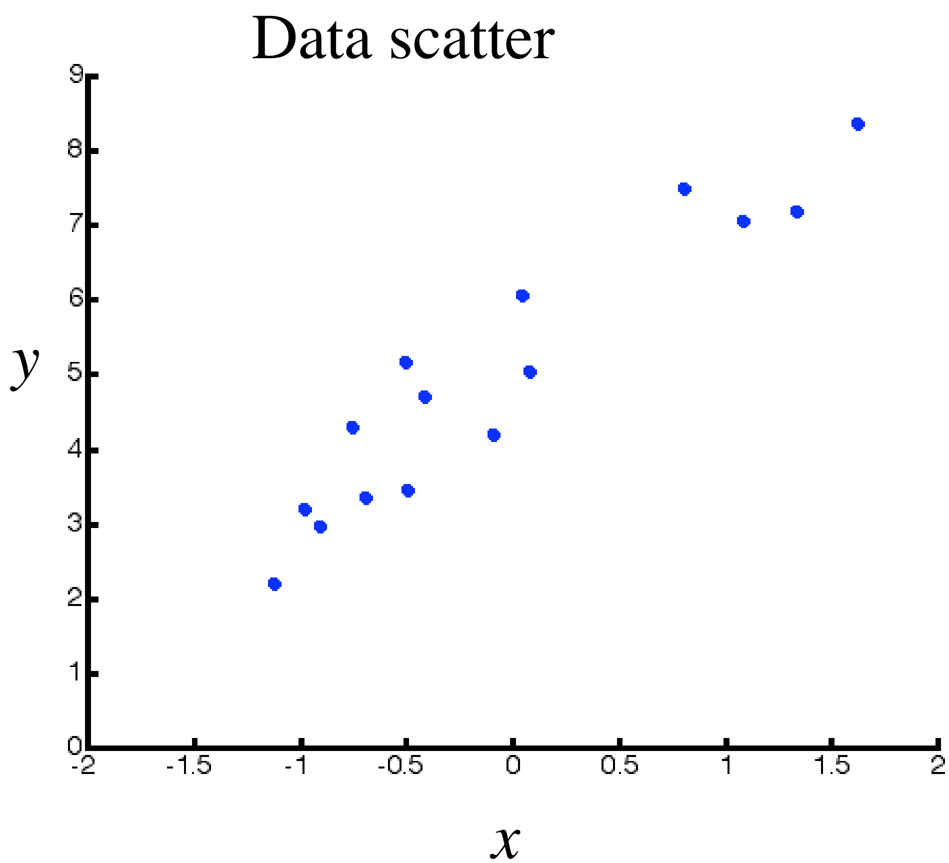
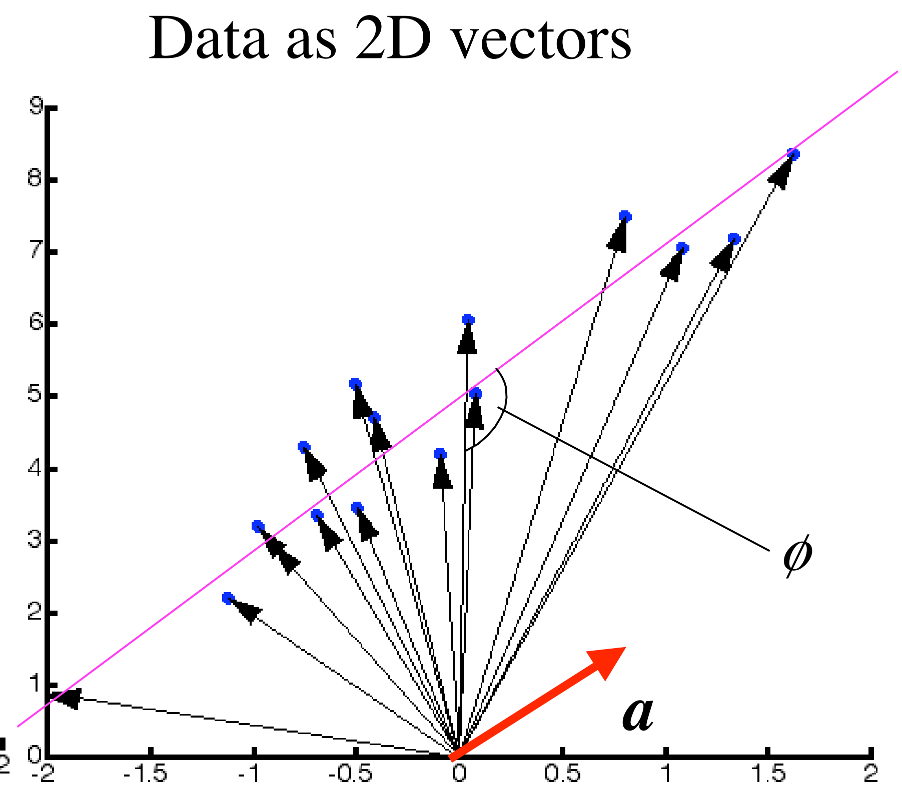
Example: Least Square Line Fitting



