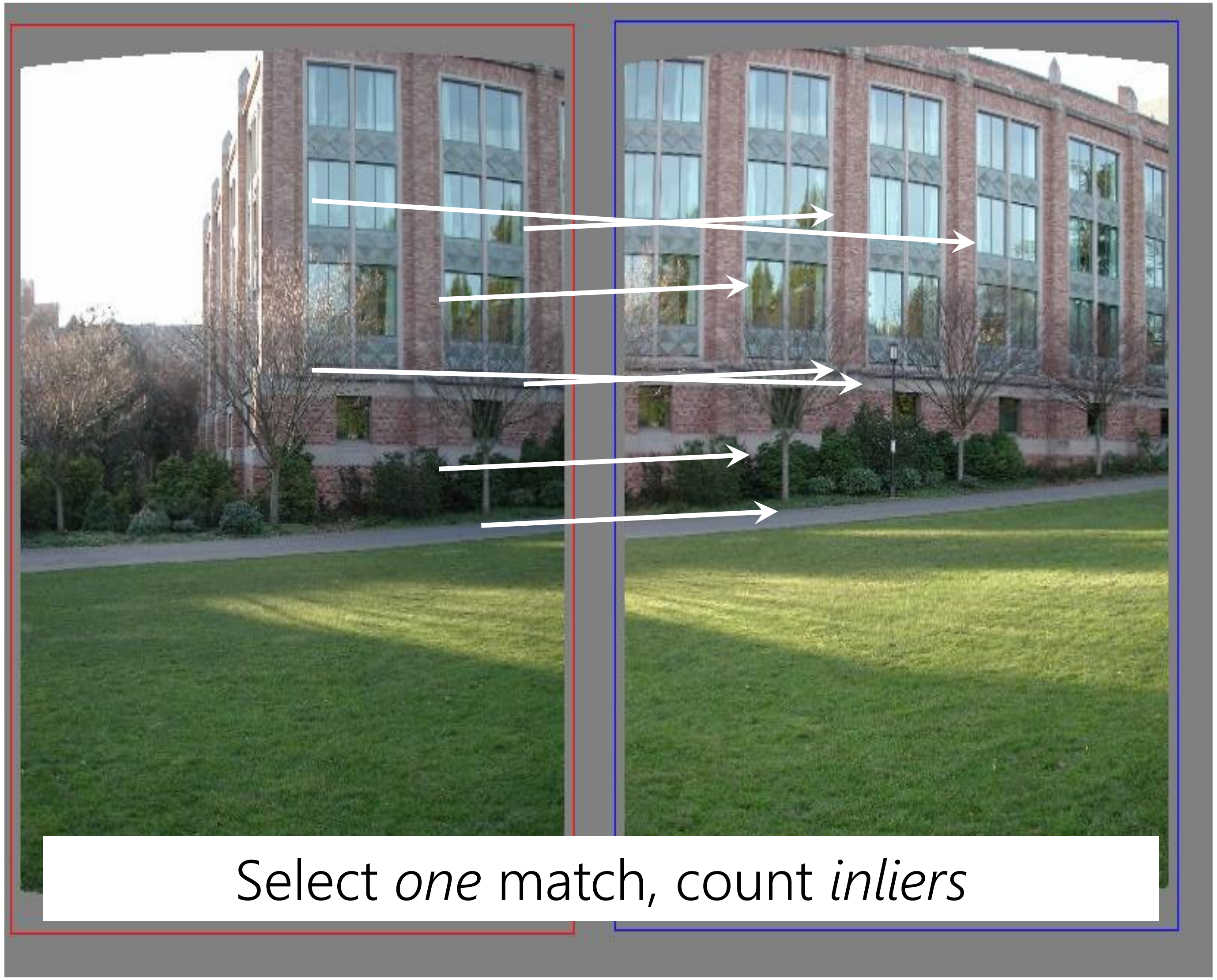
Matching features



What do we do about the “bad” matches?

