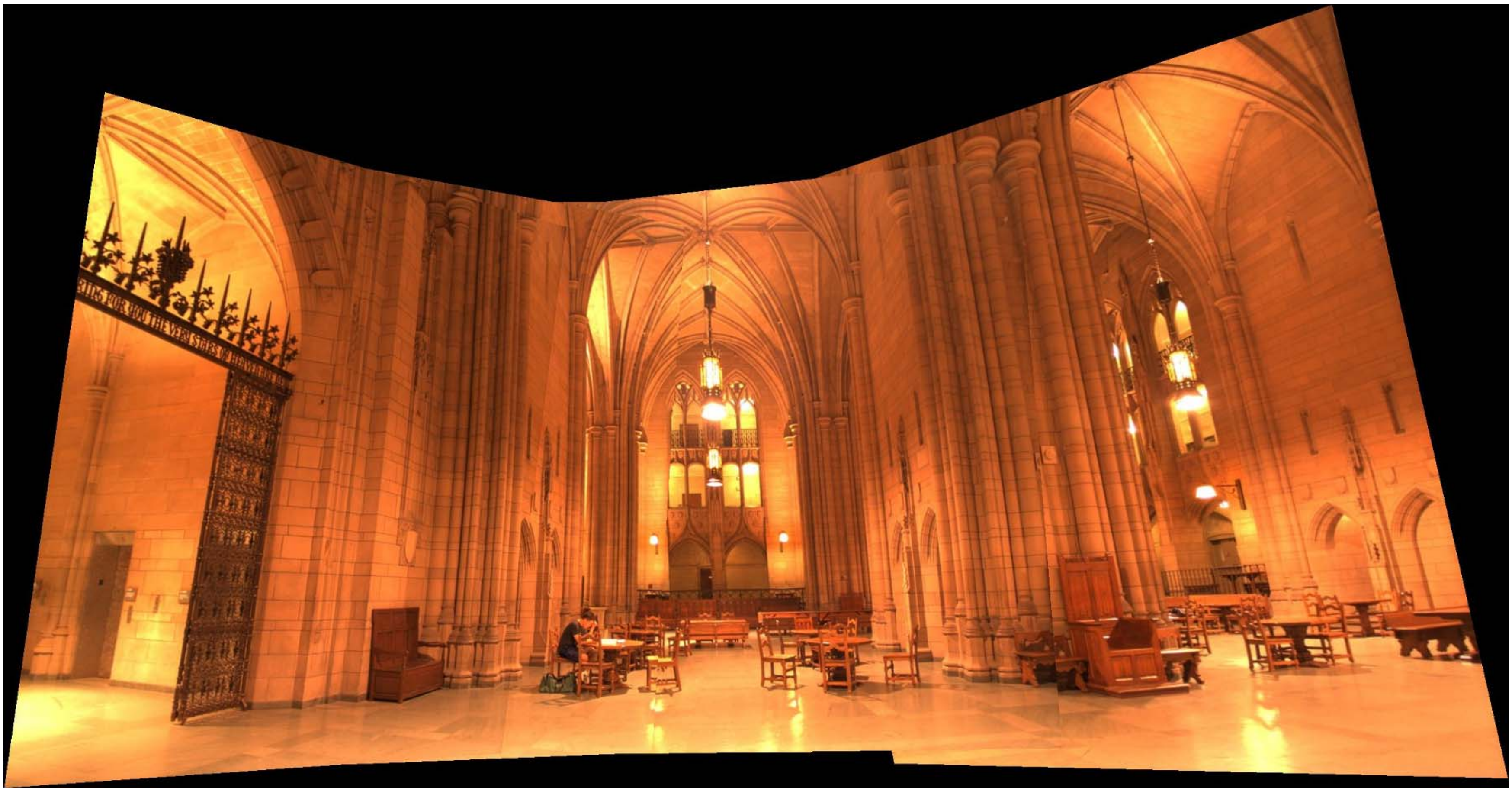
Feature Matching and RANSAC

15-463: Computational Photography
Alexei Efros, CMU, Fall 2005

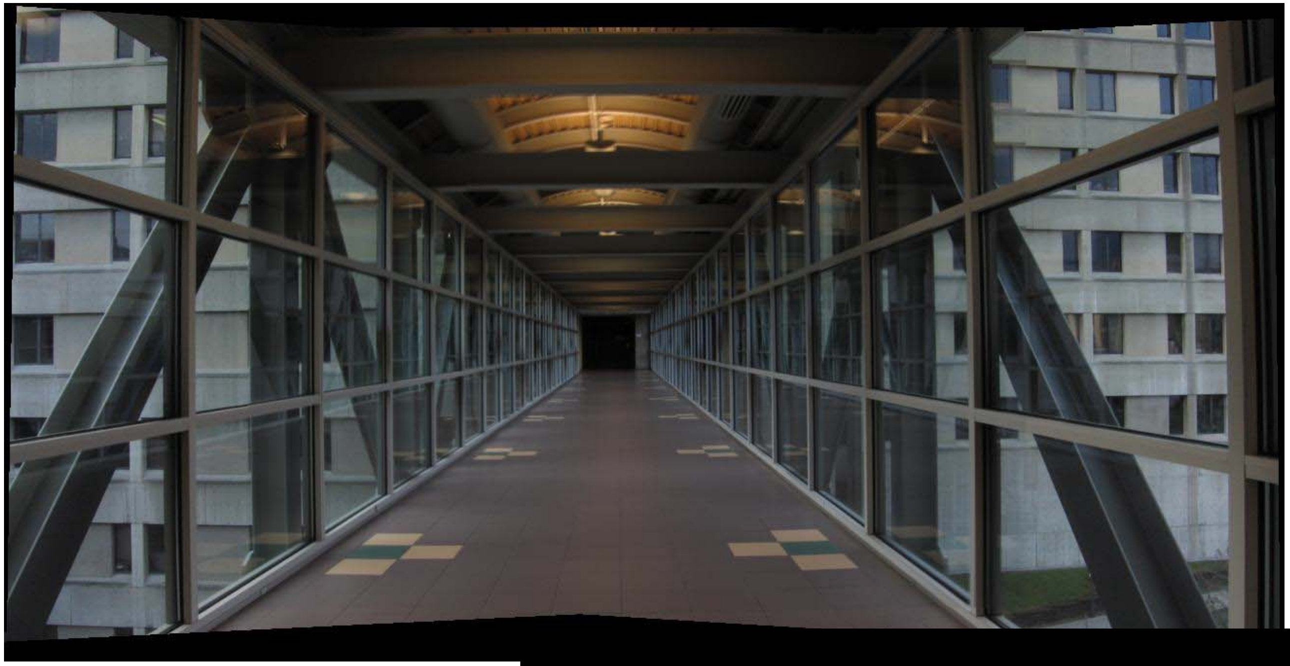
with a lot of slides stolen from Steve Seitz and Rick Szeliski