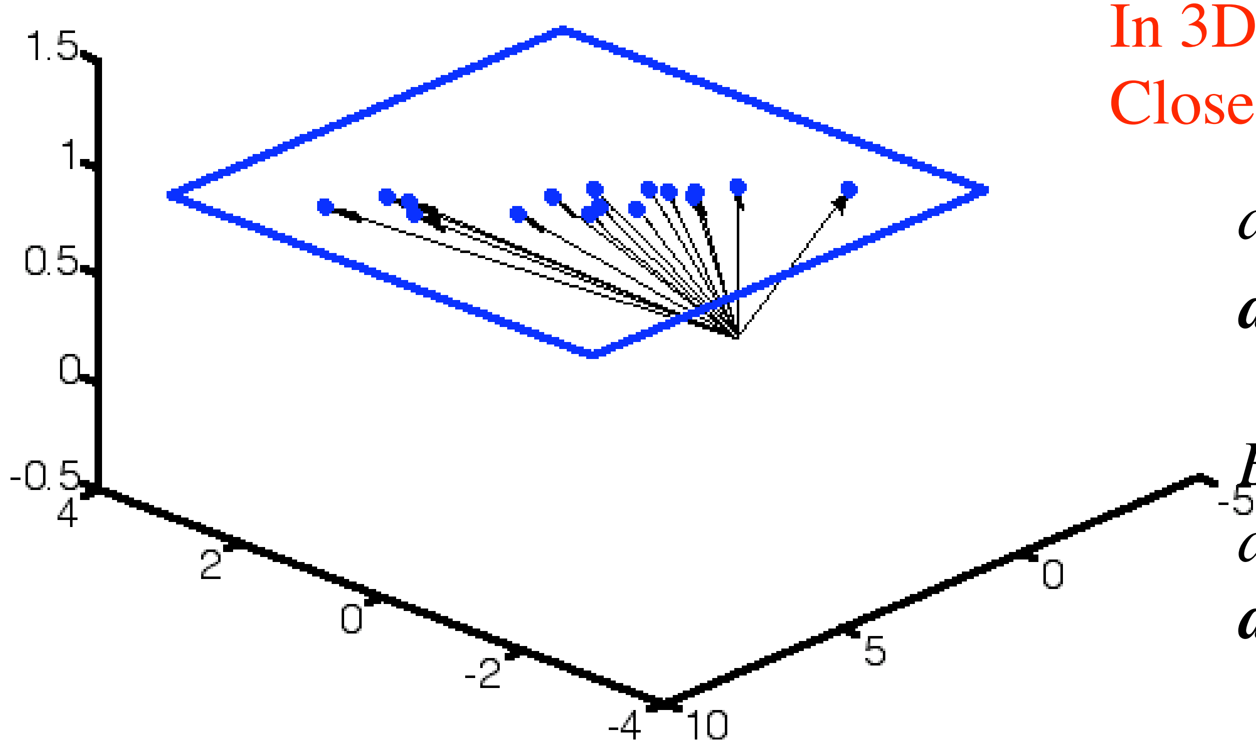


$a\, x_i + b\, y_i = c$

$\boldsymbol{a} \cdot \boldsymbol{x} = \boldsymbol{c} = \|\boldsymbol{x}\|| \cos \phi$