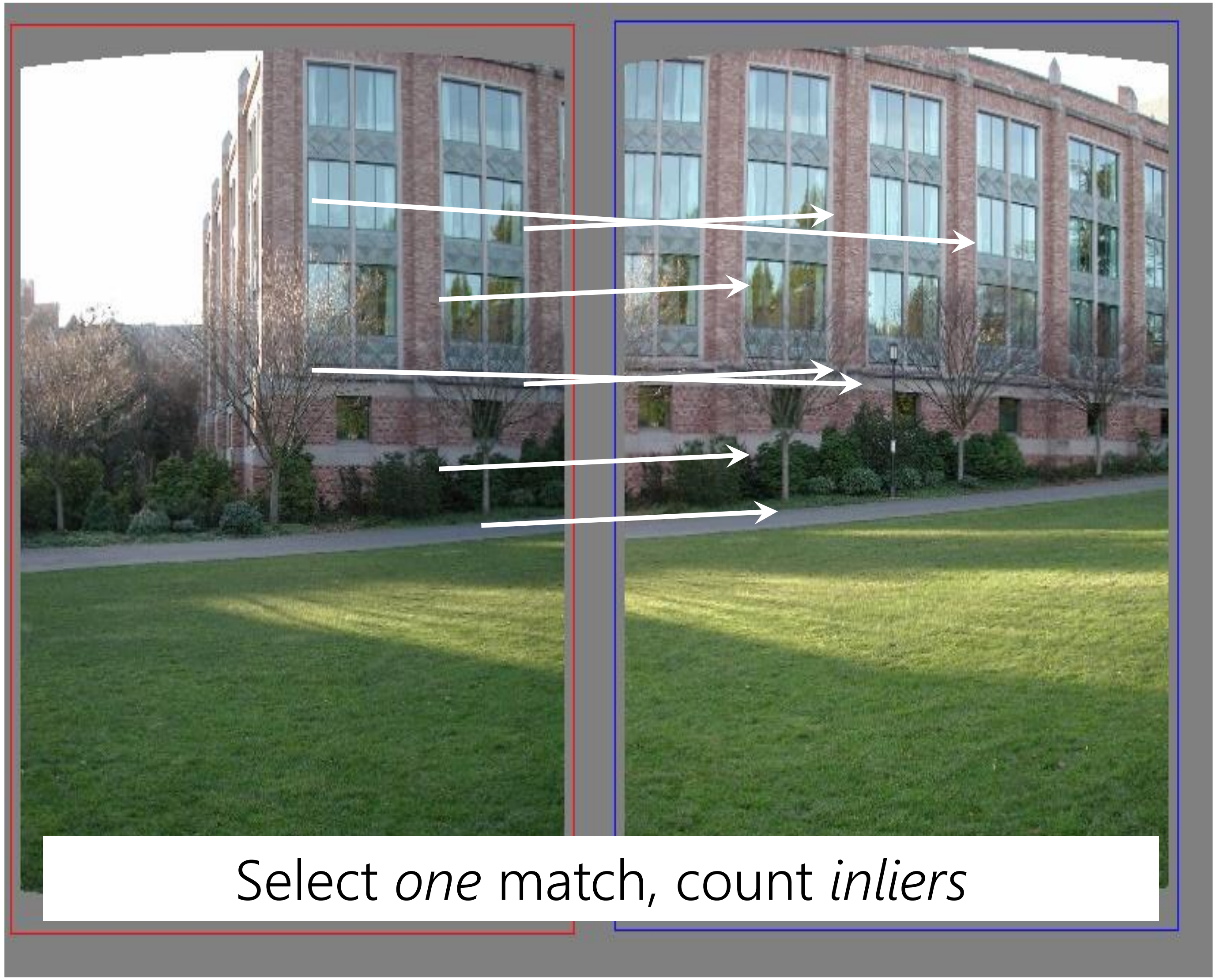
Slide by Alexei Efros, Rick Szeliski and/or Steve Seitz