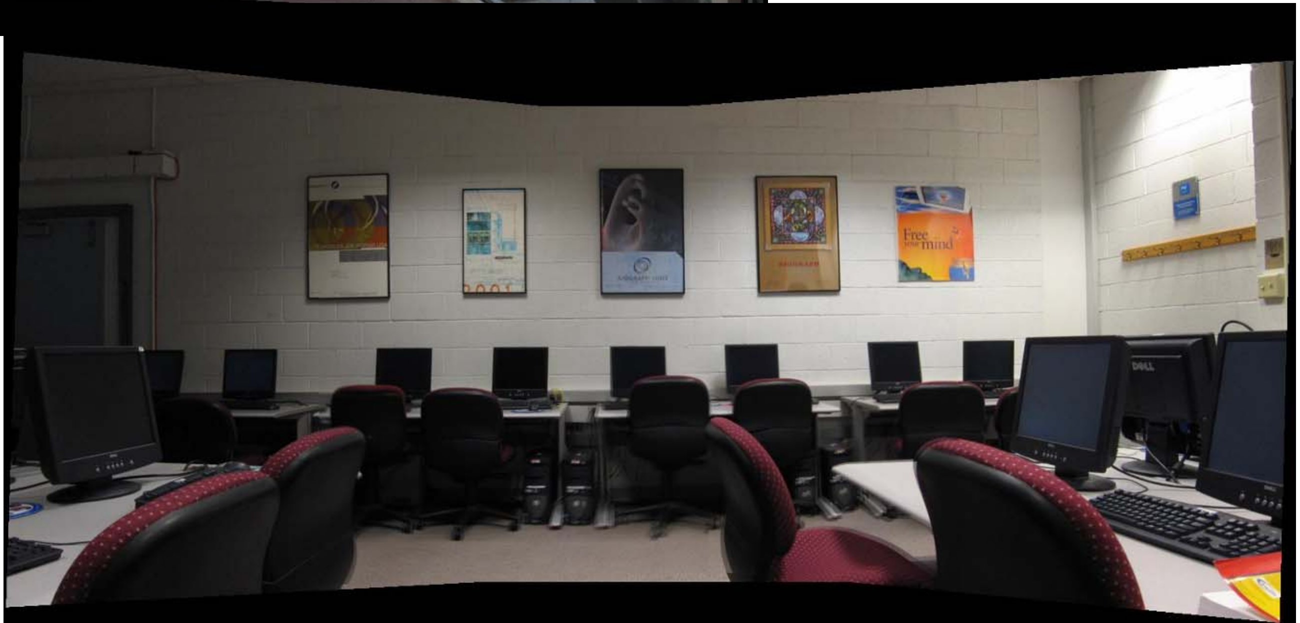
Project #3 Showcase



mrom

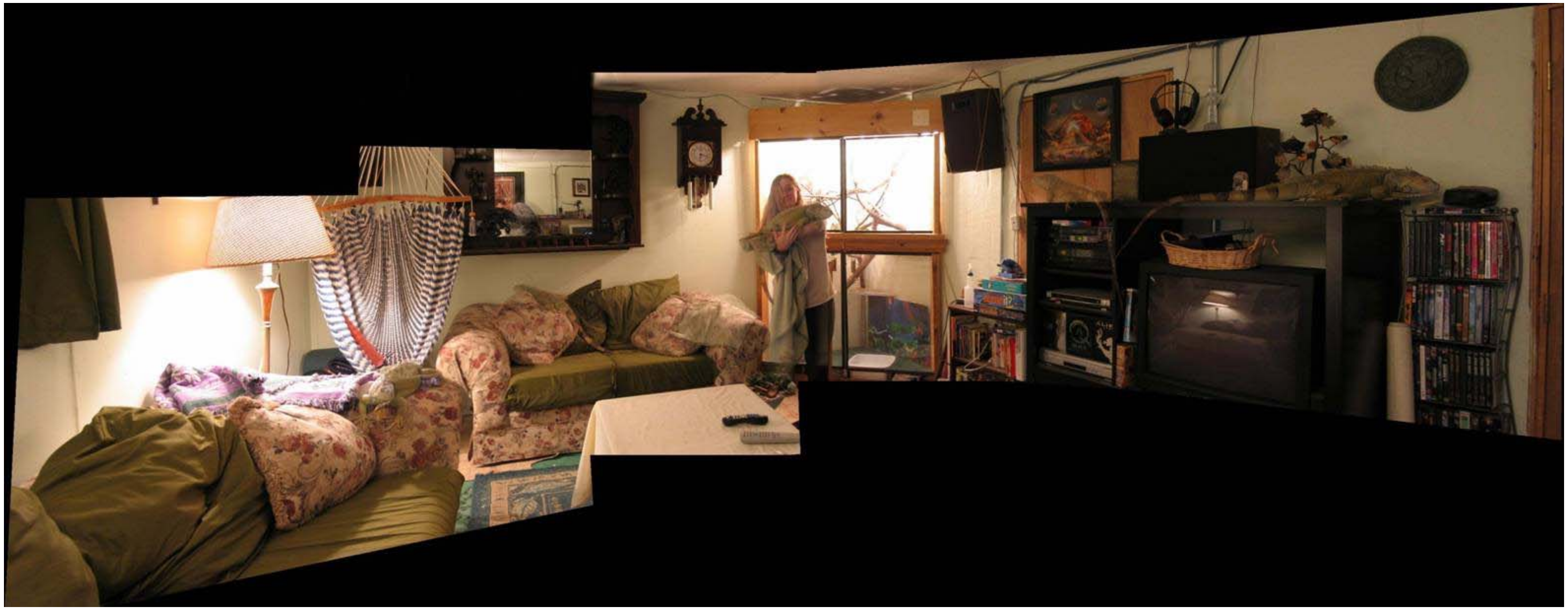
Project #3 Showcase





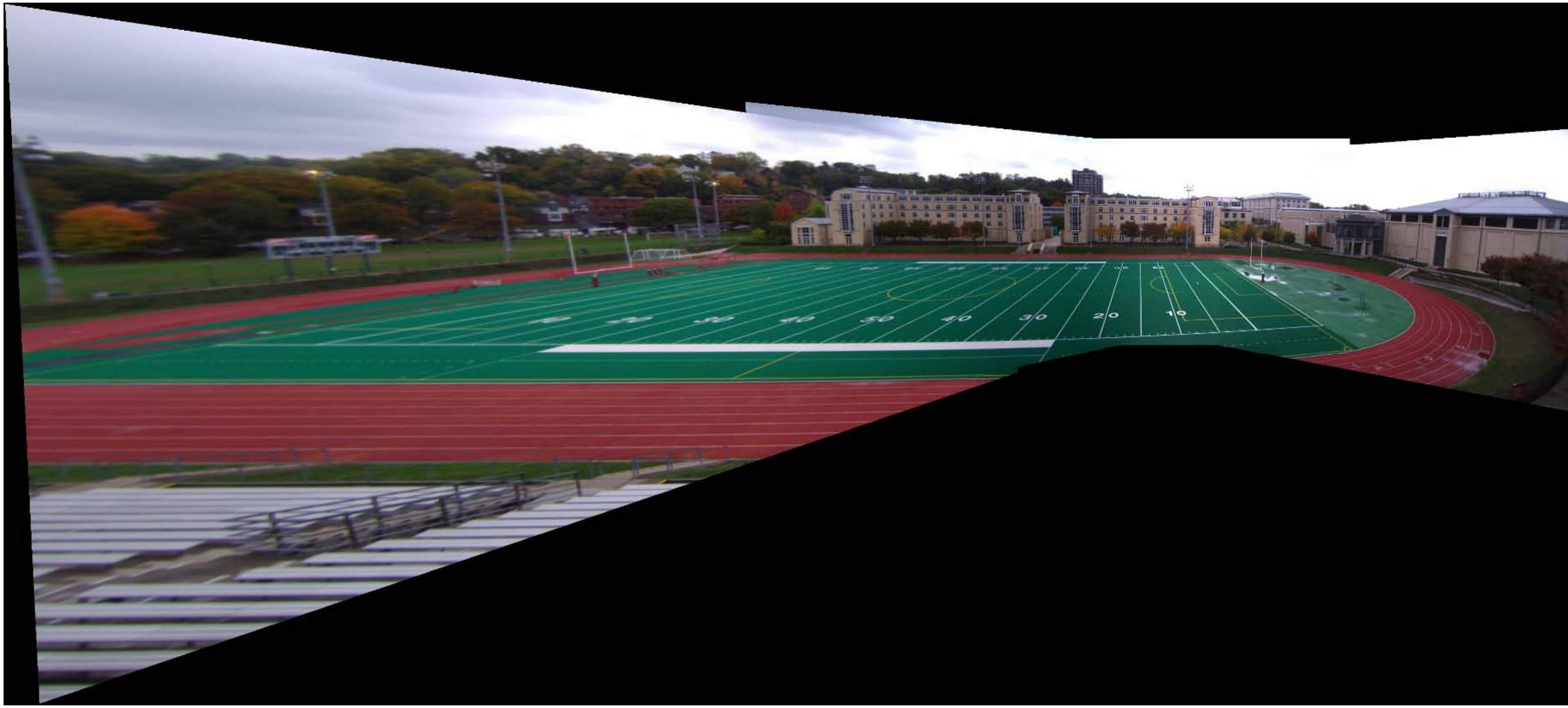
heegunl

Project #3 Showcase



jmacalli

Project #3 Showcase



lms

Project #3 Showcase

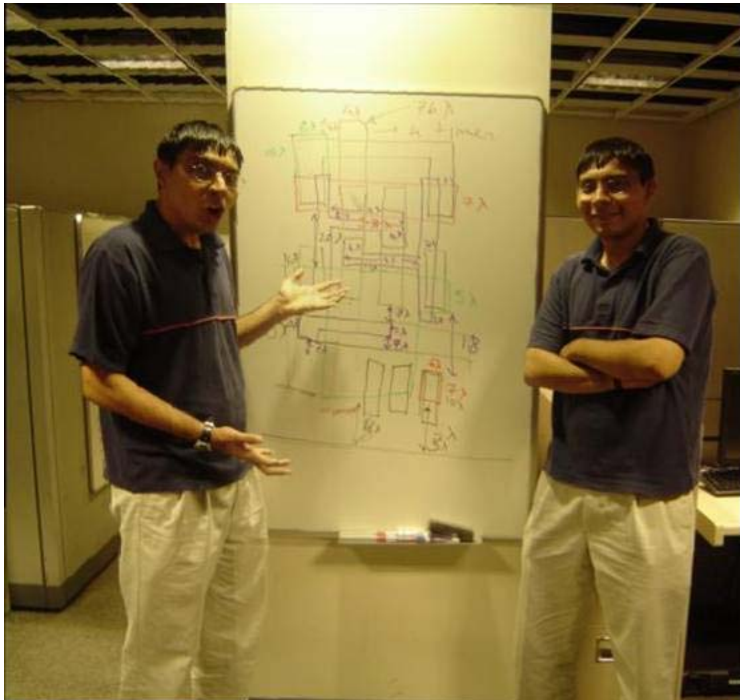
cmcamero

Project #3 Showcase

Project #3 Showcase

cmcamero

Project #3 Showcase

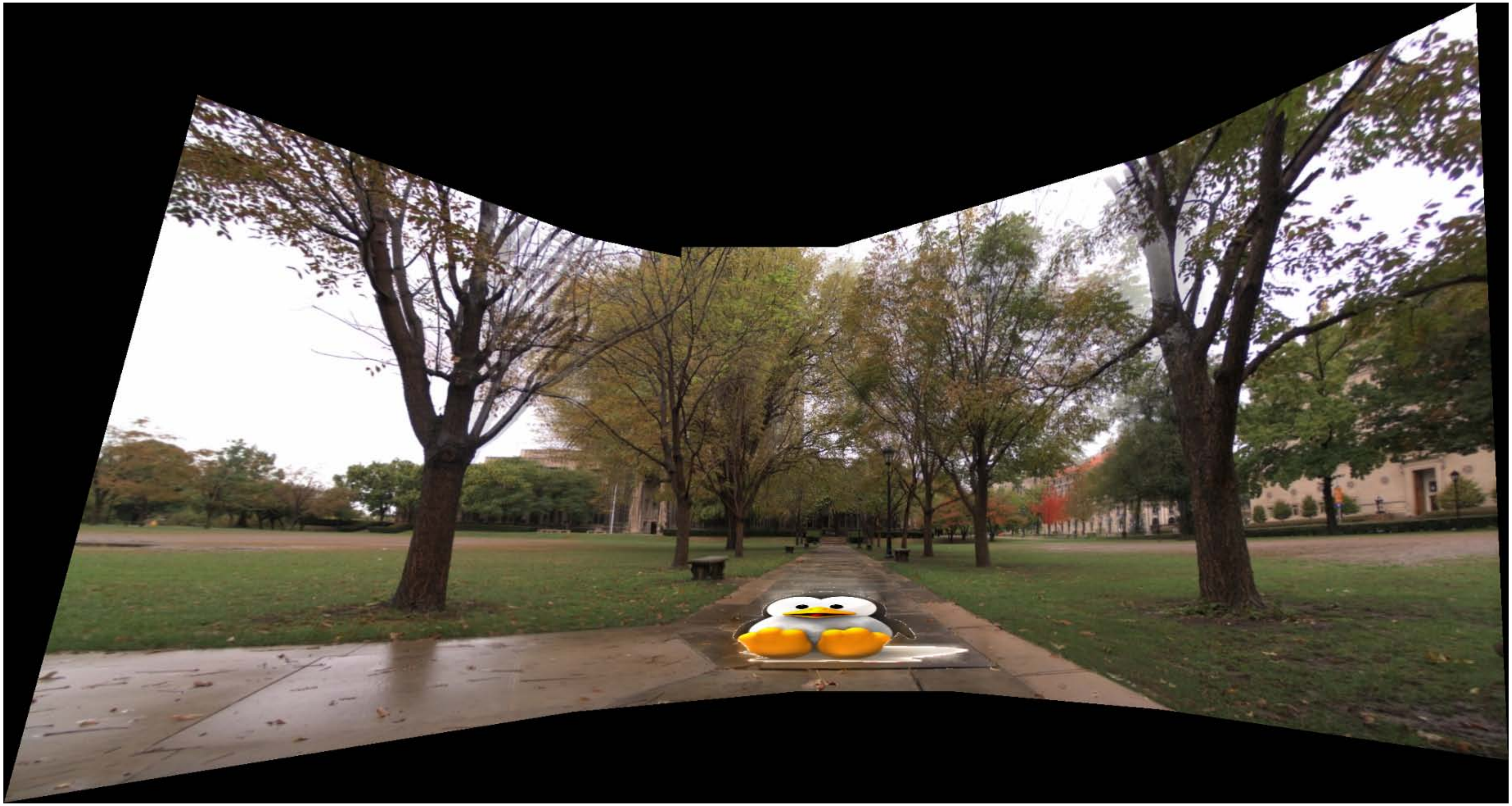


slim

nvyas

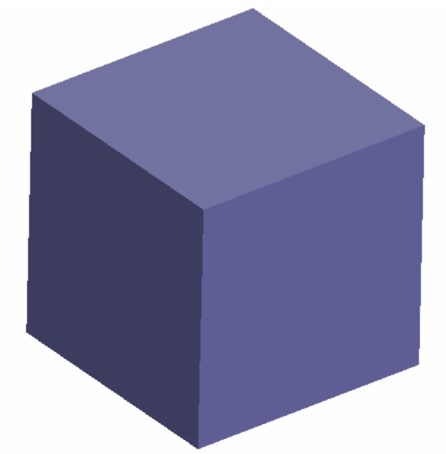
Project #3 Showcase

Project #3 Showcase



mrom

Project #3 Showcase

heegunl

Project #3 Showcase

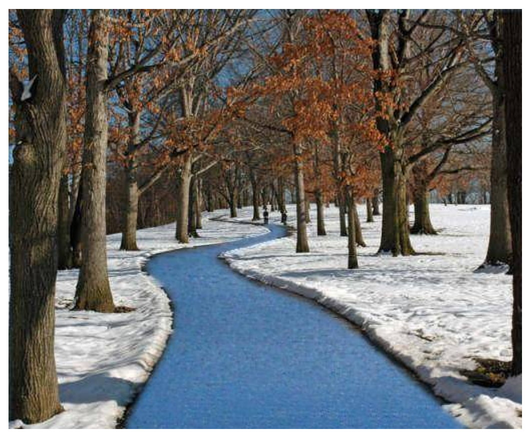
cmcamero

bhon

Project #3 Showcase








chenyuwu

Project #3 Showcase

slim

Multi-Scale Oriented Patches

Interest points

- Multi-scale Harris corners
- Orientation from blurred gradient
- Geometrically invariant to rotation

Descriptor vector

- Bias/gain normalized sampling of local patch (8x8)
- Photometrically invariant to affine changes in intensity

[Brown, Szeliski, Winder, CVPR’2005]

Descriptor Vector

Orientation = blurred gradient

Rotation Invariant Frame

- Scale-space position (x, y, s) + orientation (θ)