Introducing Homogenous Coordinates

Data as 3D homogenous vectors $p_i = [x_i \;\; y_i \;\; 1]'$



In 3D, the set of points lies
Close to a common plane

$a\, x_i + b\, y_i = c$

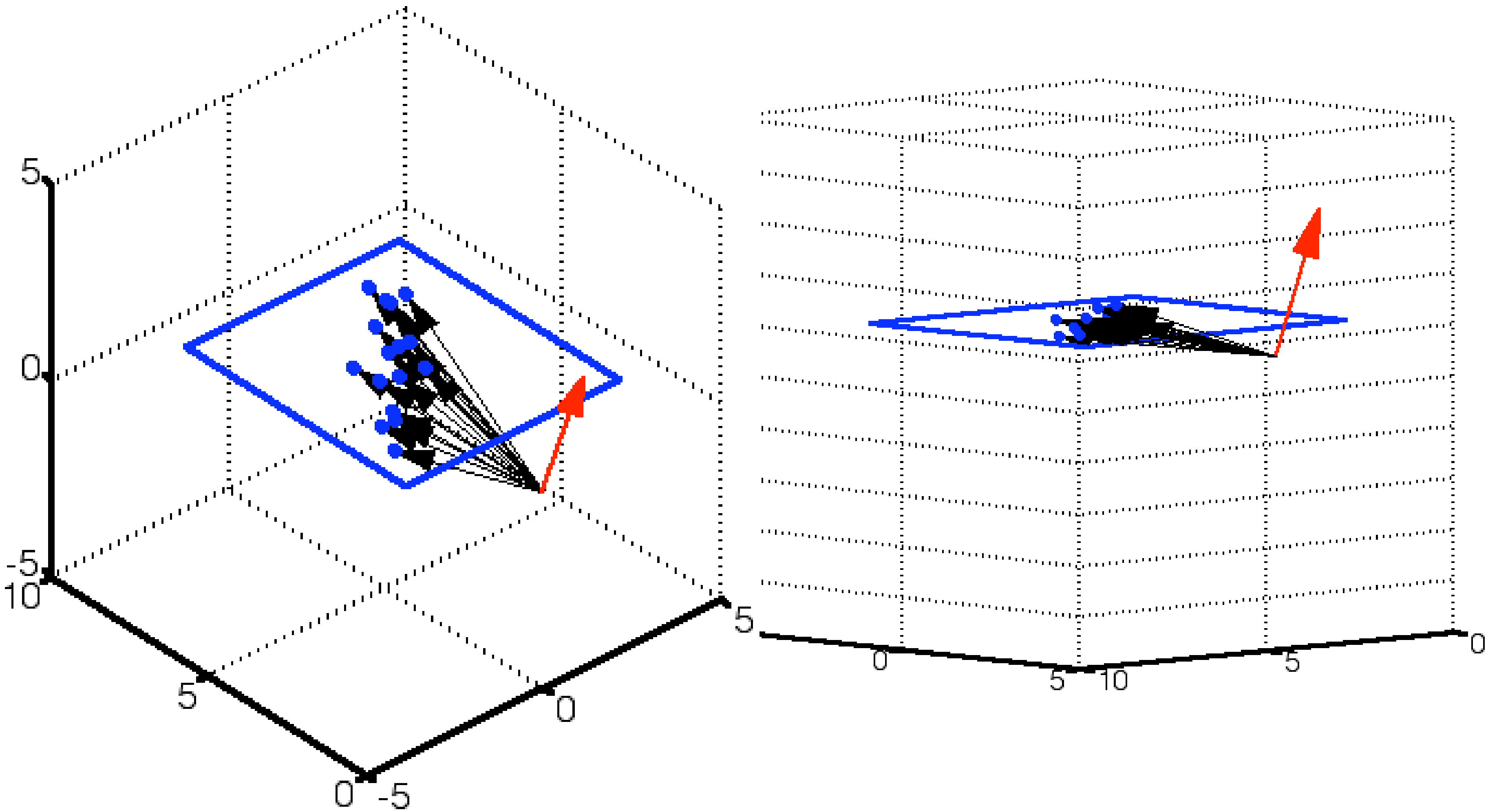
$\boldsymbol{a \cdot x} = \boldsymbol{c} = ||\boldsymbol{x}||| \cos \phi$

Becomes

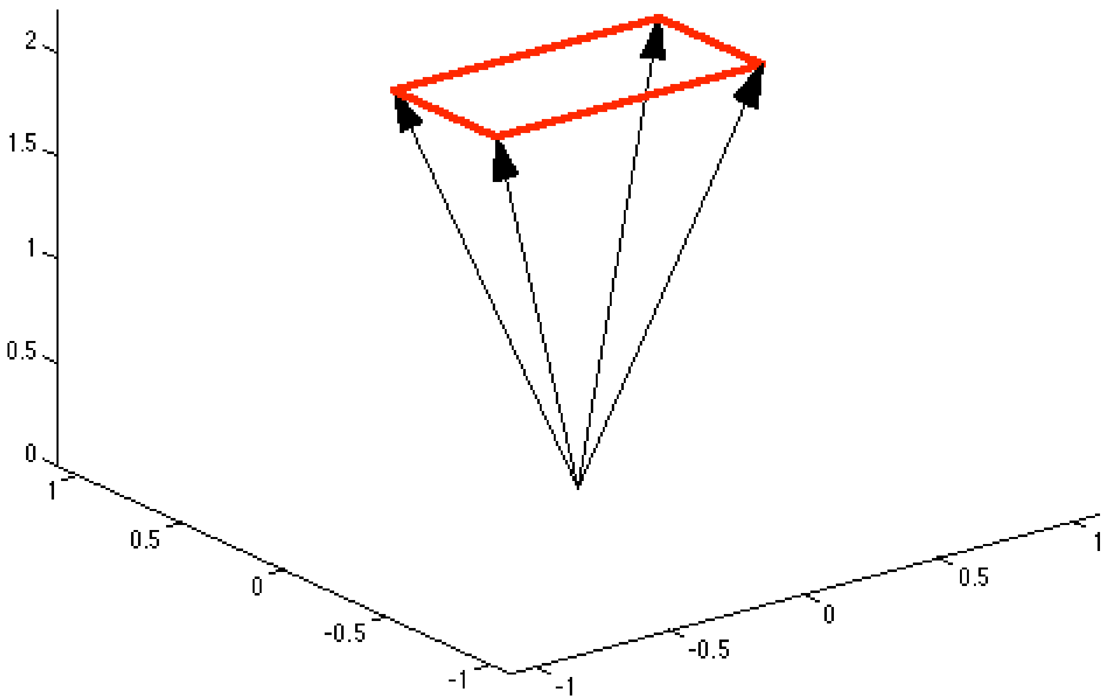
$a\, x_i + b\, y_i + c\, 1 = 0$

$\boldsymbol{a \cdot p_i} = \boldsymbol{0}$

Geometry of solution



$$A = \begin{pmatrix} -0.5 & -0.5 & 0.5 & 0.5 \\ 0.25 & -0.25 & -0.25 & 0.25 \\ 2 & 2 & 2 & 2 \end{pmatrix}$$