RAndom SAmple Consensus



Slide by Alexei Efros, Rick Szeliski and/or Steve Seitz

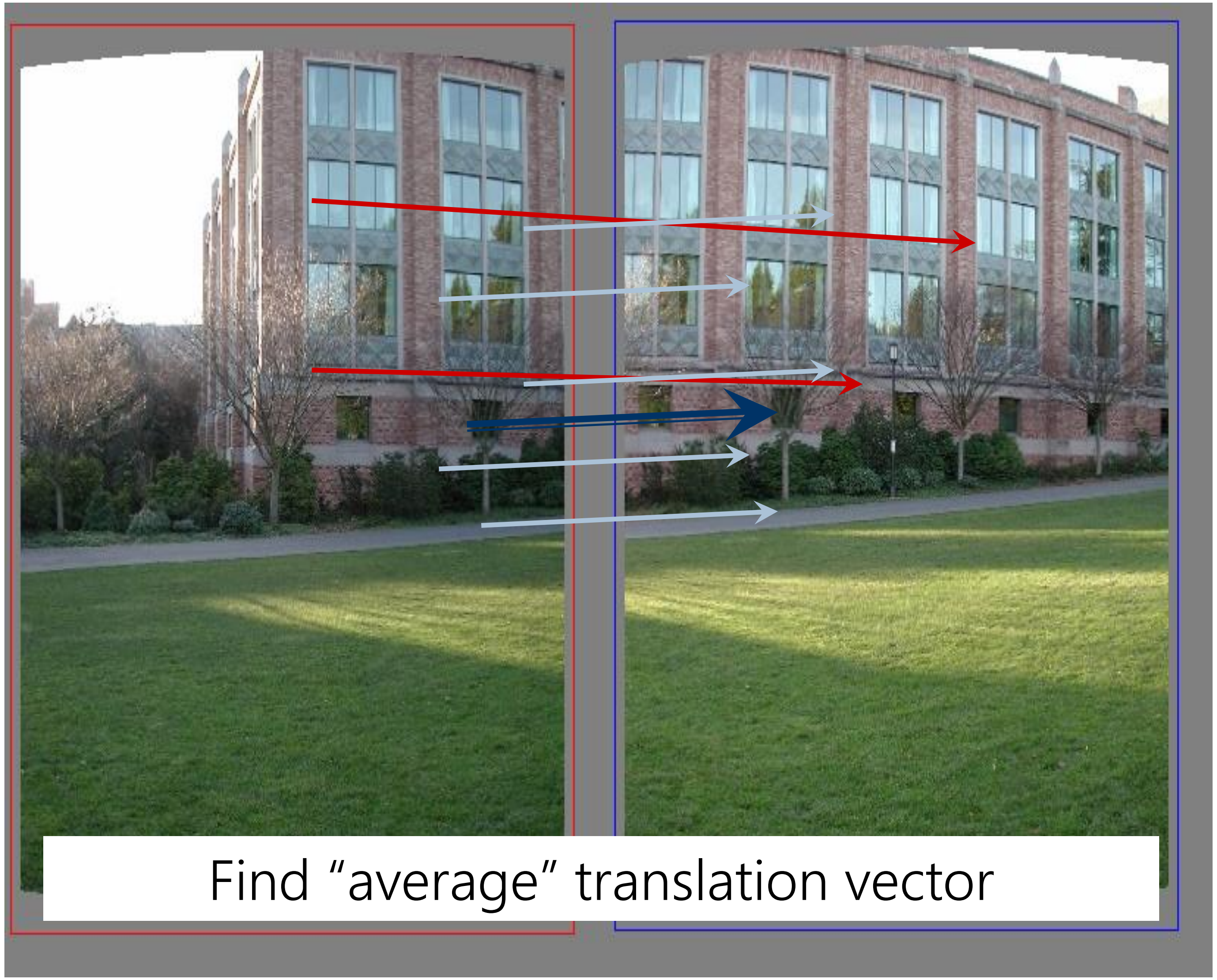
RAndom SAmple Consensus



Select *one* match, count *inliers*

Slide by Alexei Efros, Rick Szeliski and/or Steve Seitz

Least-squares fit



Find “average” translation vector

Slide by Alexei Efros, Rick Szeliski and/or Steve Seitz

RANSAC for homography estimation

RANSAC loop:

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

Image stitching

Image warping

- How do we warp images given a known affine transform?

Forward mapping

$$\tilde{\mathbf{p}}' = \mathbf{T}\tilde{\mathbf{p}}$$

Backward mapping

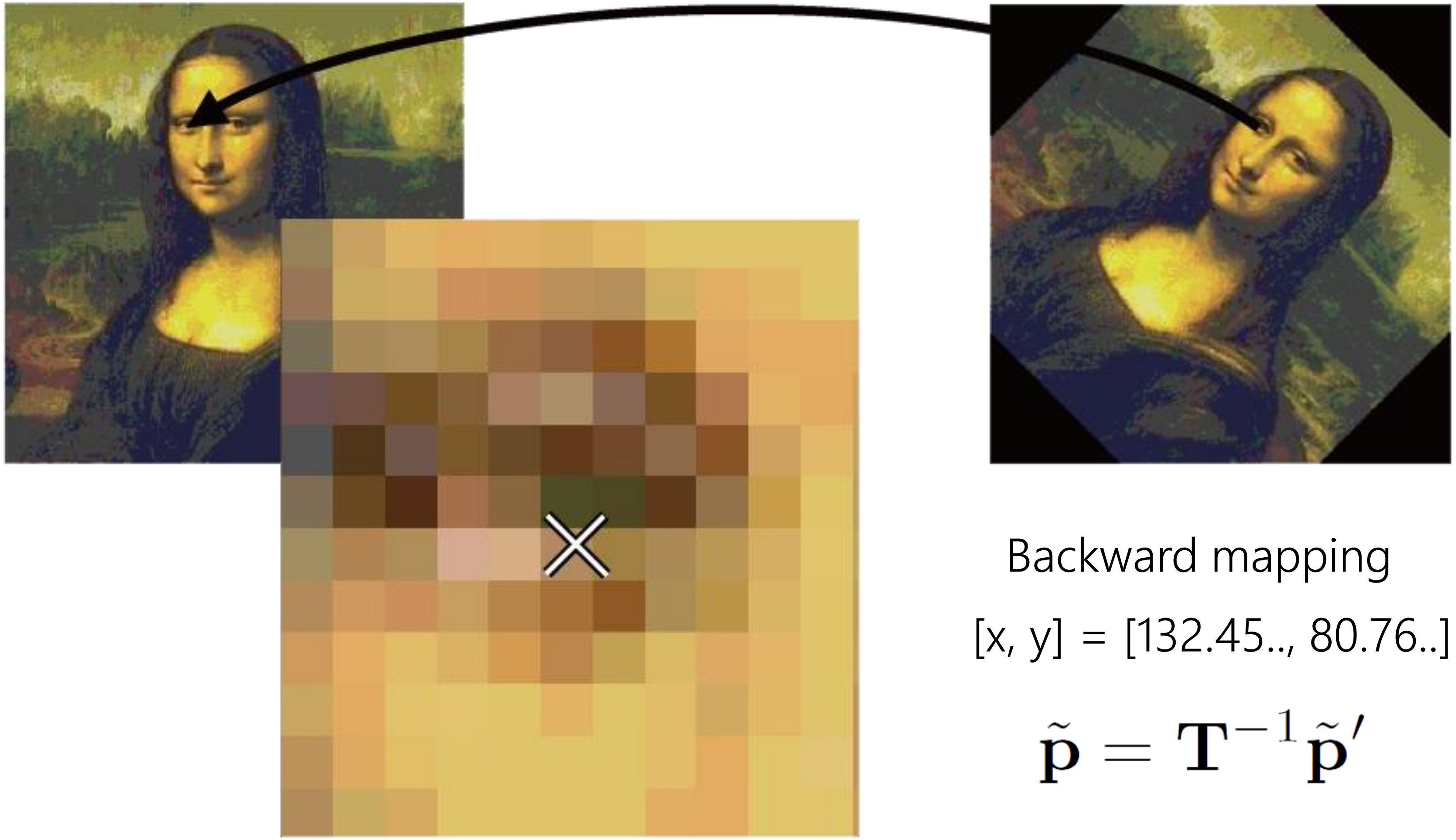
$$\tilde{\mathbf{p}} = \mathbf{T}^{-1}\tilde{\mathbf{p}}'$$

(if we round coordinates, image will contain gaps!)

[CM20219 Notes, Section 4.5]

Image warping

- How do we warp images given a known affine transform?



