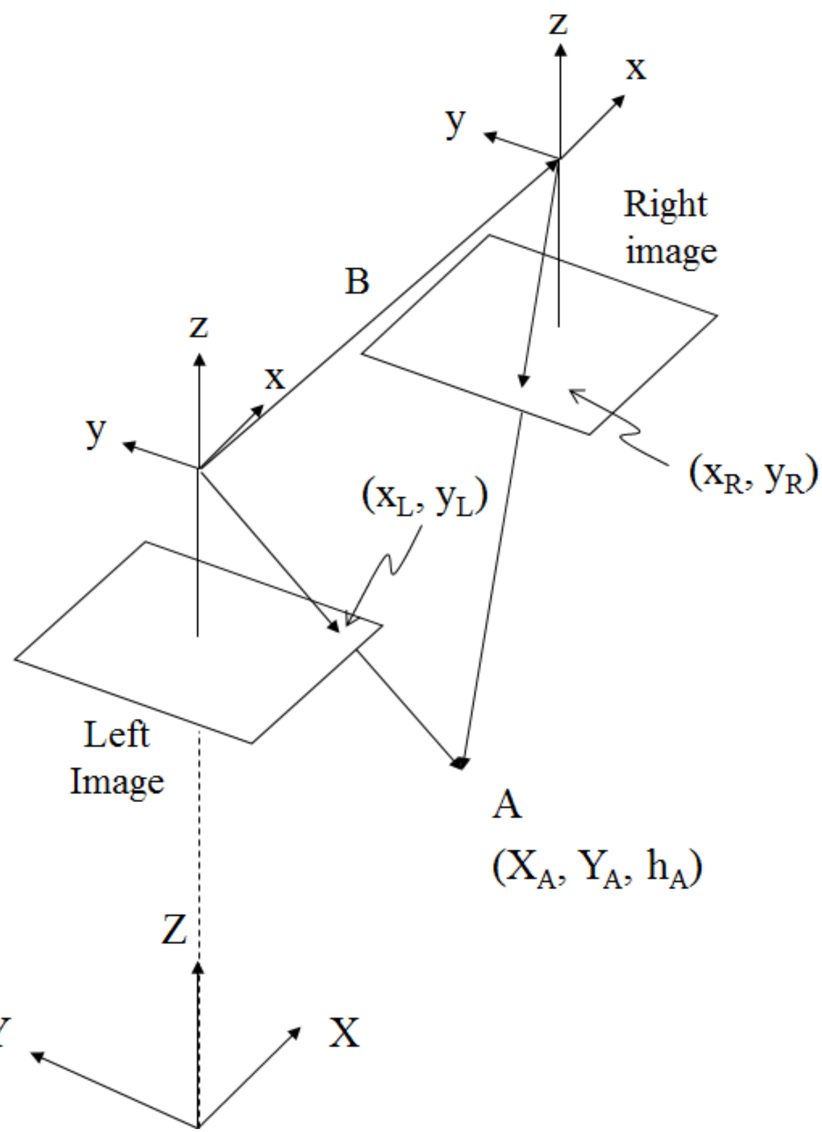
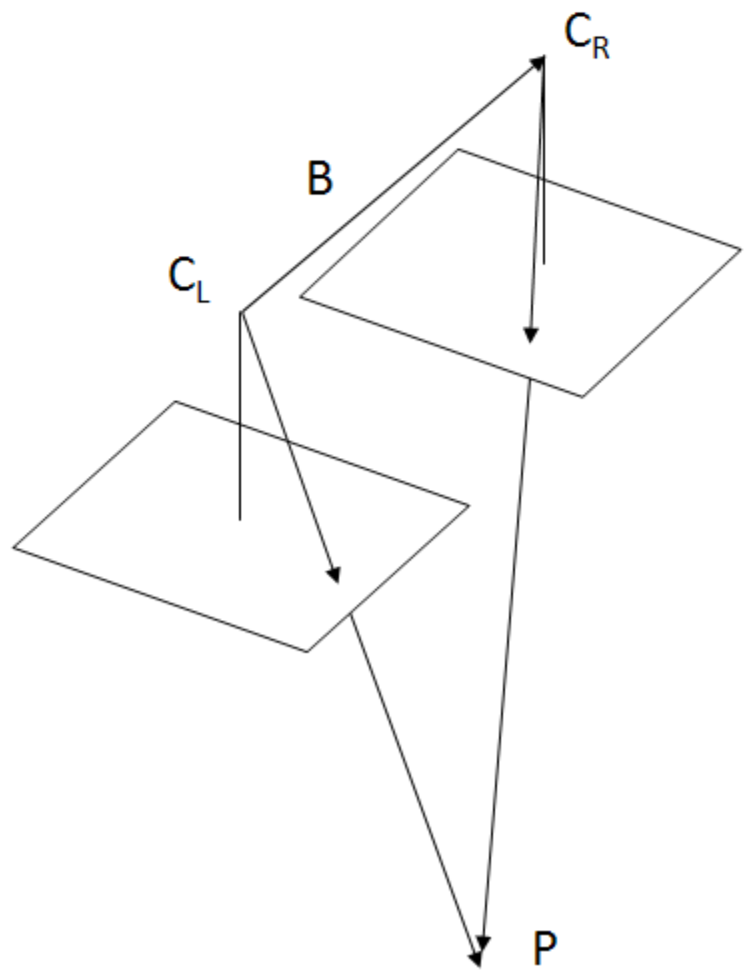