[U,S,V] = svd(A)

$$U = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

$$S = \begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \end{pmatrix}$$

- E.g. from image points alone we cannot determine latitude and longitude or which way is north.
- Can only determine the location of the road up to a Euclidean transformation from the world coordinate frame.

- Cannot determine scale from image points either.
- Hence the scaling factor.

- Combining the Euclidean transformation and the scaling factor gives us a similarity transform.

$$\mathbf{T}_{Sim} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & k \end{bmatrix}$$

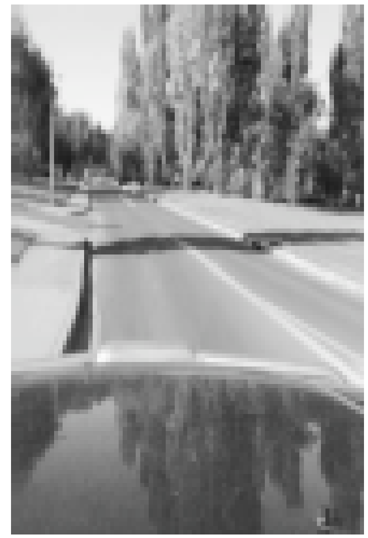
rotation ($\mathbf{R}$), translation ($\mathbf{t}$), scaling by k^{-1} (k)

Fundamental Matrix, why are 8 point matches enough?

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad f_{33} = 1$$

Thus only 8 free parameters =>
Need 8 or more constraints.

Stereo Reconstruction Ambiguity



- Without knowledge of scene's placement with respect to a 3D coordinate frame it is not possible to determine the absolute position and orientation of the scene from 2 (or any number of) views.

- What does this mean mathematically?
- Given
  - a set of 3D points $\tilde{\mathbf{X}}_i$,
  - two cameras $\mathbf{P}$, $\mathbf{P'}$ and
  - image points $\tilde{\mathbf{x}}_i$, $\tilde{\mathbf{x}}'_i$.

- Remember these are related by:

$$\tilde{\mathbf{x}}_i = \mathbf{P}\tilde{\mathbf{X}}_i$$

$$\tilde{\mathbf{x}}'_i = \mathbf{P}'\tilde{\mathbf{X}}_i$$

- Replacing

  $\tilde{\mathbf{X}}_i$ with $\mathbf{T}_{\text{sim}}\tilde{\mathbf{X}}_i$,

  $\mathbf{P}, \mathbf{P'}$ with $\mathbf{P}\mathbf{T}_{\text{sim}}^{-1}, \mathbf{P'}\mathbf{T}_{\text{sim}}^{-1}$

- Does not change the observed image points

$$\tilde{\mathbf{x}}_i = \mathbf{P}\tilde{\mathbf{X}}_i = (\mathbf{P}\mathbf{T}_{\text{sim}}^{-1})(\mathbf{T}_{\text{sim}}\tilde{\mathbf{X}}_i)$$

$$\tilde{\mathbf{x}}'_i = \mathbf{P'}\tilde{\mathbf{X}}'_i = (\mathbf{P'}\mathbf{T}_{\text{sim}}^{-1})(\mathbf{T}_{\text{sim}}\tilde{\mathbf{X}}'_i)$$