Detections at multiple scales

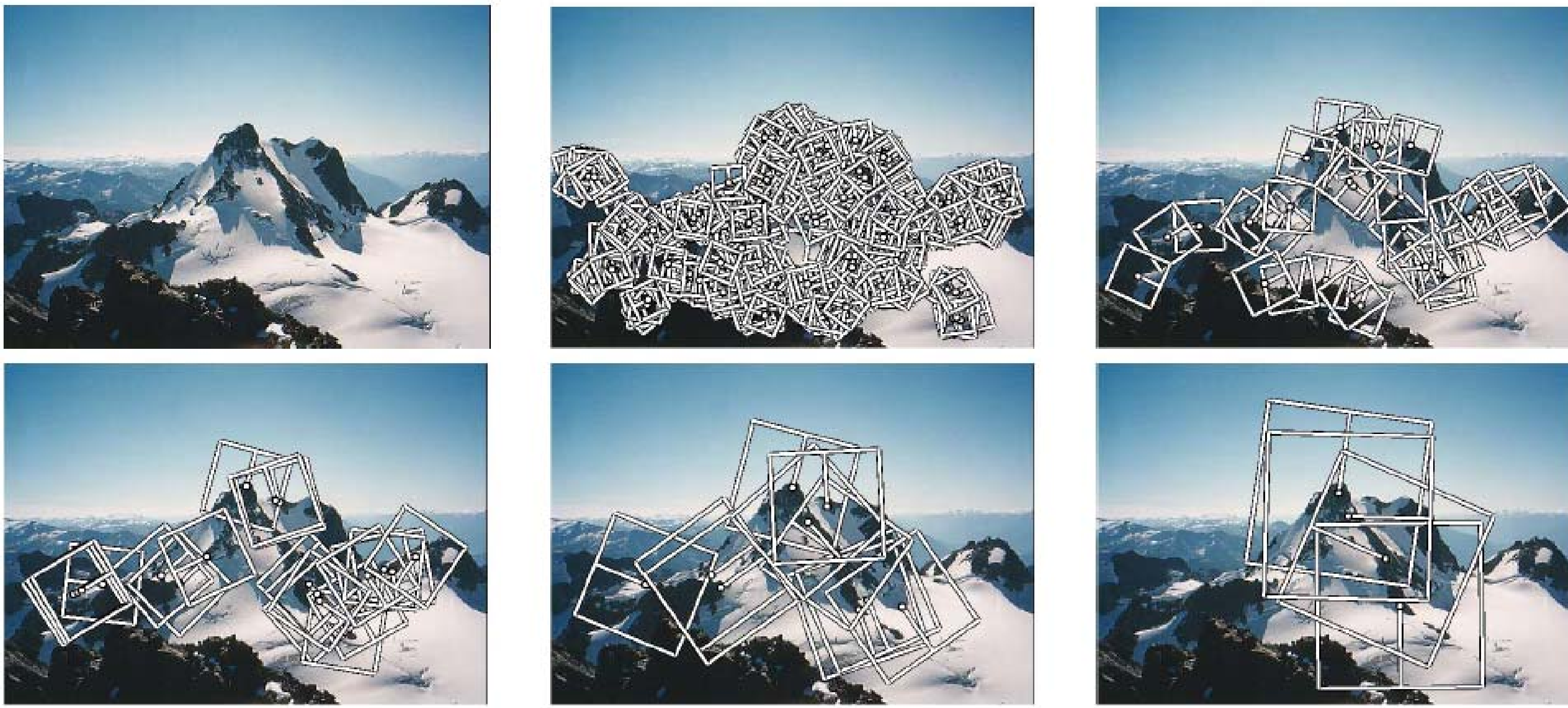


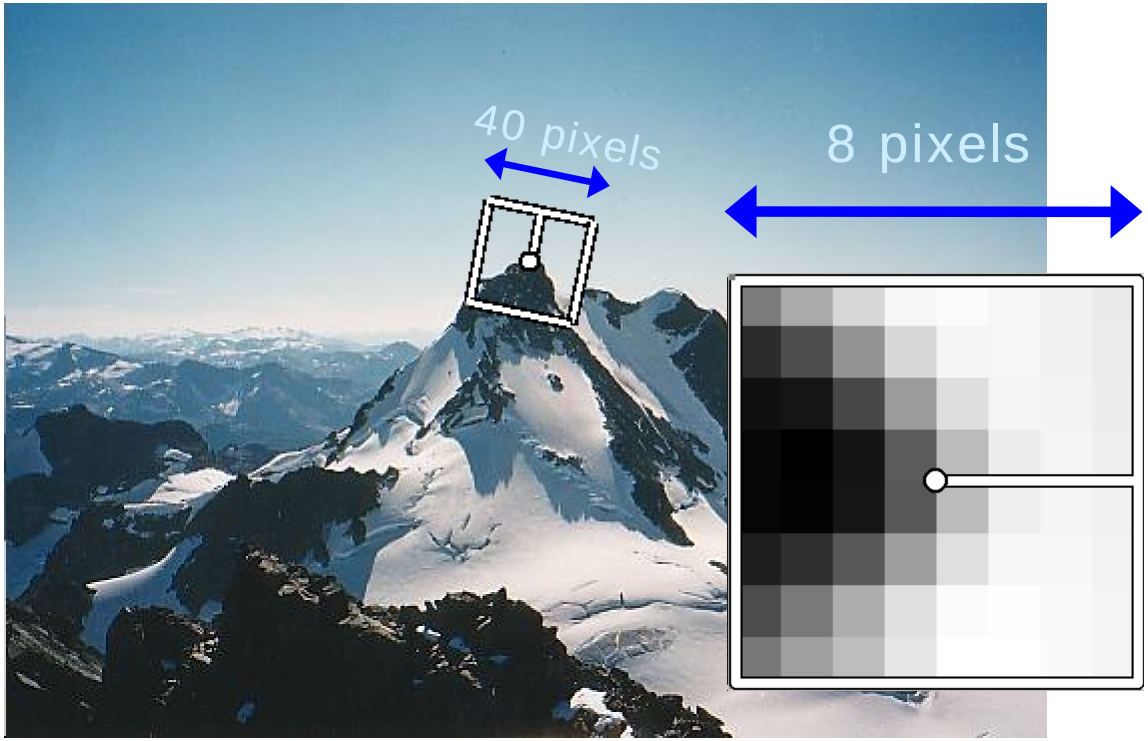
Figure 1. Multi-scale Oriented Patches (MOPS) extracted at five pyramid levels from one of the Matier images. The boxes show the feature orientation and the region from which the descriptor vector is sampled.

MOPS descriptor vector

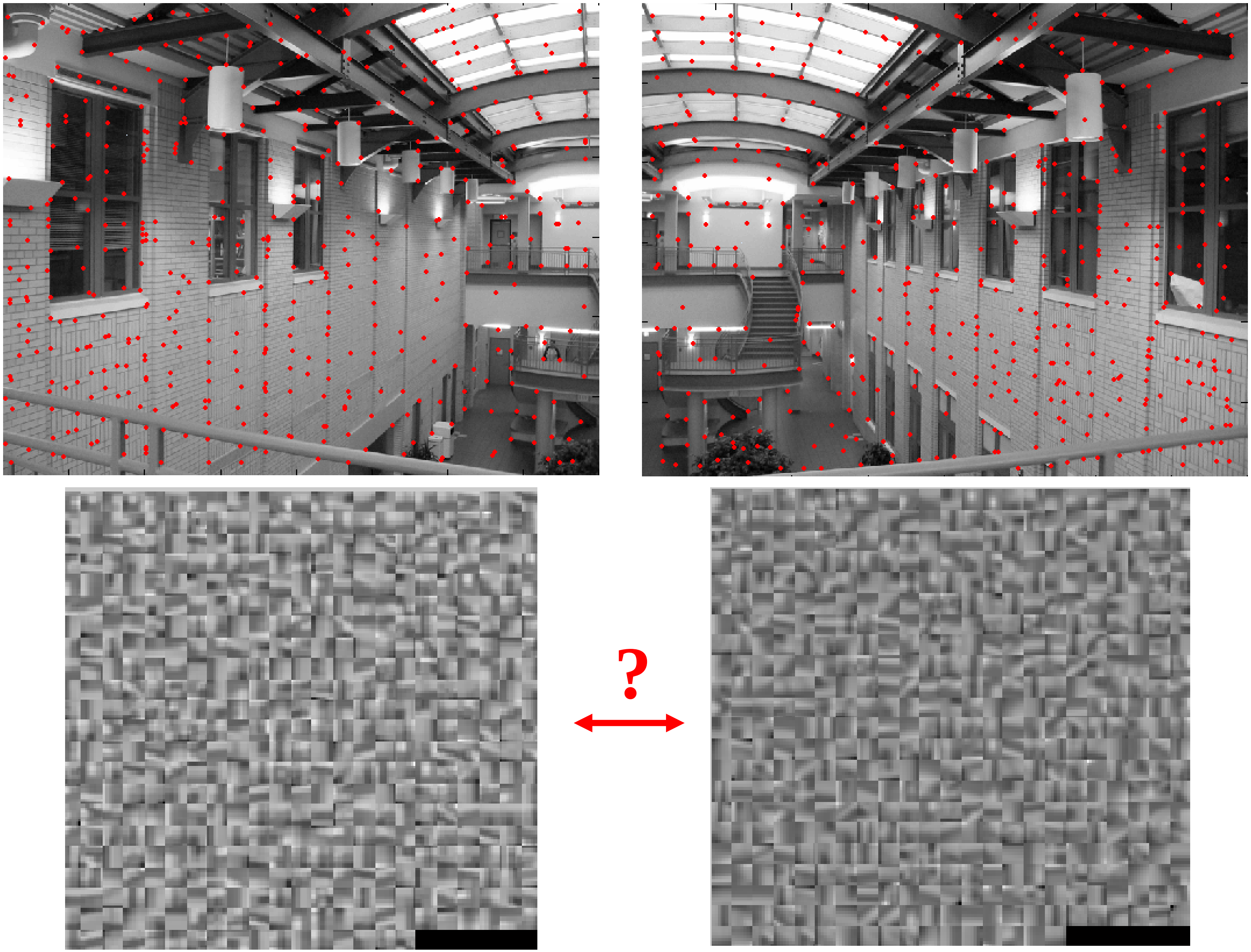
8x8 oriented patch

- Sampled at 5 x scale

Bias/gain normalisation: $I' = (I - \mu)/\sigma$



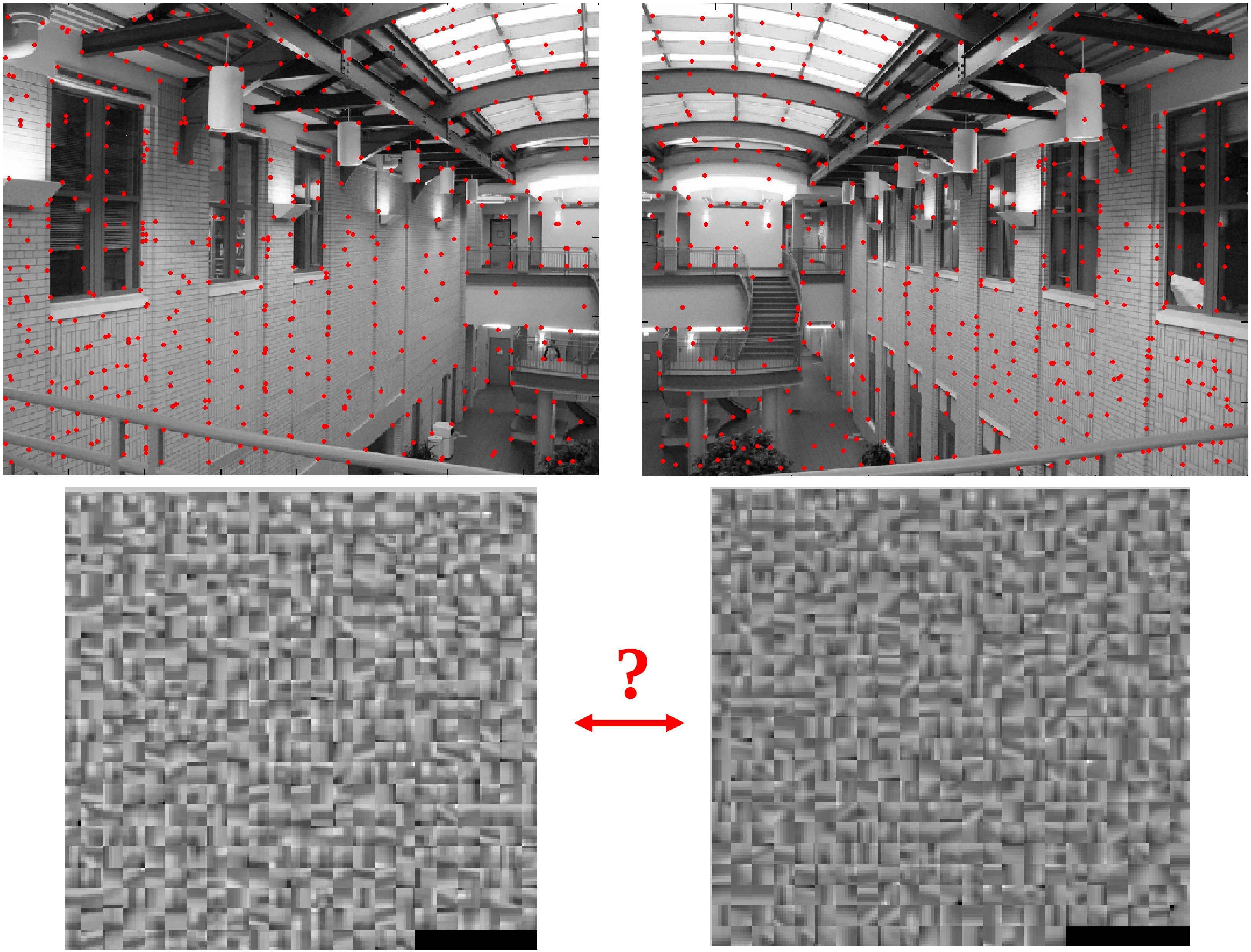
Feature matching



Feature matching

- Exhaustive search
  - for each feature in one image, look at *all* the other features in the other image(s)
- Hashing
  - compute a short descriptor from each feature vector, or hash longer descriptors (randomly)
- Nearest neighbor techniques
  - *k*-trees and their variants

What about outliers?