Backward mapping

[x, y] = [132.45.., 80.76..]

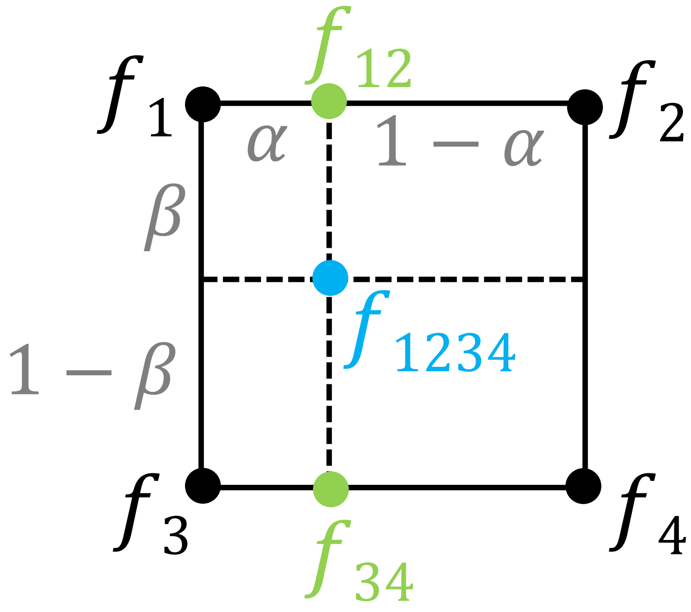
$$\tilde{\mathbf{p}} = \mathbf{T}^{-1}\tilde{\mathbf{p}}'$$

Nearest-neighbour interpolation

- Just round to nearest integer
- e.g. [x, y] = [132.45.., 80.76..] → [132, 81]