Extrinsic Parameter ambiguity

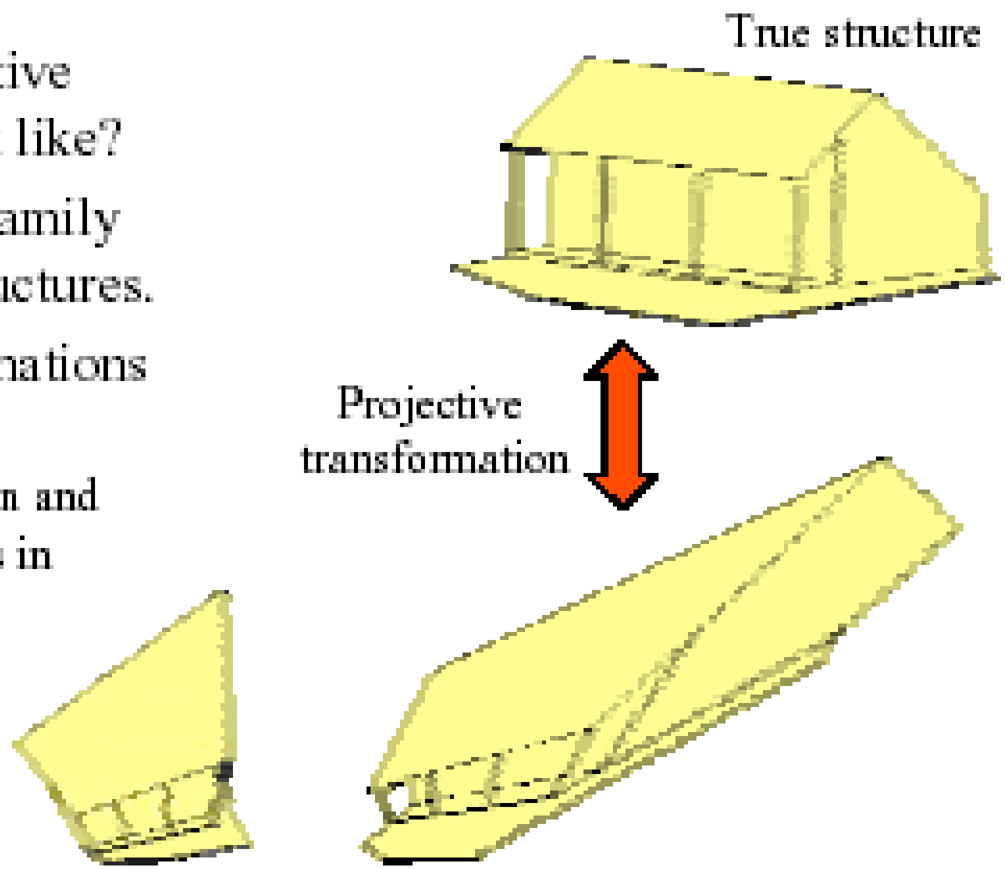
- If the camera calibration matrices are not known then the scene can only be constructed up to a *projective transformation* of the actual structure.
- A projective transformation is a homogeneous transformation of the form

$$\tilde{\mathbf{X}}_{new} = \mathbf{T}_{proj}\,\tilde{\mathbf{X}}$$

- where $\mathbf{T}_{proj}$ is any 4×4 invertible matrix.

Projective structure

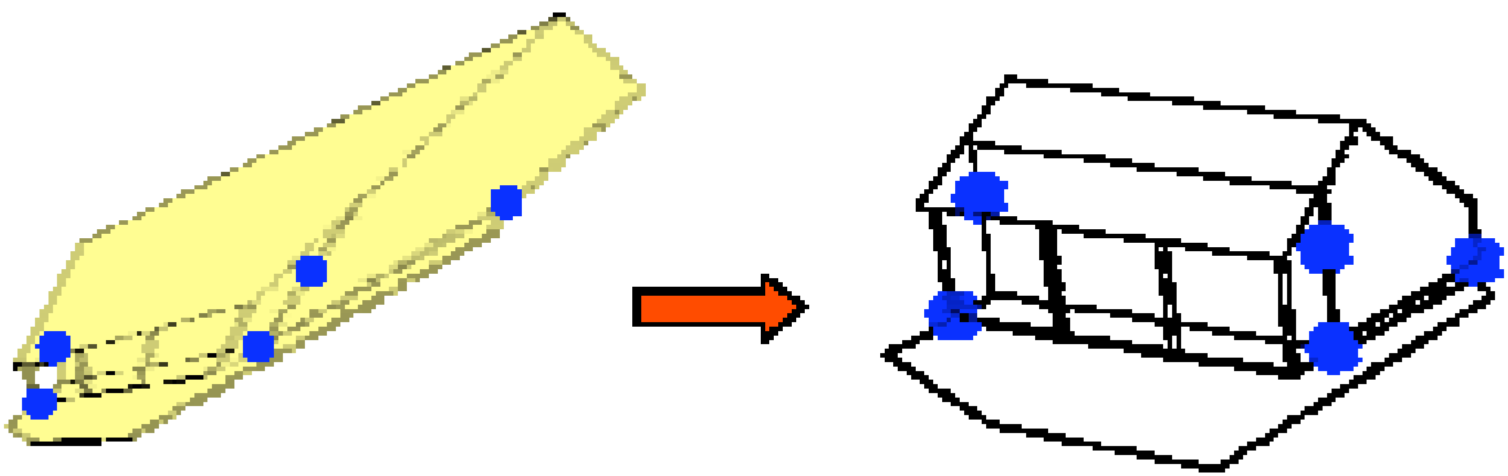
- What does a projective transformation look like?
- There are a whole family of these warped structures.
- Projective transformations
  - Map lines to lines
  - Preserve intersection and tangency if surfaces in contact.



2 views of a structure projectively equivalent to the true structure

Metric Structure

- If *control points* are available we can go from a projective to a true metric reconstruction.
- A projective reconstruction can be upgraded to a true metric reconstruction by specifying the 3D locations of 5 (or more) world points.