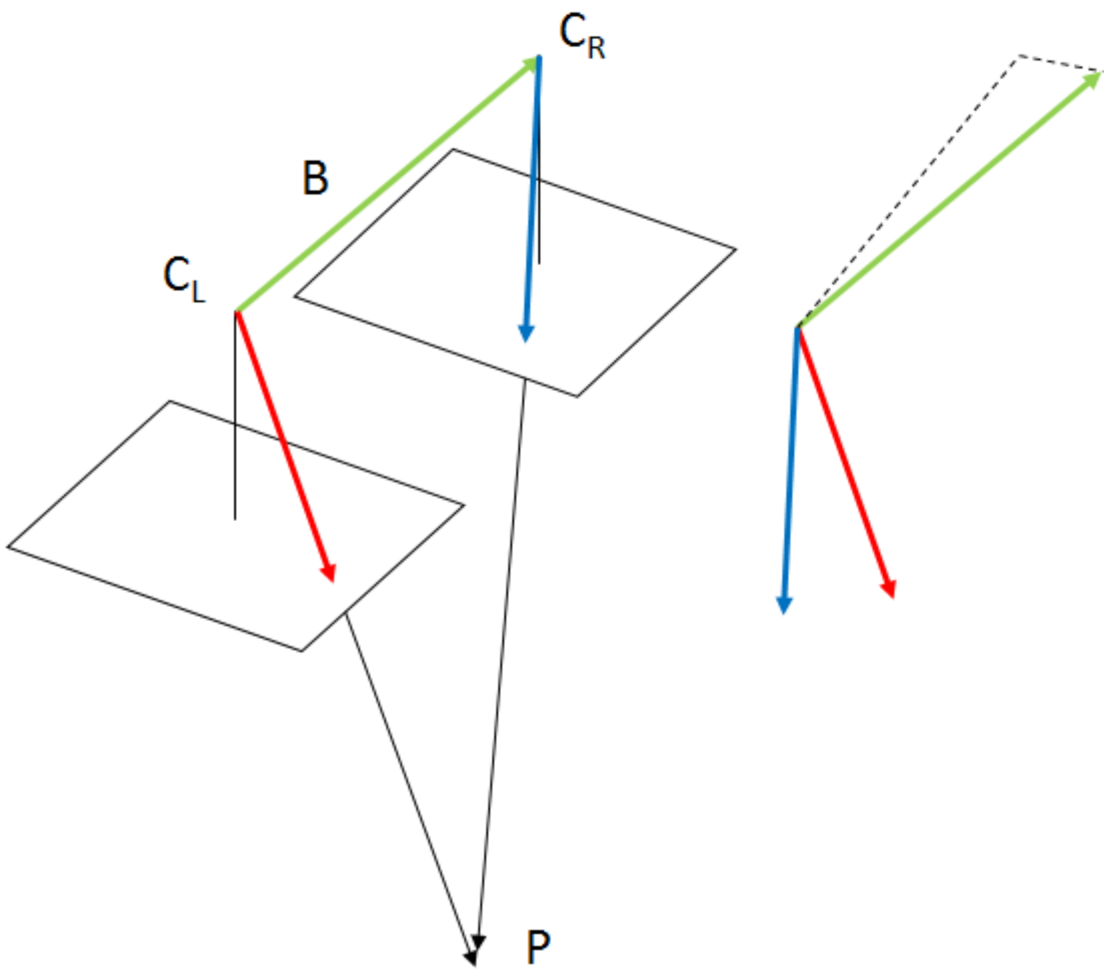
Stereo Image Geometry