Bilinear interpolation

- Bilinear interpolation between samples

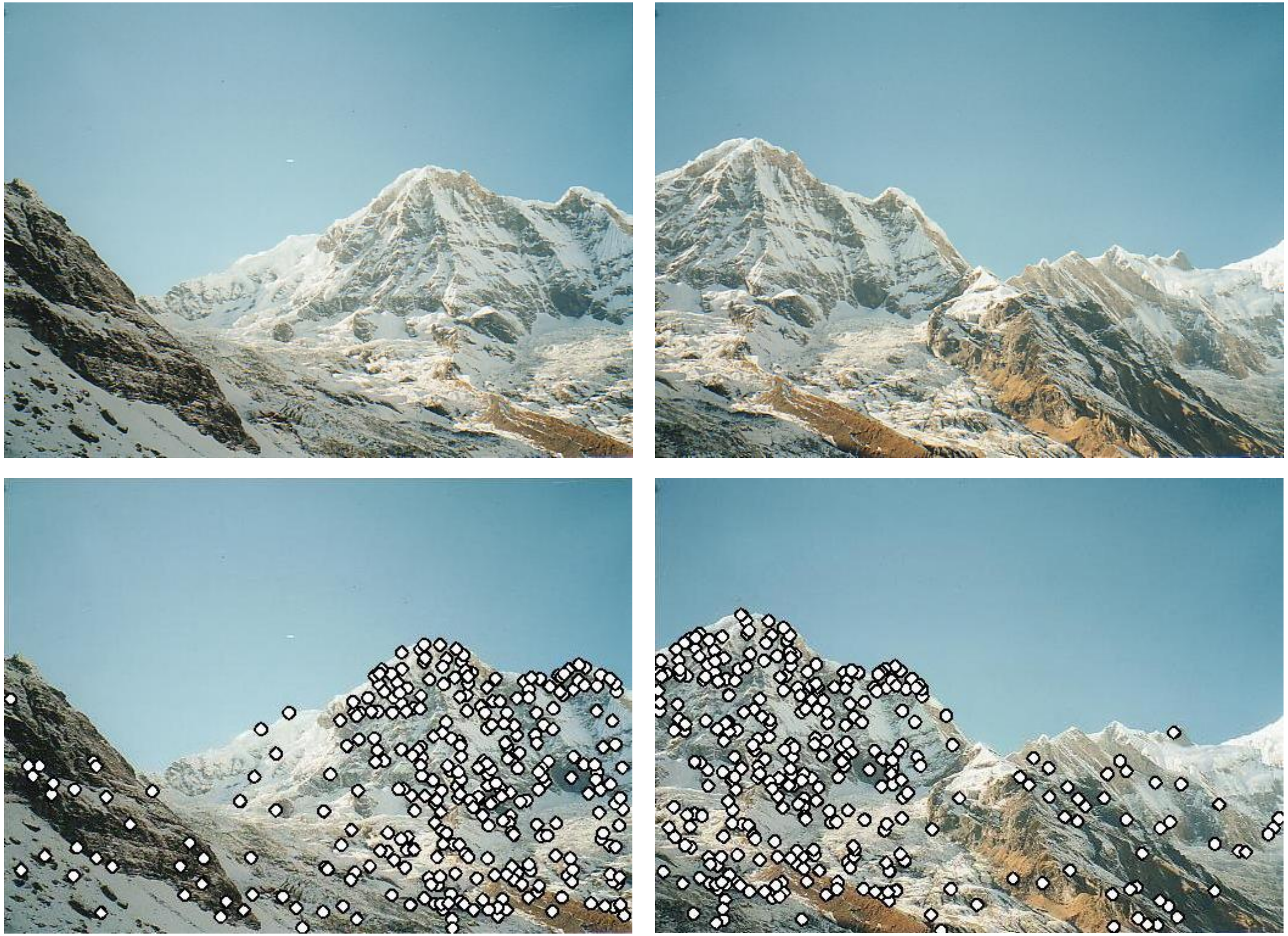


$$f_{12} = (1-\alpha) f_1 + \alpha f_2$$

$$f_{34} = (1-\alpha) f_3 + \alpha f_4$$

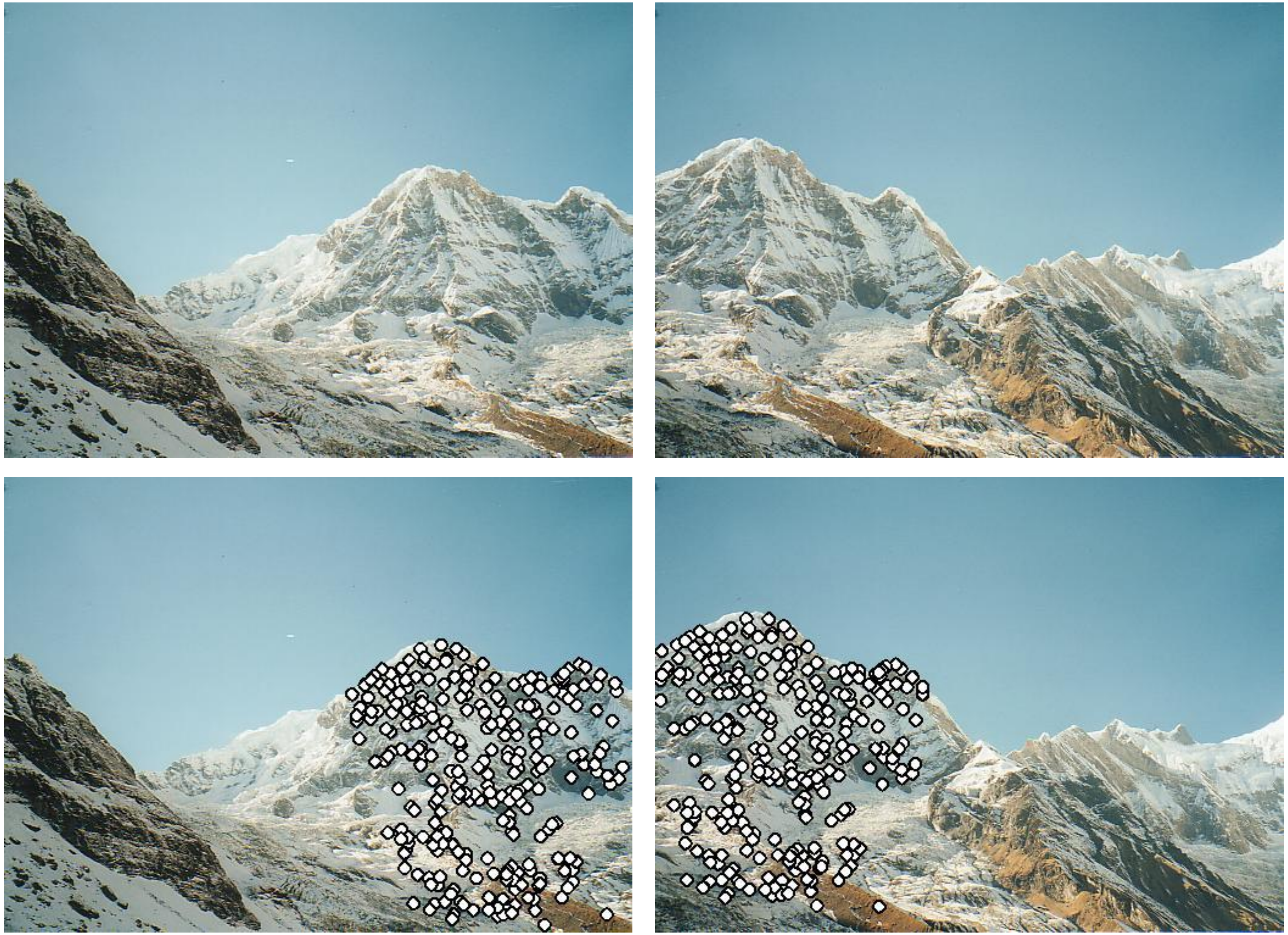
$$f_{1234} = (1-\beta) f_{12} + \beta f_{34}$$