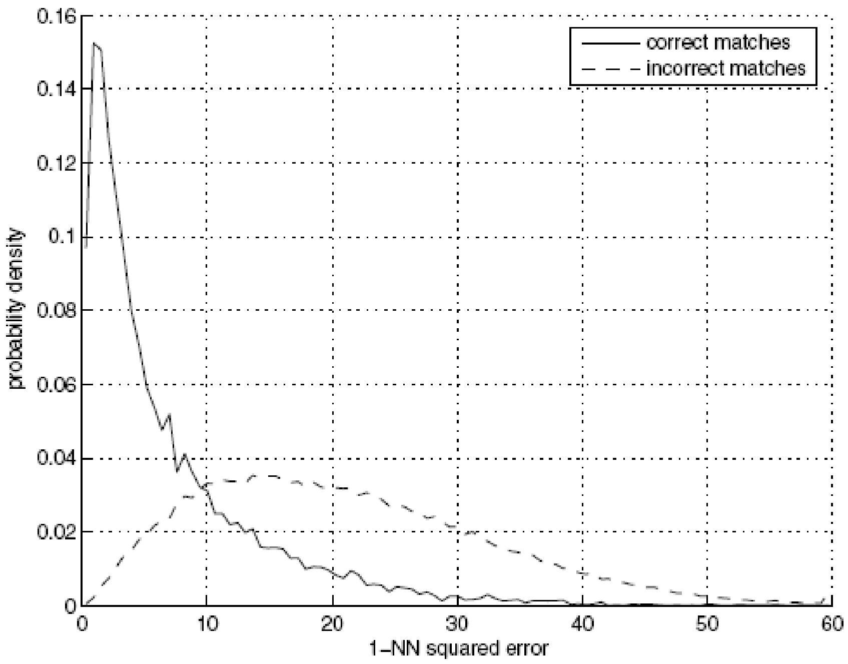


Feature-space outlier rejection

Let's not match all features, but only these that have "similar enough" matches?

How can we do it?

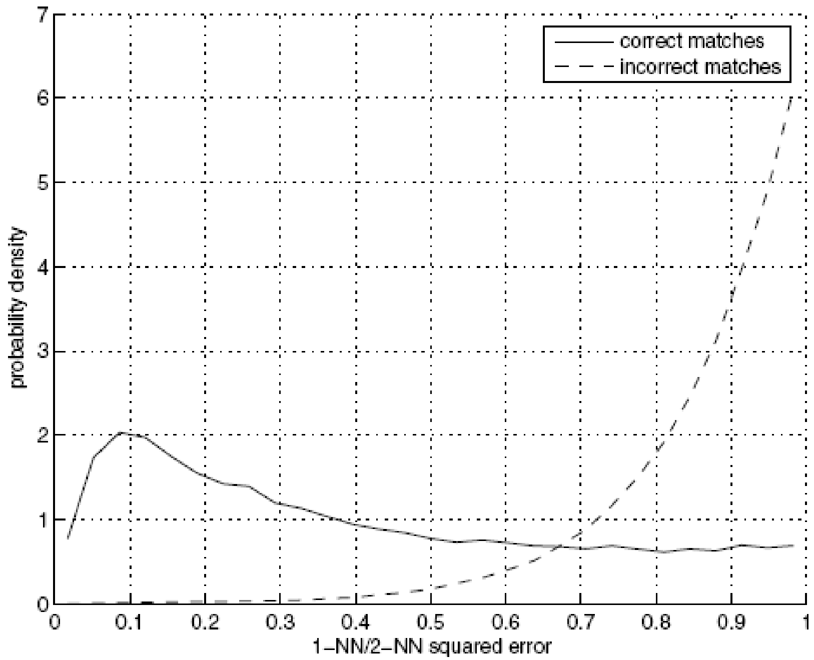
- SSD(patch1,patch2) < threshold
- How to set threshold?



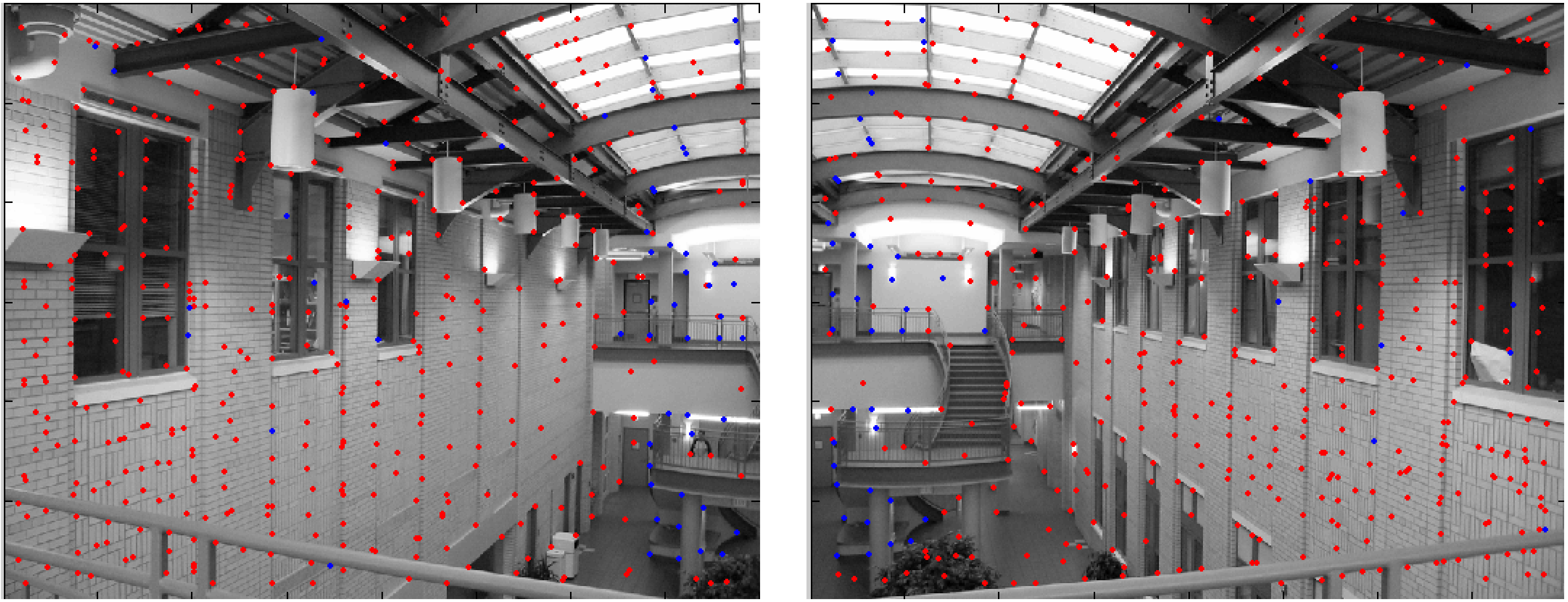
Feature-space outlier rejection

A better way [Lowe, 1999]:

- 1-NN: SSD of the closest match
- 2-NN: SSD of the <u>second-closest</u> match
- Look at how much better 1-NN is than 2-NN, e.g. 1-NN/2-NN
- That is, is our best match so much better than the rest?



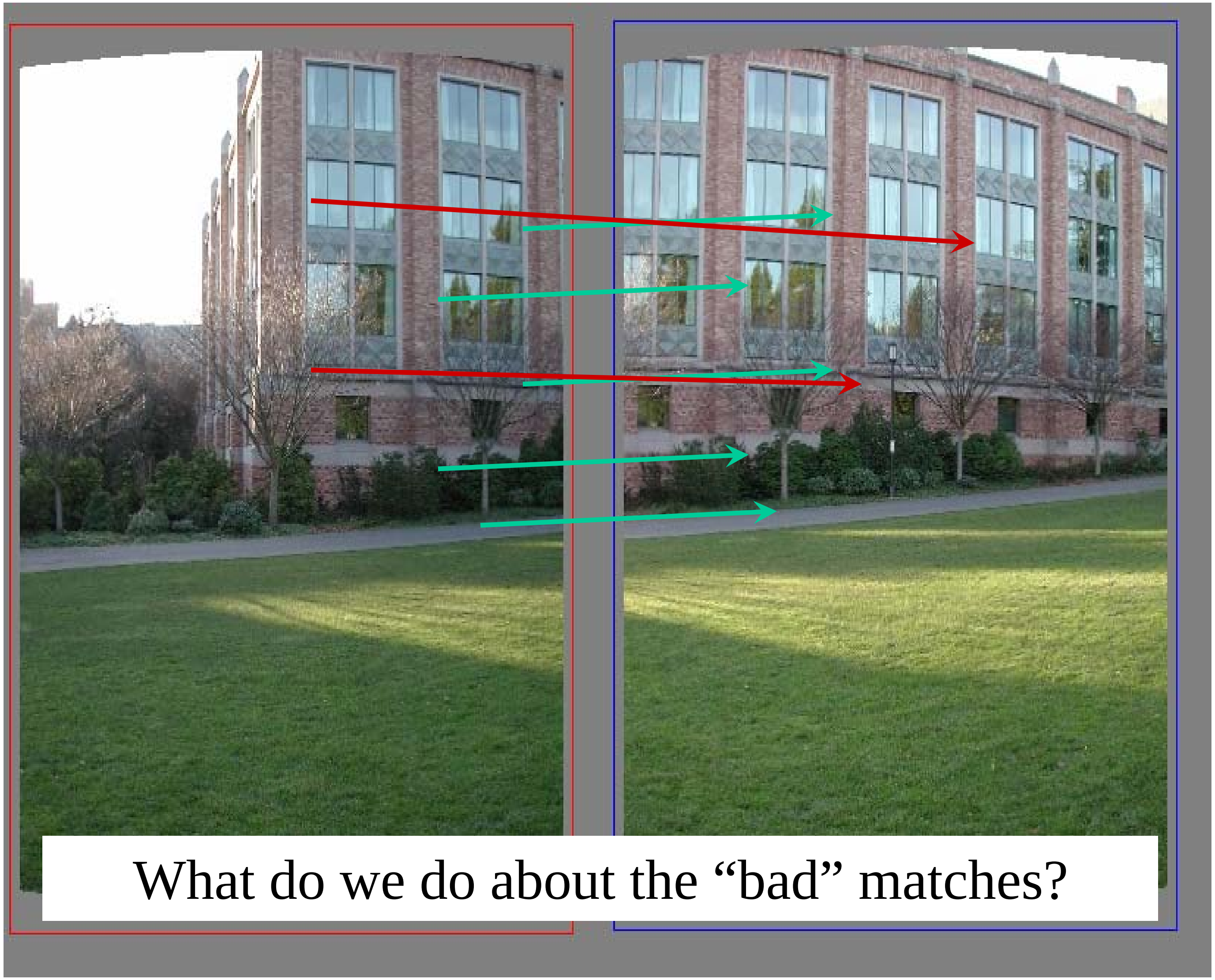
Feature-space outliner rejection



Can we now compute H from the blue points?