Projective reconstruction

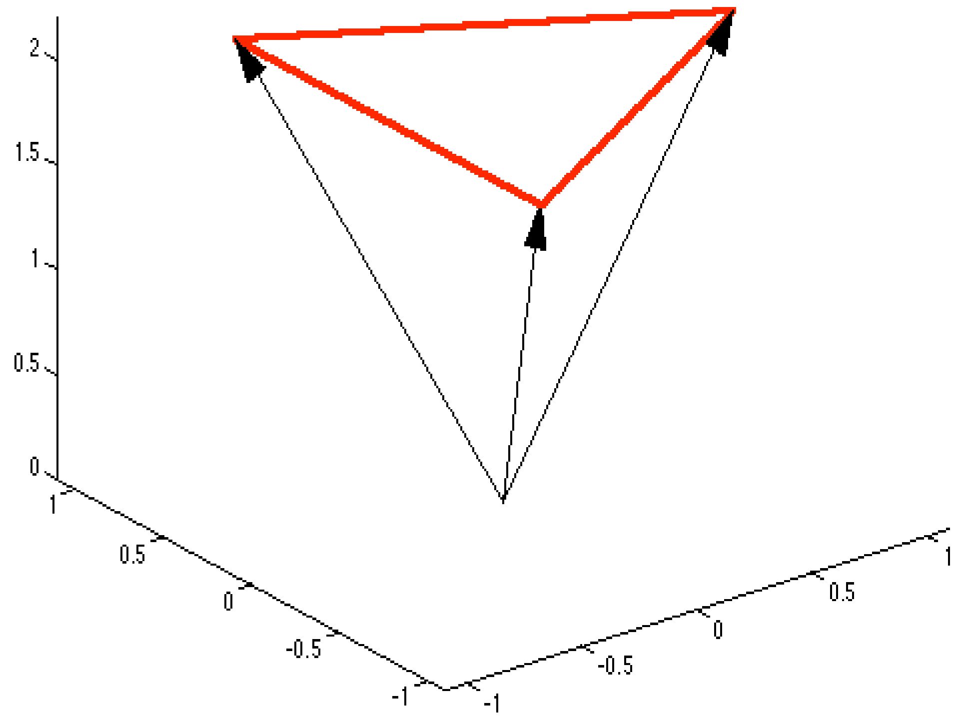
Metric reconstruction

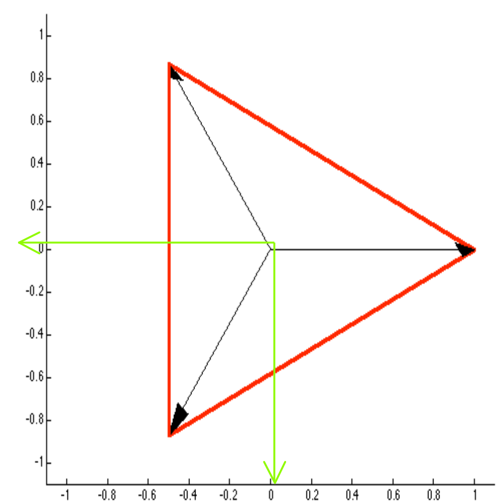
B =

$$\begin{matrix} 1 & -0.5 & -0.5 \\ 0 & 0.86603 & -0.86603 \\ 2 & 2 & 2 \end{matrix}$$

[U,S,V] = svd(B)