RANSAC for Homography



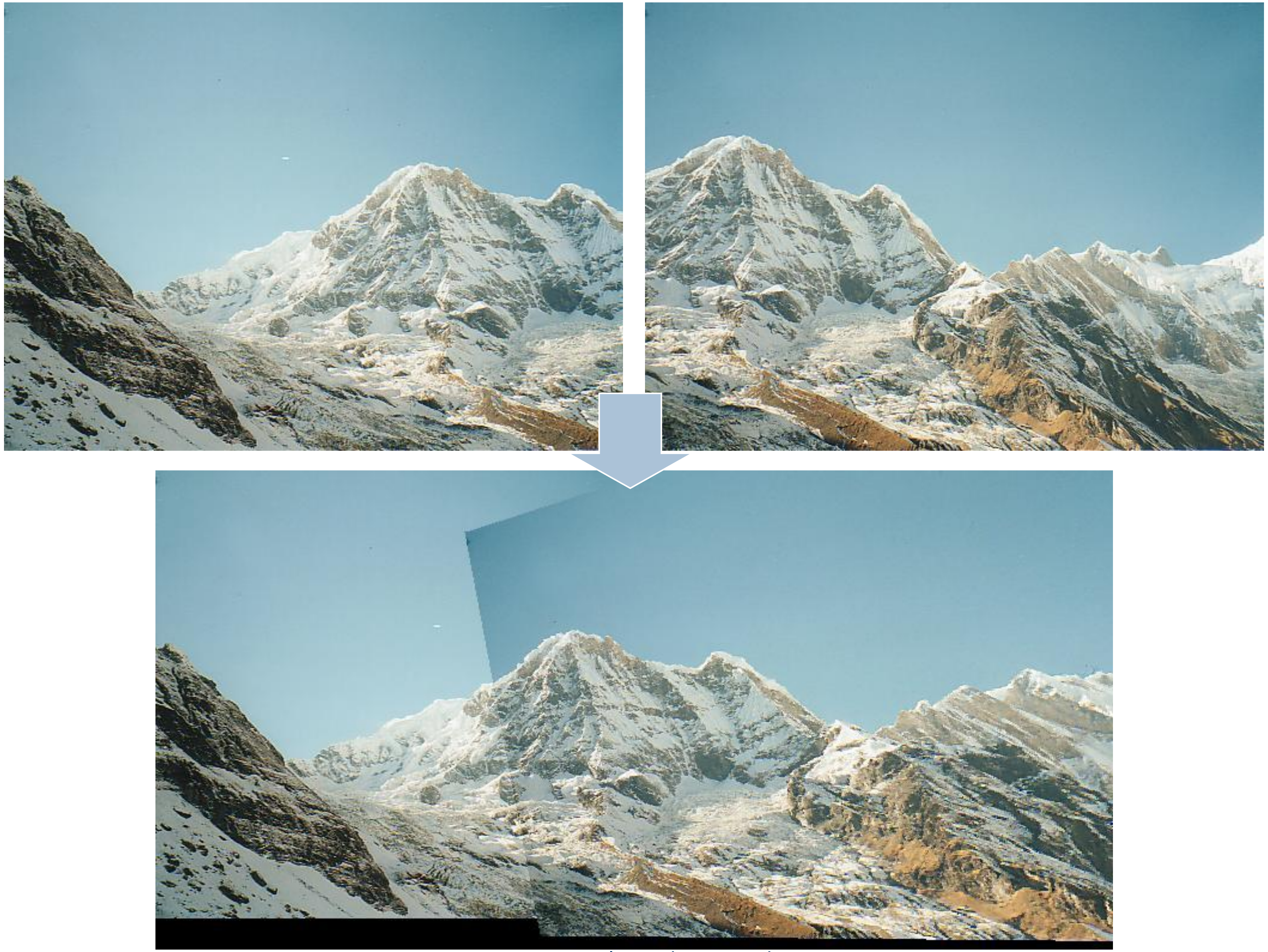
Slide by Matthew Brown and David Lowe

RANSAC for Homography



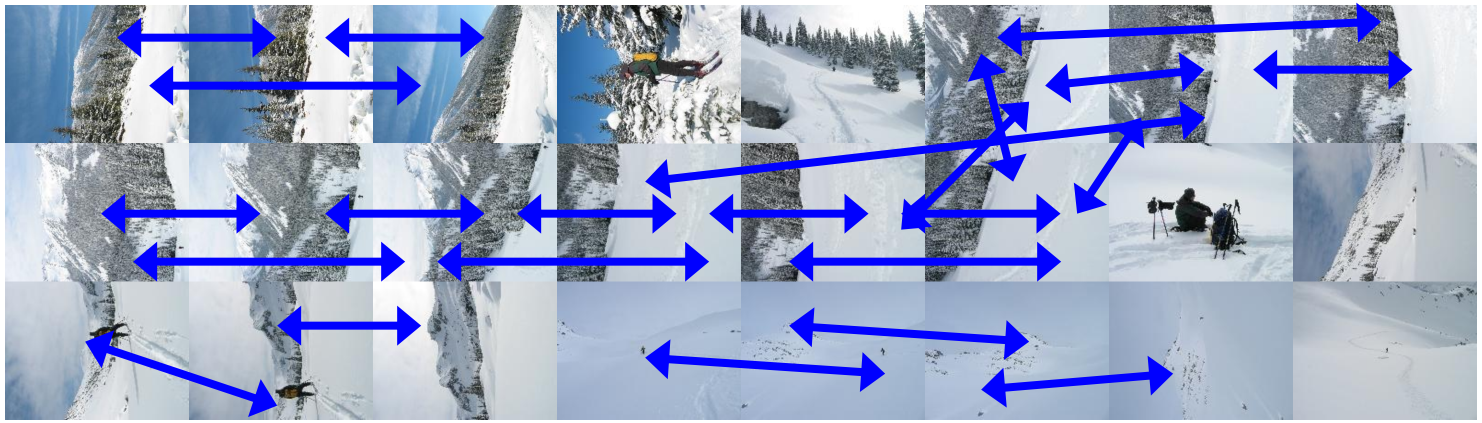
Slide by Matthew Brown and David Lowe

RANSAC for Homography



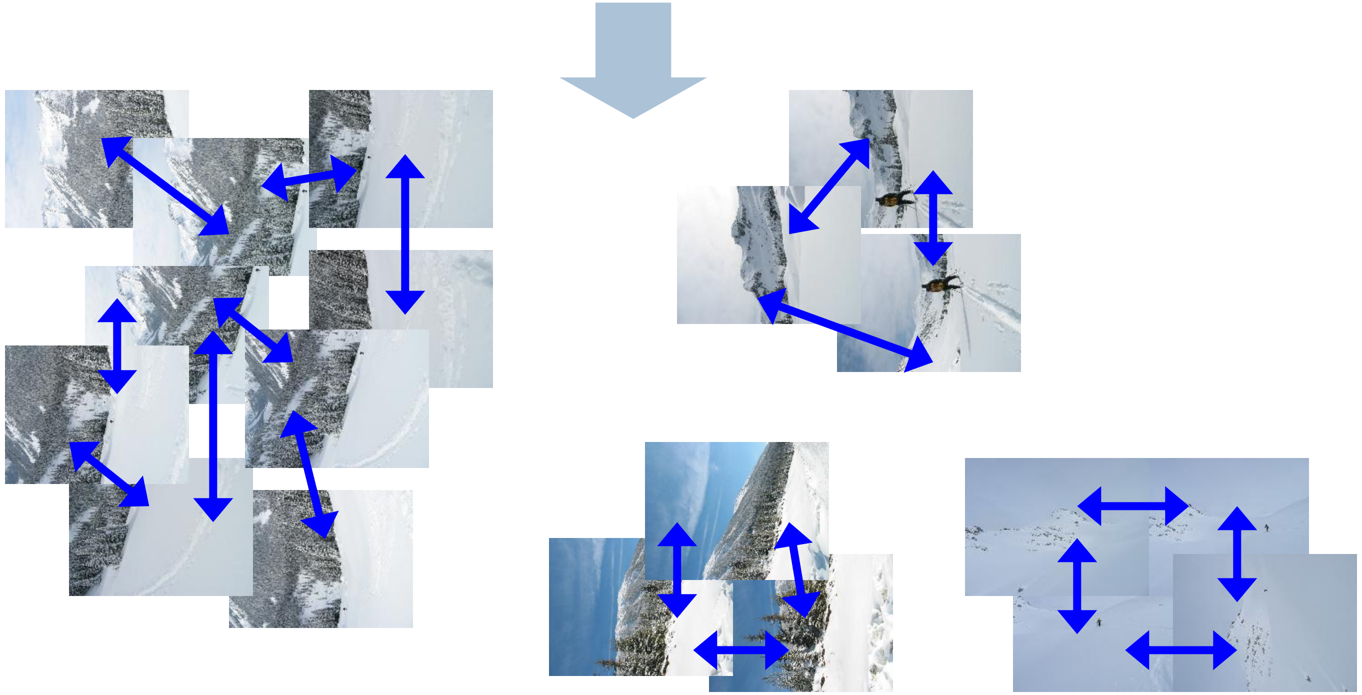
Slide by Matthew Brown and David Lowe

Finding the panoramas