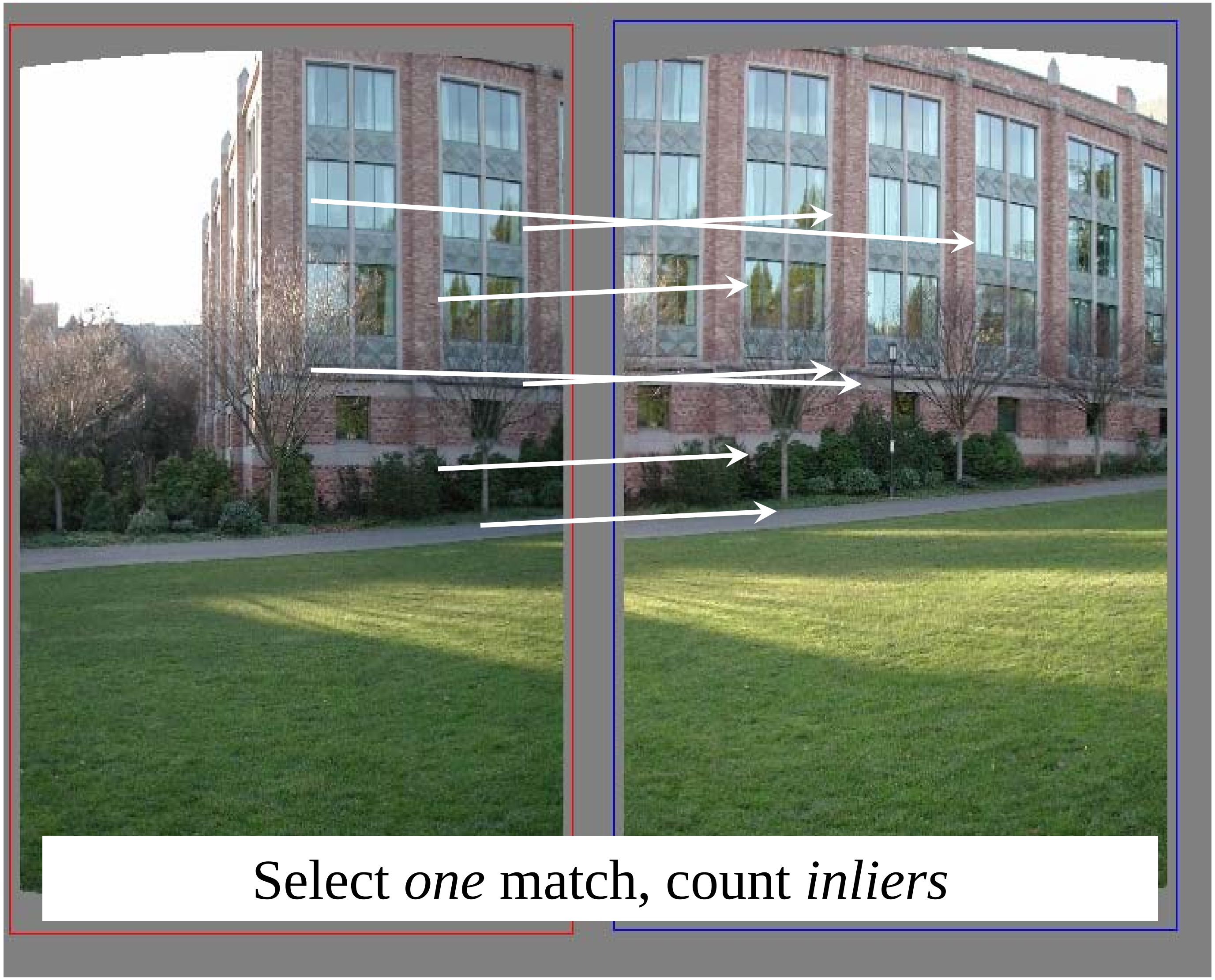
- No! Still too many outliers…
- What can we do?

Matching features



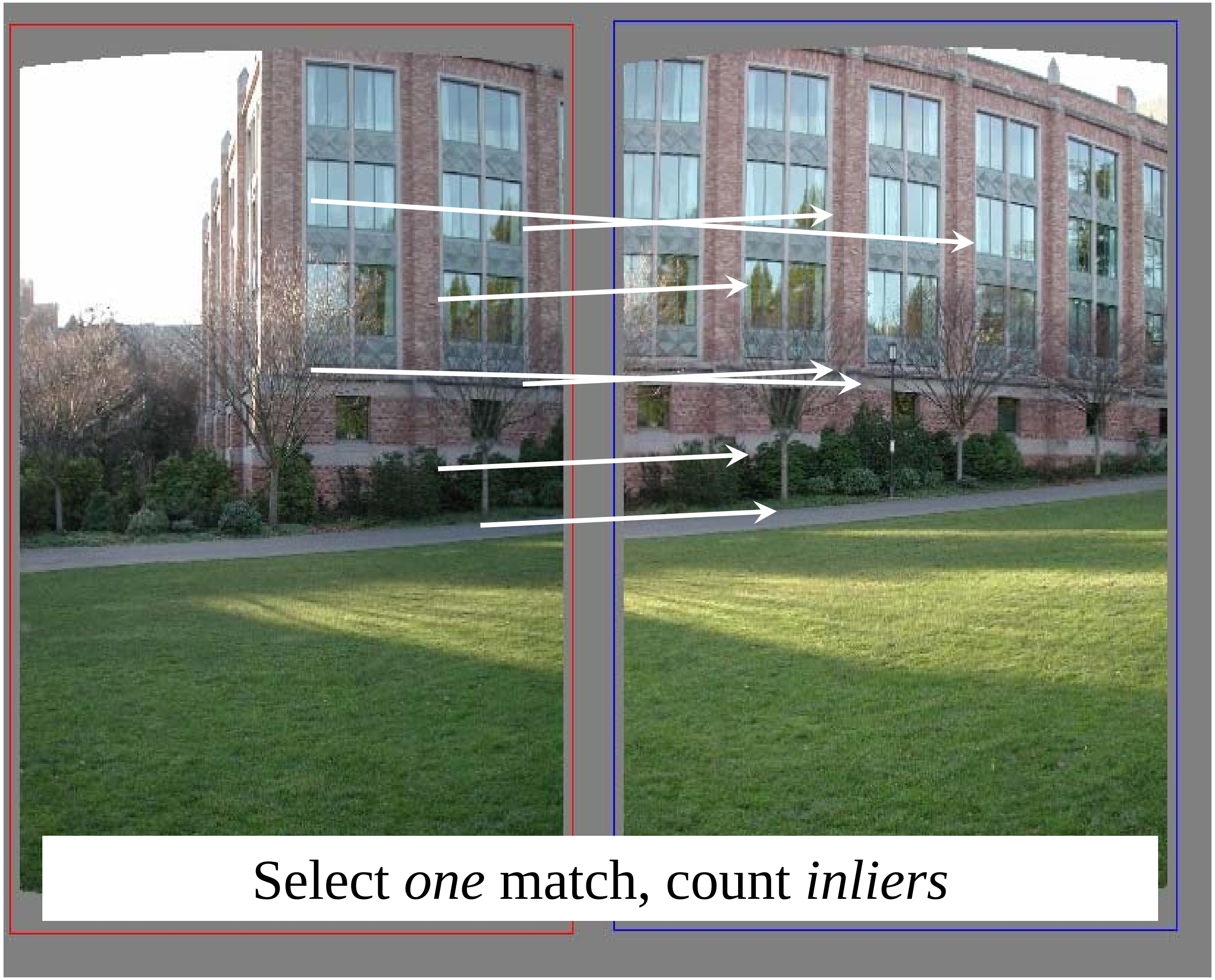
What do we do about the “bad” matches?