$$U = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{pmatrix}$$





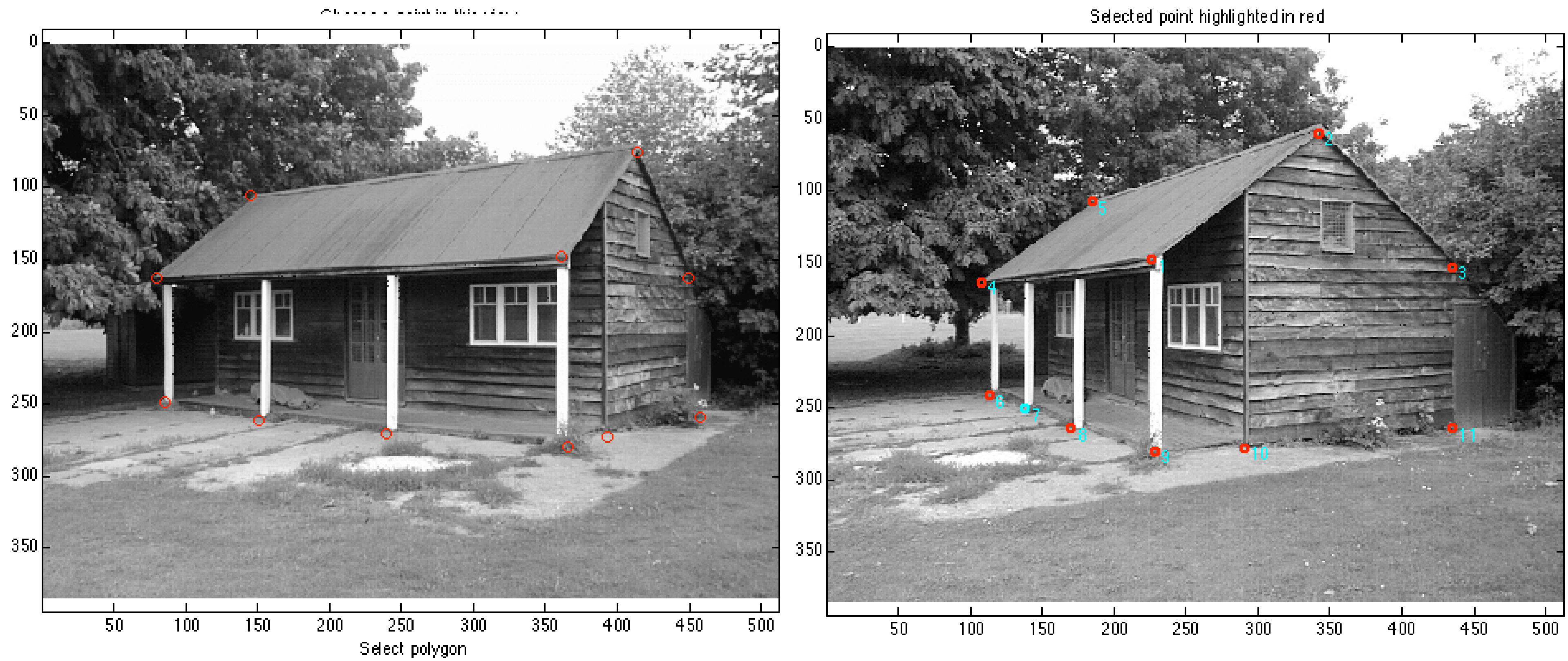
[V,D]=eig(B)