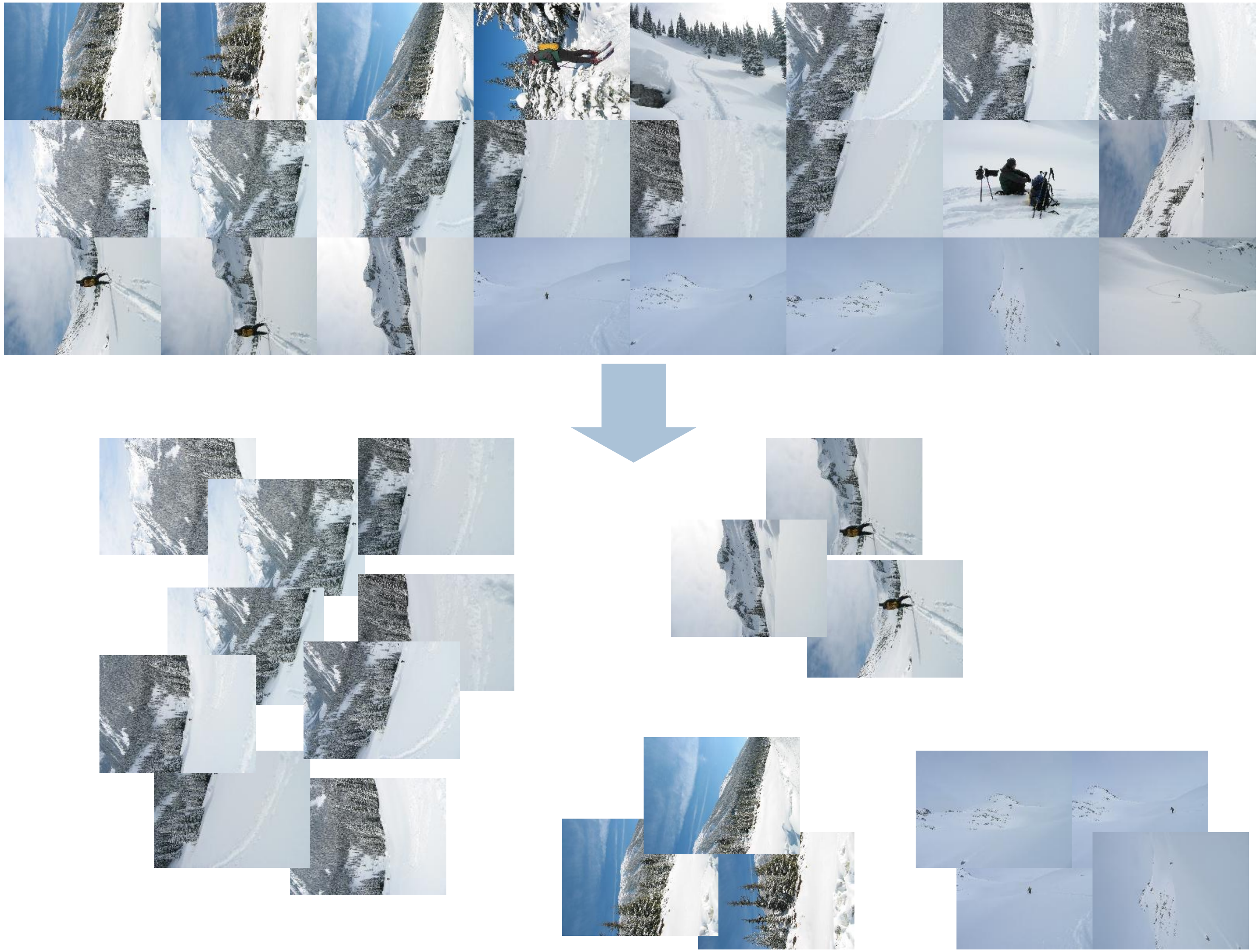
Slide by Matthew Brown and David Lowe

Finding the panoramas



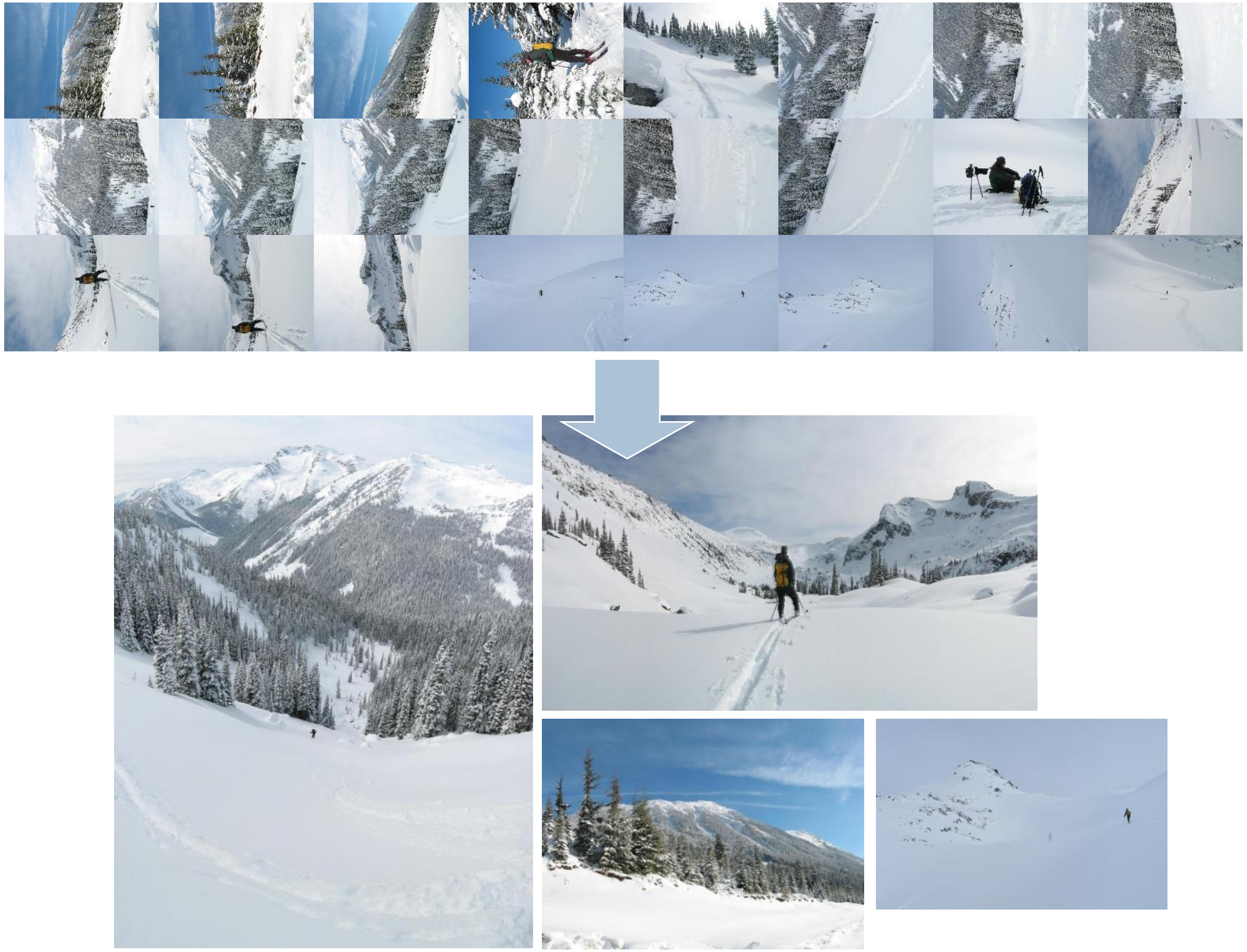
Slide by Matthew Brown and David Lowe

Finding the panoramas



Slide by Matthew Brown and David Lowe

Finding the panoramas



Bundle adjustment

New images initialised with rotation,
focal length of best matching image

Slide by Matthew Brown and David Lowe

Bundle adjustment

New images initialised with rotation,
focal length of best matching image

Slide by Matthew Brown and David Lowe

Panorama stitching

- Multi-band blending [Burt & Adelson 1983]
  - Blend frequency bands over different scales

Slide by Matthew Brown and David Lowe

Conclusion + Questions