RAndom SAmple Consensus

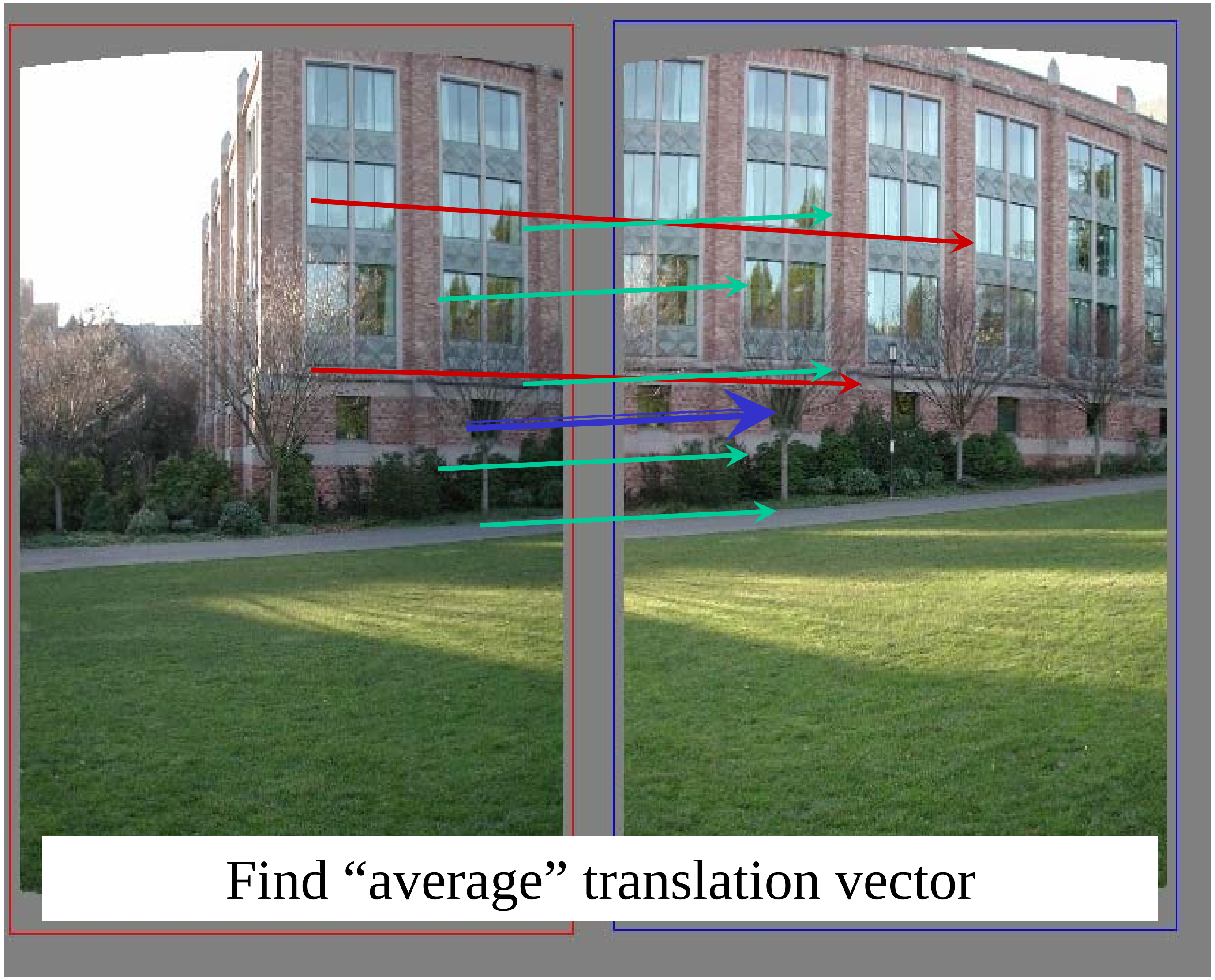


Select *one* match, count *inliers*

RAndom SAmple Consensus



Least squares fit



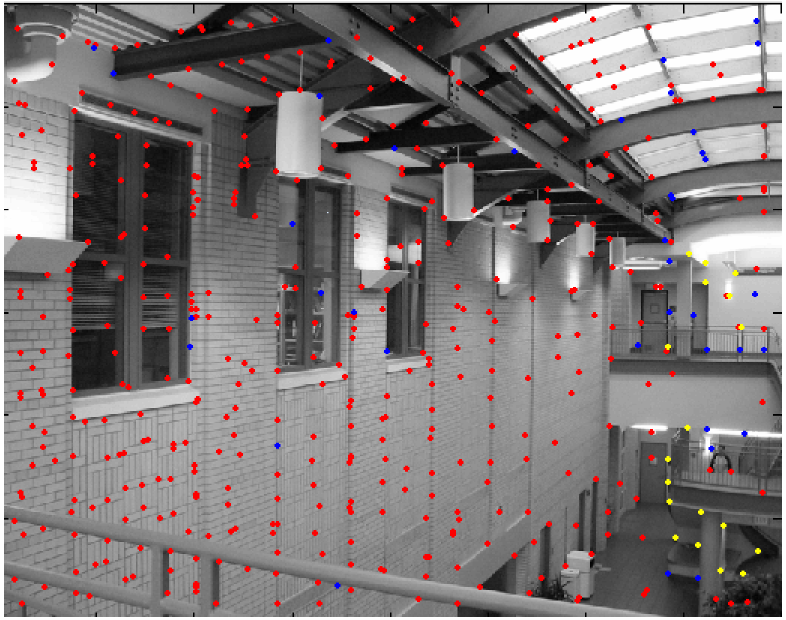
Find “average” translation vector

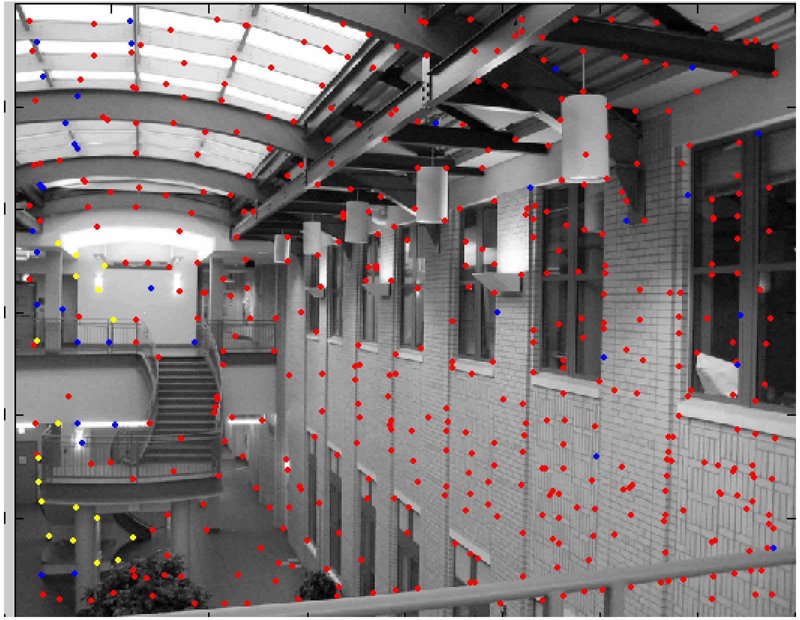
RANSAC for estimating homography

RANSAC loop:

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

RANSAC





Example: Recognising Panoramas

M. Brown and D. Lowe,
University of British Columbia

Why “Recognising Panoramas”?

Why "Recognising Panoramas"?

1D Rotations (θ)

- Ordering $\Rightarrow$ matching images

Why “Recognising Panoramas”?

1D Rotations (θ)

- Ordering $\Rightarrow$ matching images

Why “Recognising Panoramas”?

1D Rotations (θ)

- Ordering $\Rightarrow$ matching images

Why “Recognising Panoramas”?

1D Rotations (θ)

- Ordering $\Rightarrow$ matching images

- 2D Rotations (θ, ϕ)
  - Ordering $\not\Rightarrow$ matching images

Why “Recognising Panoramas”?

1D Rotations (θ)

- Ordering $\Rightarrow$ matching images

- 2D Rotations (θ, ϕ)
  - Ordering $\nRightarrow$ matching images

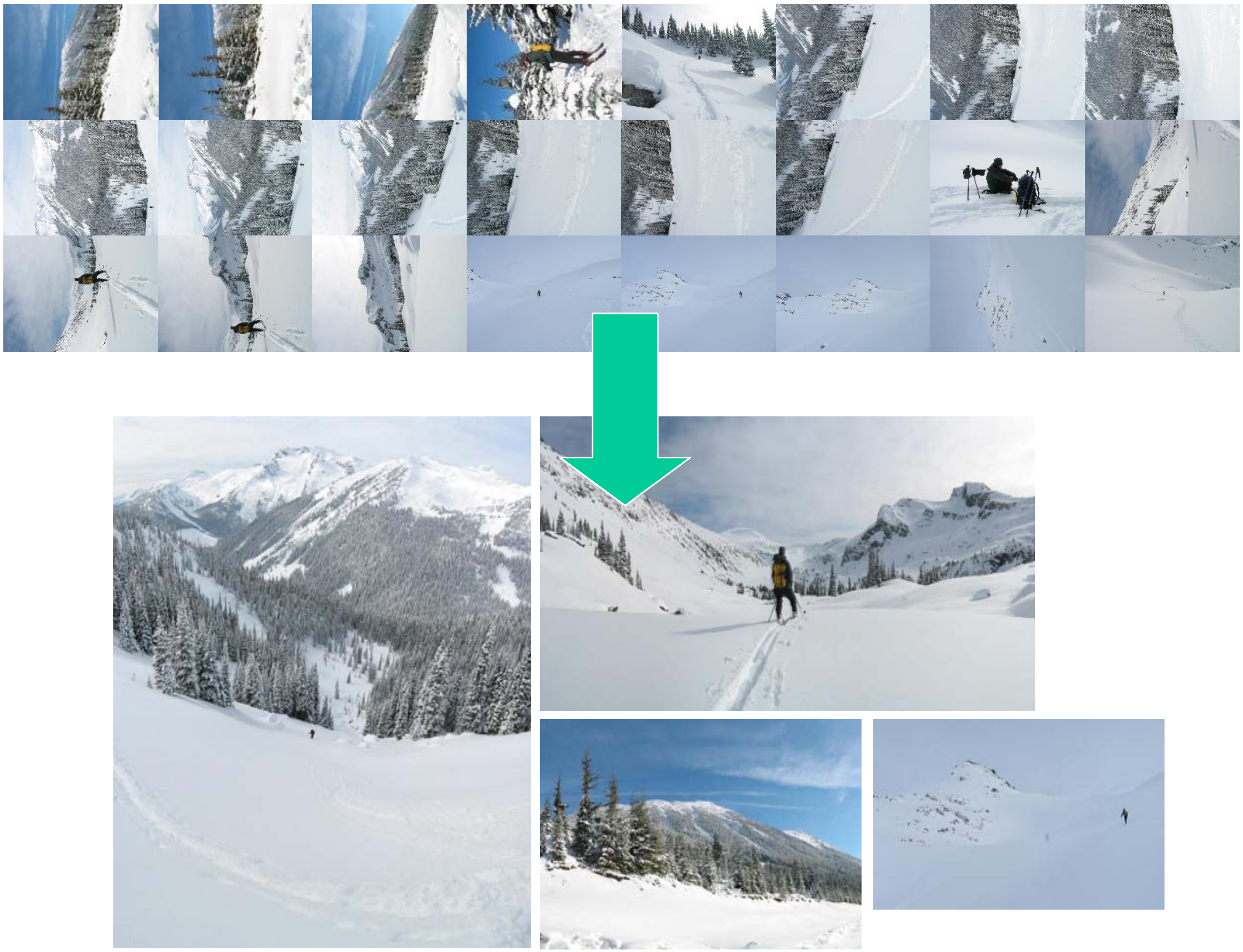
Why "Recognising Panoramas"?

1D Rotations (θ)

- Ordering $\Rightarrow$ matching images

- 2D Rotations (θ, ϕ)
  - Ordering $\nRightarrow$ matching images

Why “Recognising Panoramas”?