Yields z-axis and

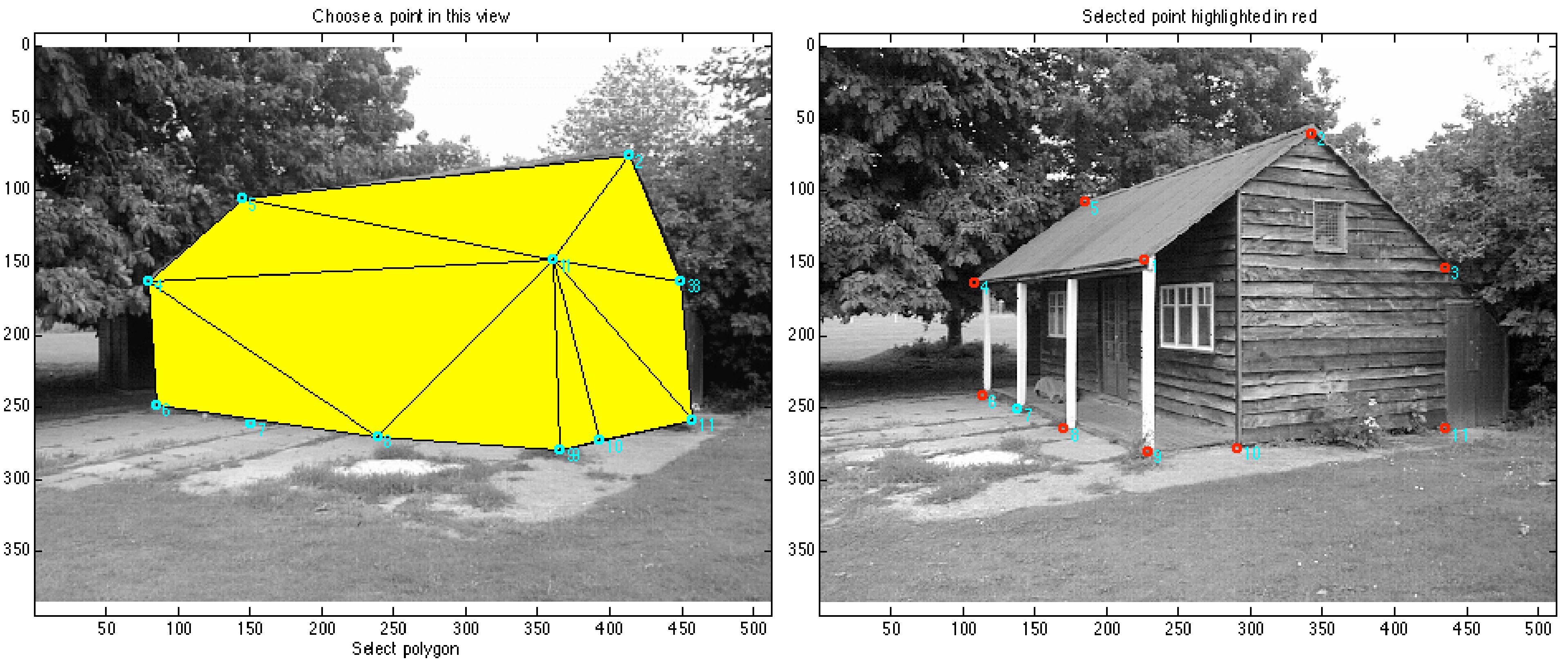
Complex eigenvalues

Representing the ambiguity

Matlab examples

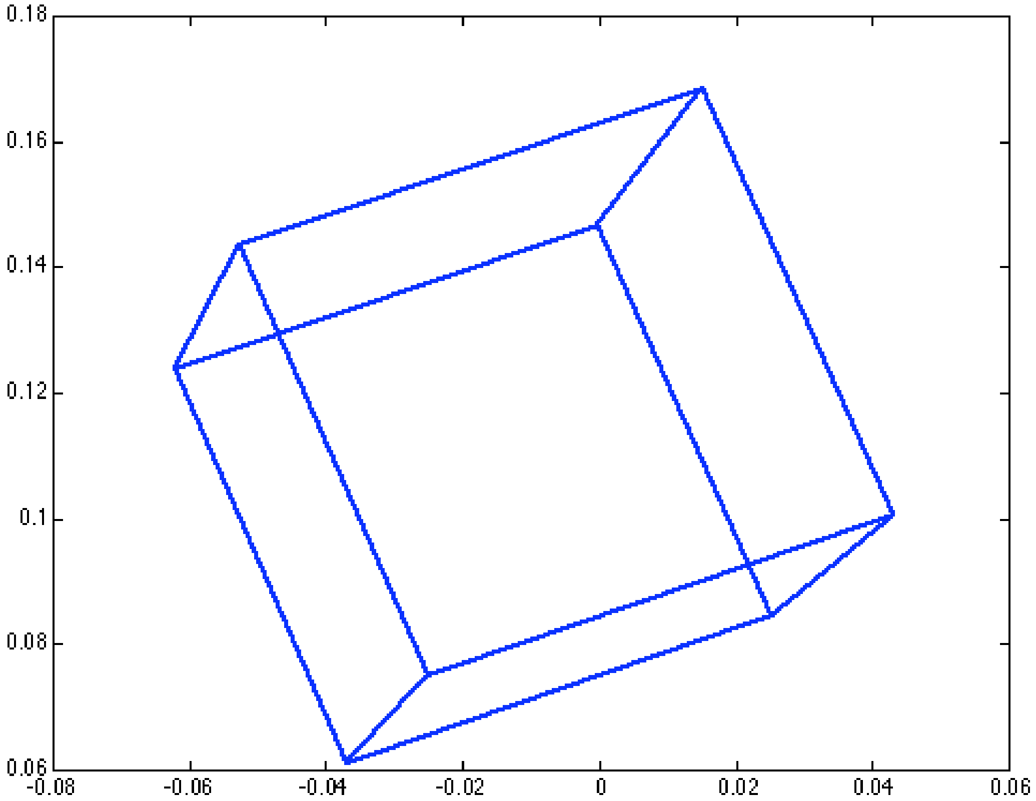


Select polygon
Selected point highlighted in red


Choose a point in this view
Select polygon
Selected point highlighted in red
0
50
100
150
200
250
300
350
400
450
500

Homework #2 solns

Problem #2
Projected image of a cube



Problem #3

```
% 3) Show that a straight line in 3D projects to a straight line
% in an image, using the simplest projection model, u = f x/z, v = f y/z.
% Hint: A straight line can be defined by two points L = a*p1+(1-a)*p2),
% where a is a scalar.
% p1 = [x1,y1,z1]';
% Show that a third point on the line p3 = a*p1+(1-a)*p2
% projects to an image point (u3,v3) that can be written in the form
% u3 = b u1 + (1-b) u2
% v3 = b v1 + (1-b) v2
% where b = a*z1/(a z1 + (1-a) z2)
```

$$p1 = 1/z1\ M\ x1$$
$$p2 = 1/z2\ M\ x2$$

$$x3 = a\ x1 + (1-a)\ x2$$

$$\begin{aligned} p3 &= 1/z3\ M\ x3 \\ &= a\ 1/z3\ M\ x1 + (1-a)\ 1/z3\ M\ x2 \\ &= a\ 1/z3\ (z1/z1)\ M\ x1 + (1-a)\ 1/z3\ (z2/z2)\ M\ x2 \\ &= a\ z1/z3\ p1 + (1-a)\ z2/z3\ p2 \end{aligned}$$

But now $x3 = a\ x1 + (1-a)\ x2$

And $z3 = a\ z1 + (1-a)\ z2$

Thus

$$p3 = a\ z1/(a\ z1 + (1-a)\ z2)\ p1 + (1-a)\ z2/(a\ z1 + (1-a)\ z2)\ p2$$

Set $b = a\ z1/(a\ z1 + (1-a)\ z2)$

Then $1-b = (1-a)\ z2/(a\ z1 + (1-a)\ z2)$

And we have:

$$p3 = b\ p1 + (1-b)\ p2$$ {QED}