Overview

Feature Matching

Image Matching

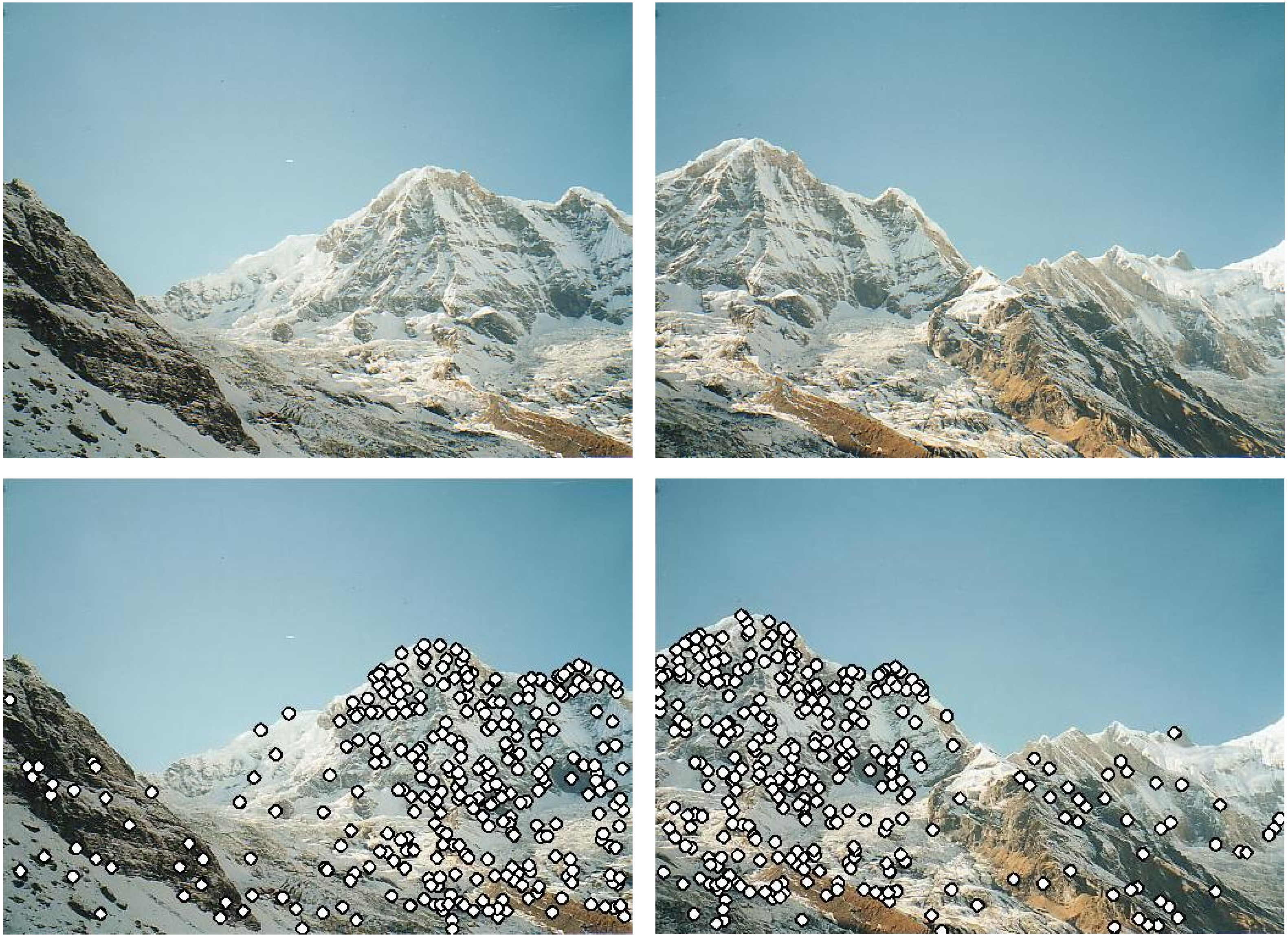
Bundle Adjustment

Multi-band Blending

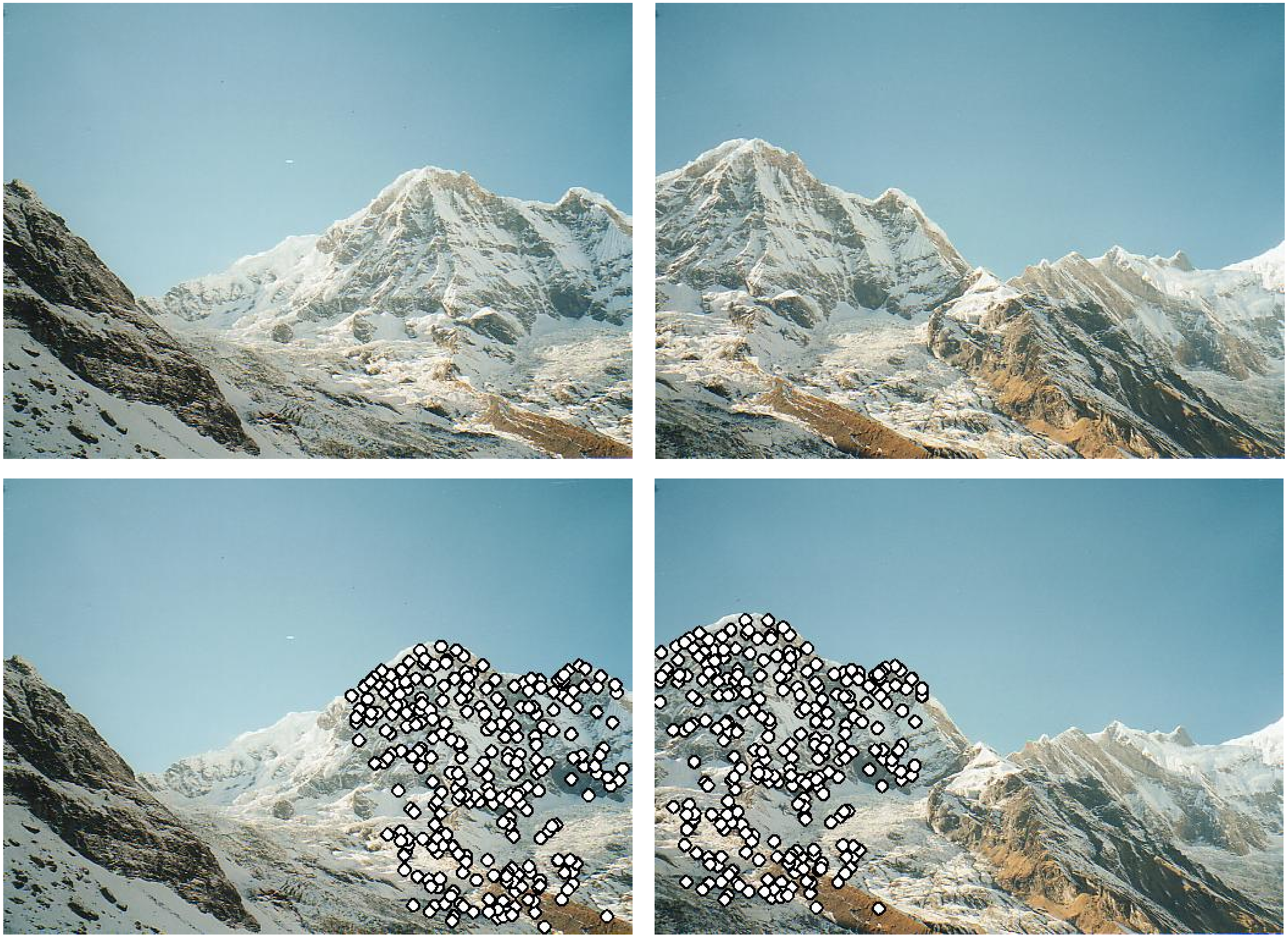
Results

Conclusions

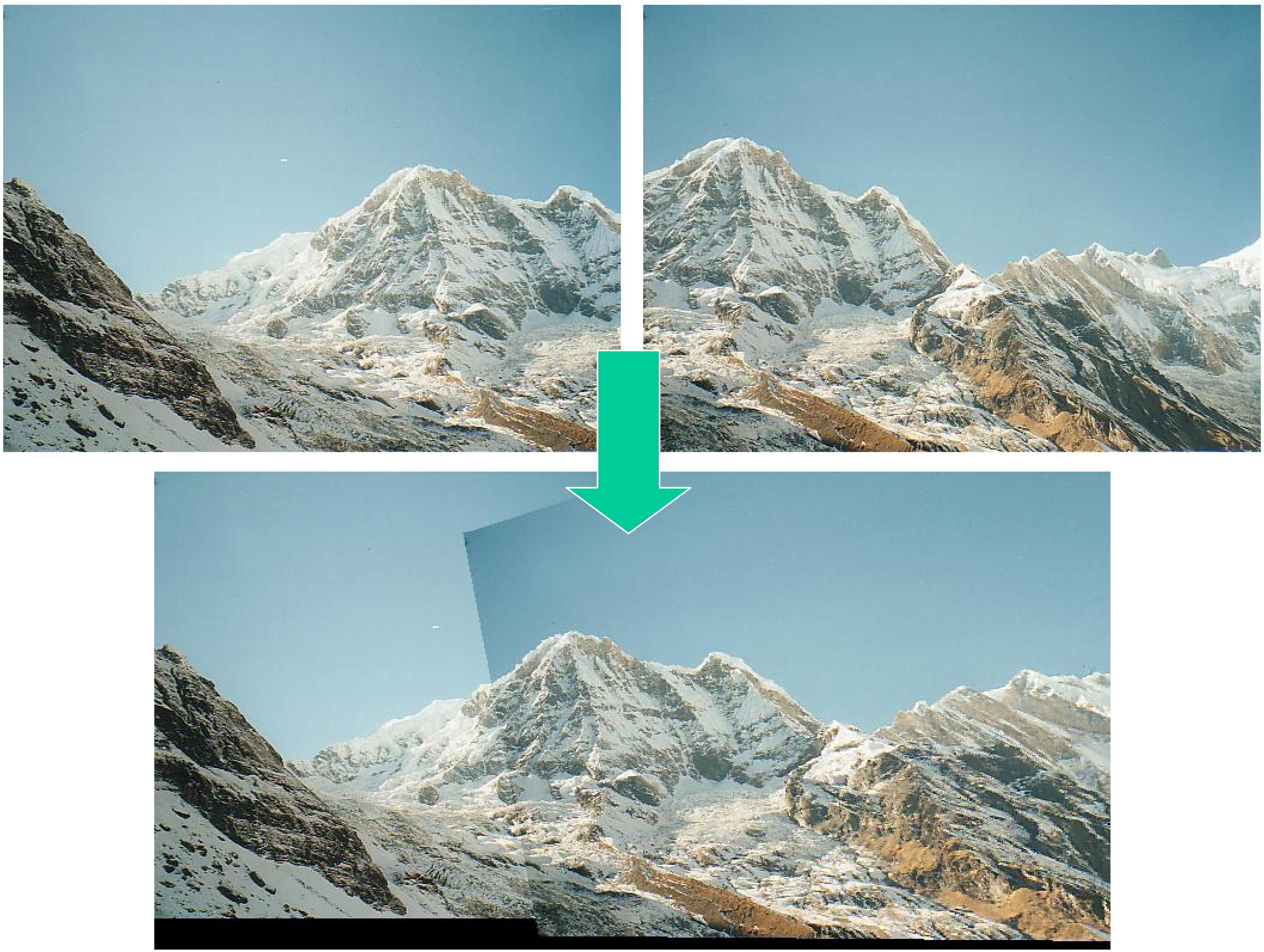
RANSAC for Homography



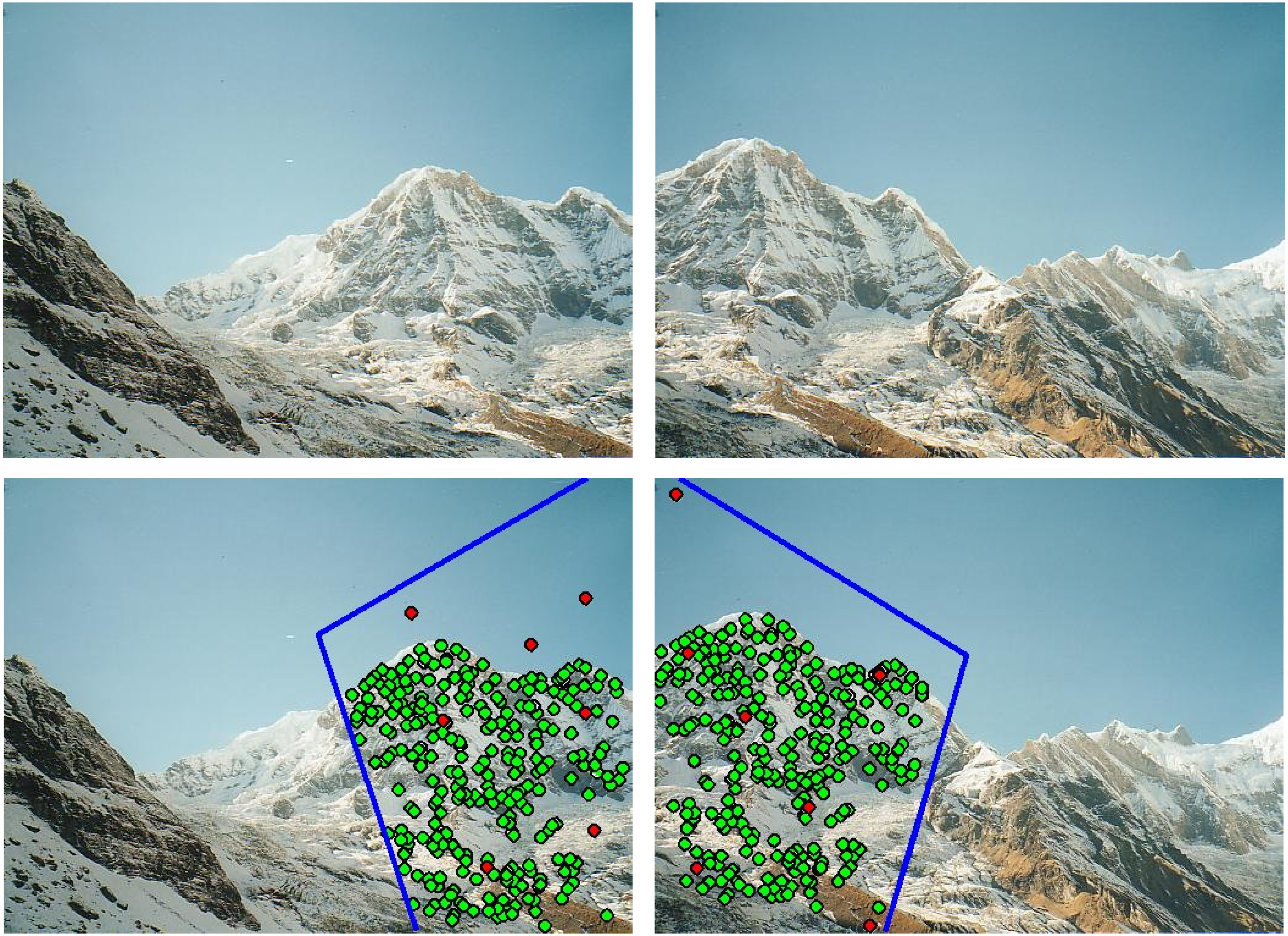
RANSAC for Homography



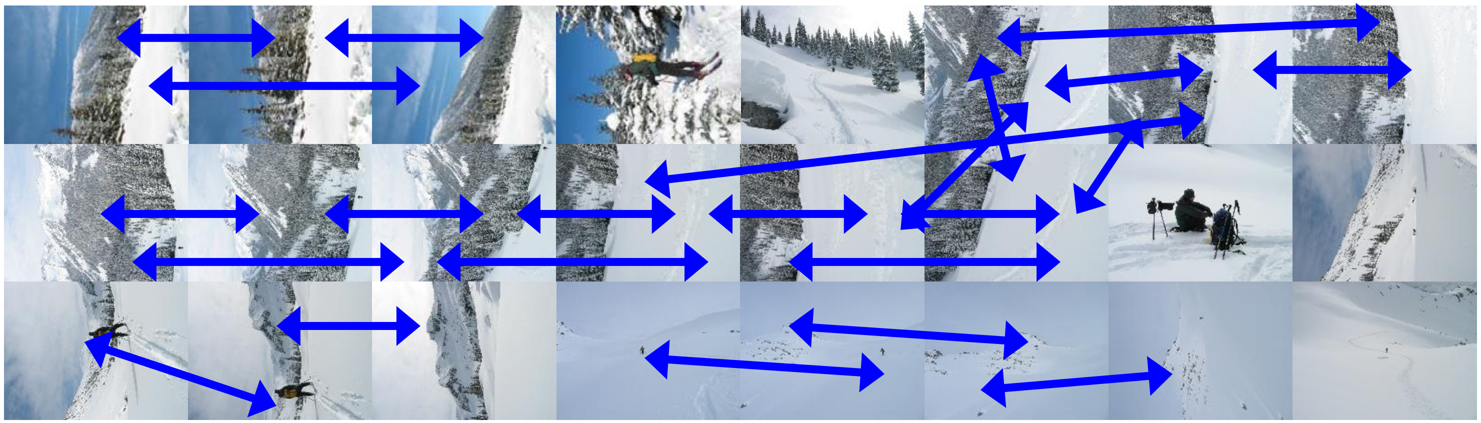
RANSAC for Homography



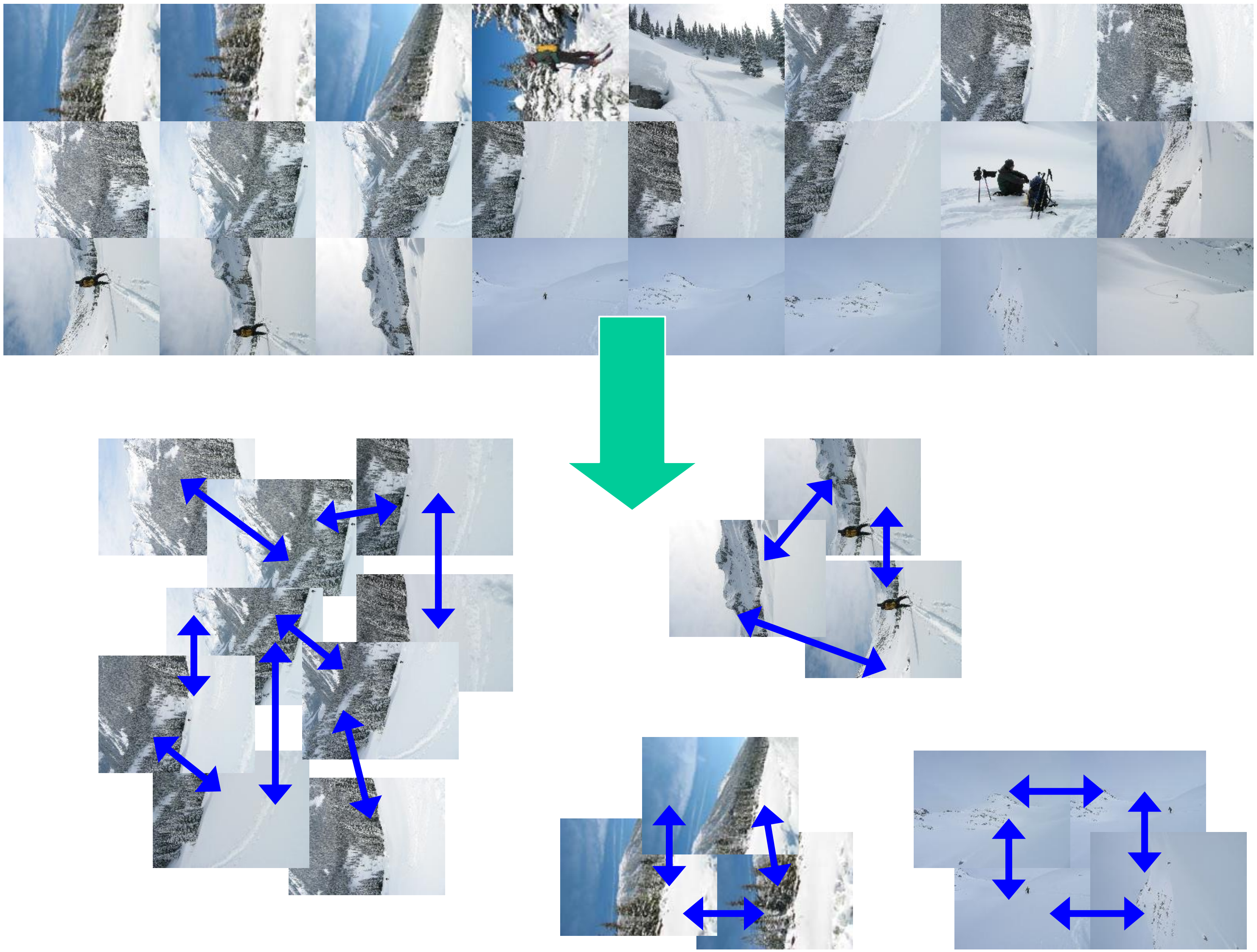
Probabilistic model for verification



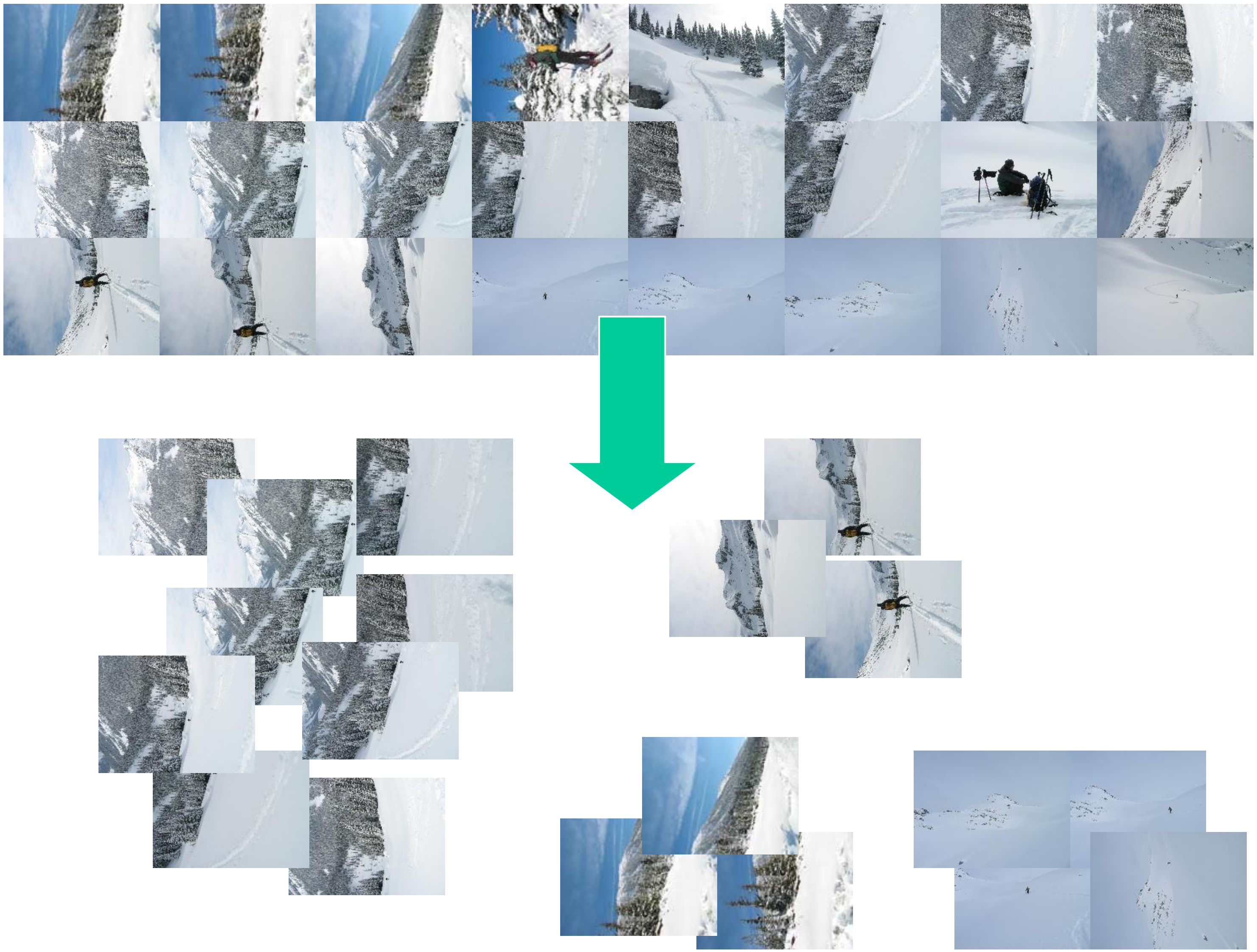
Finding the panoramas



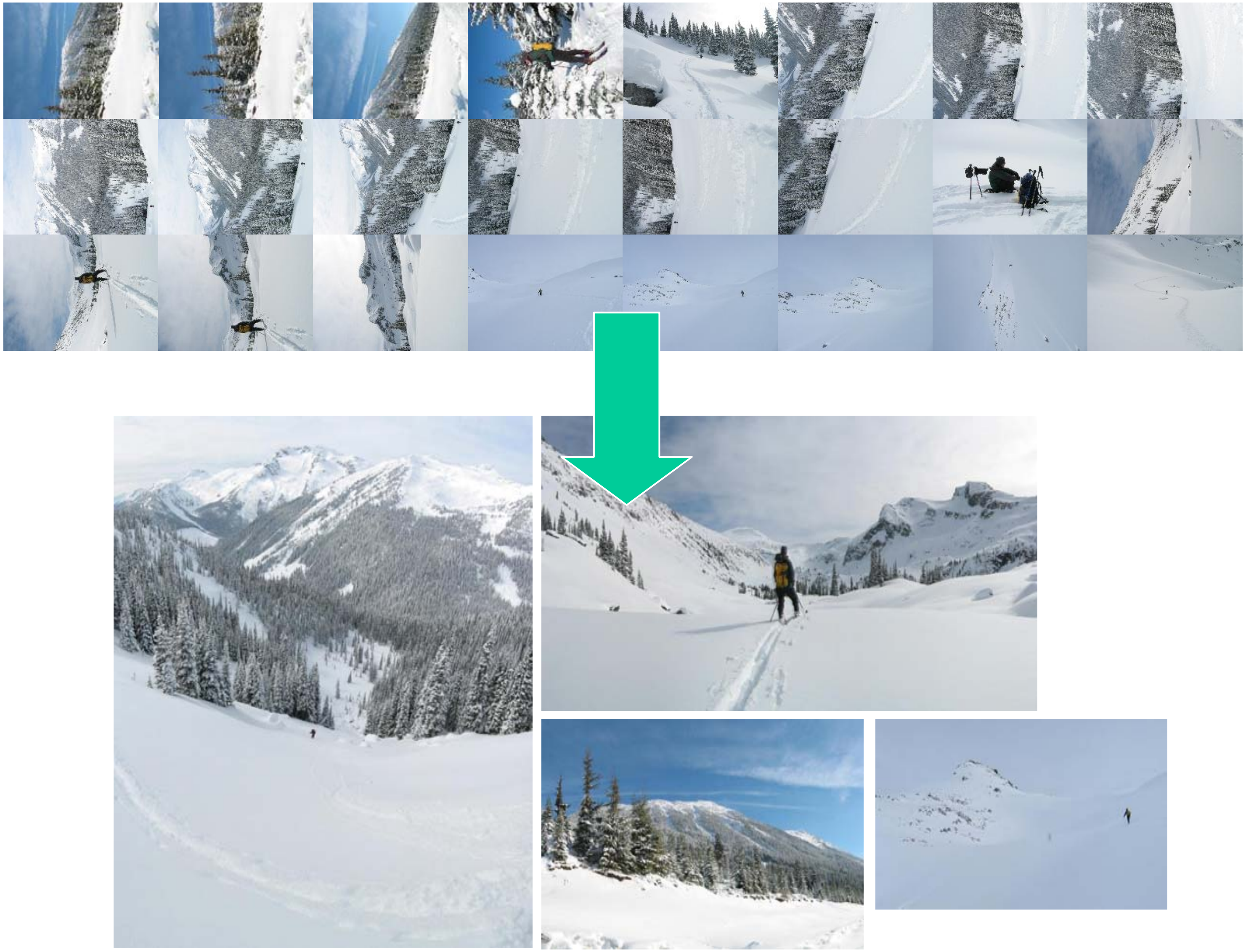
Finding the panoramas



Finding the panoramas



Finding the panoramas



Homography for Rotation

Parameterise each camera by rotation and focal length

$$\mathbf{R}_i = e^{[\boldsymbol{\theta}_i]_\times}, \ \ [\boldsymbol{\theta}_i]_\times = \begin{bmatrix} 0 & -\theta_{i3} & \theta_{i2} \\ \theta_{i3} & 0 & -\theta_{i1} \\ -\theta_{i2} & \theta_{i1} & 0 \end{bmatrix}$$

$$\mathbf{K}_i = \begin{bmatrix} f_i & 0 & 0 \\ 0 & f_i & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

This gives pairwise homographies

$$\tilde{\mathbf{u}}_i = \mathbf{H}_{ij} \tilde{\mathbf{u}}_j, \ \ \mathbf{H}_{ij} = \mathbf{K}_i \mathbf{R}_i \mathbf{R}_j^T \mathbf{K}_j^{-1}$$

Bundle Adjustment

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

Bundle Adjustment

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

Multi-band Blending

Burt & Adelson 1983

- Blend frequency bands over range $\propto \lambda$

Results