Coplanarity



Coplanarity



Dependent Relative Orientation (RO)

<u>Left Image</u>

(Fixed)

$X^c=0$

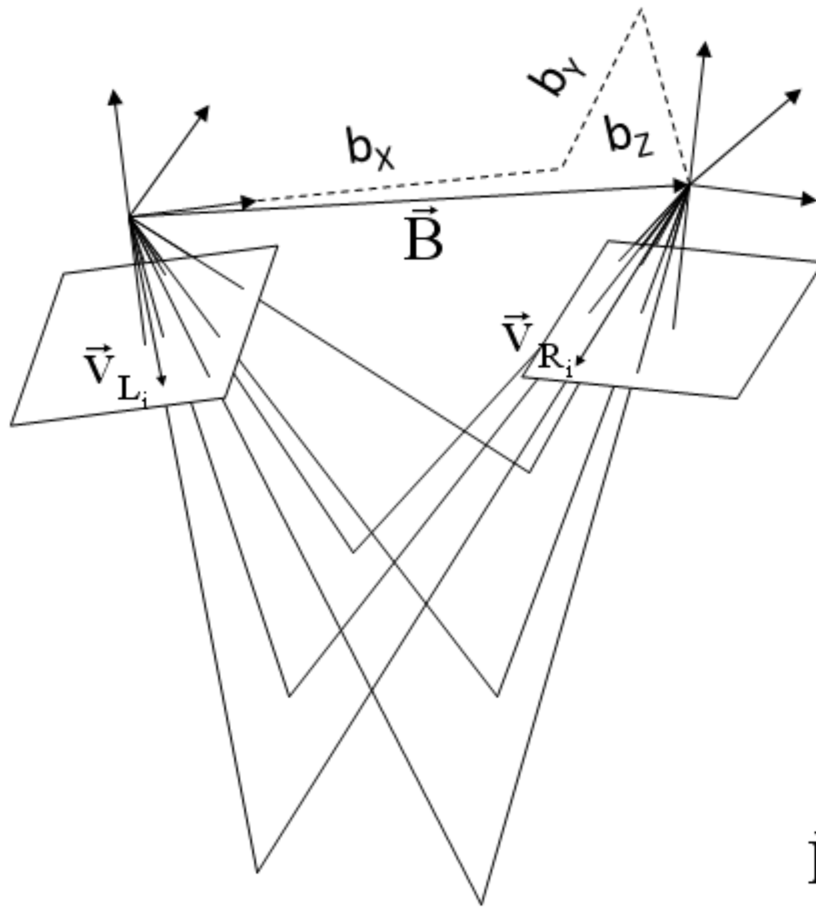
$Y^c=0$

$Z^c=0$

$\omega=0$

$\phi=0$

$\kappa=0$



<u>Right Image</u>

$X^c= b_X$ (fixed)

$Y^c= b_Y$ (solved)

$Z^c= b_Z$ (solved)

$\omega=0$ (solved)

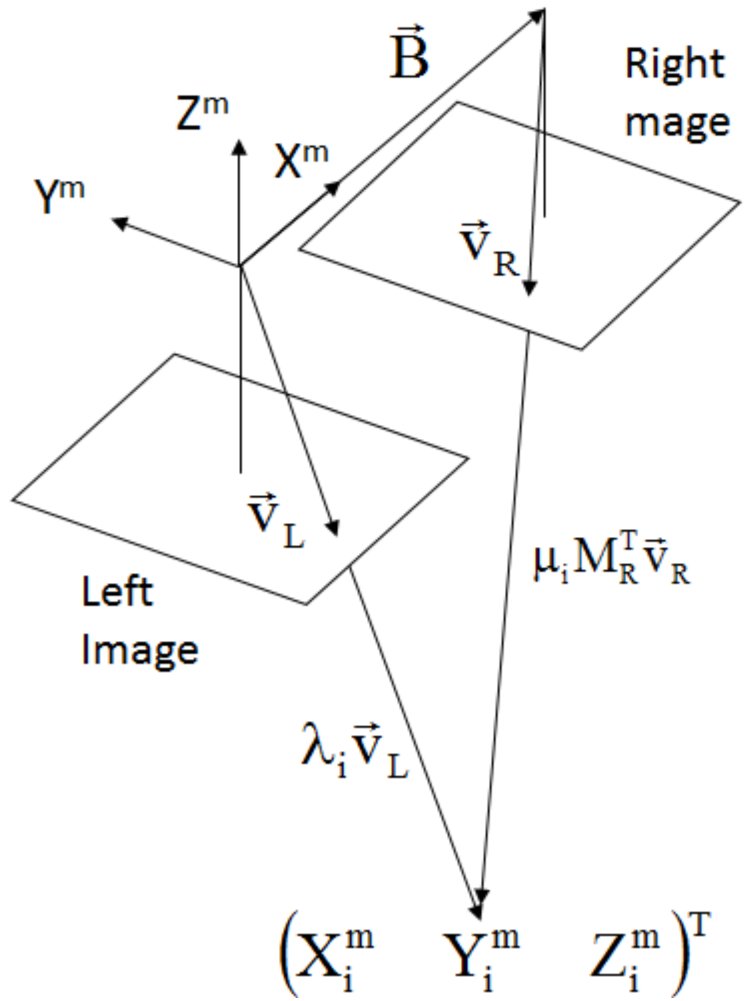
$\phi=0$ (solved)

$\kappa=0$ (solved)

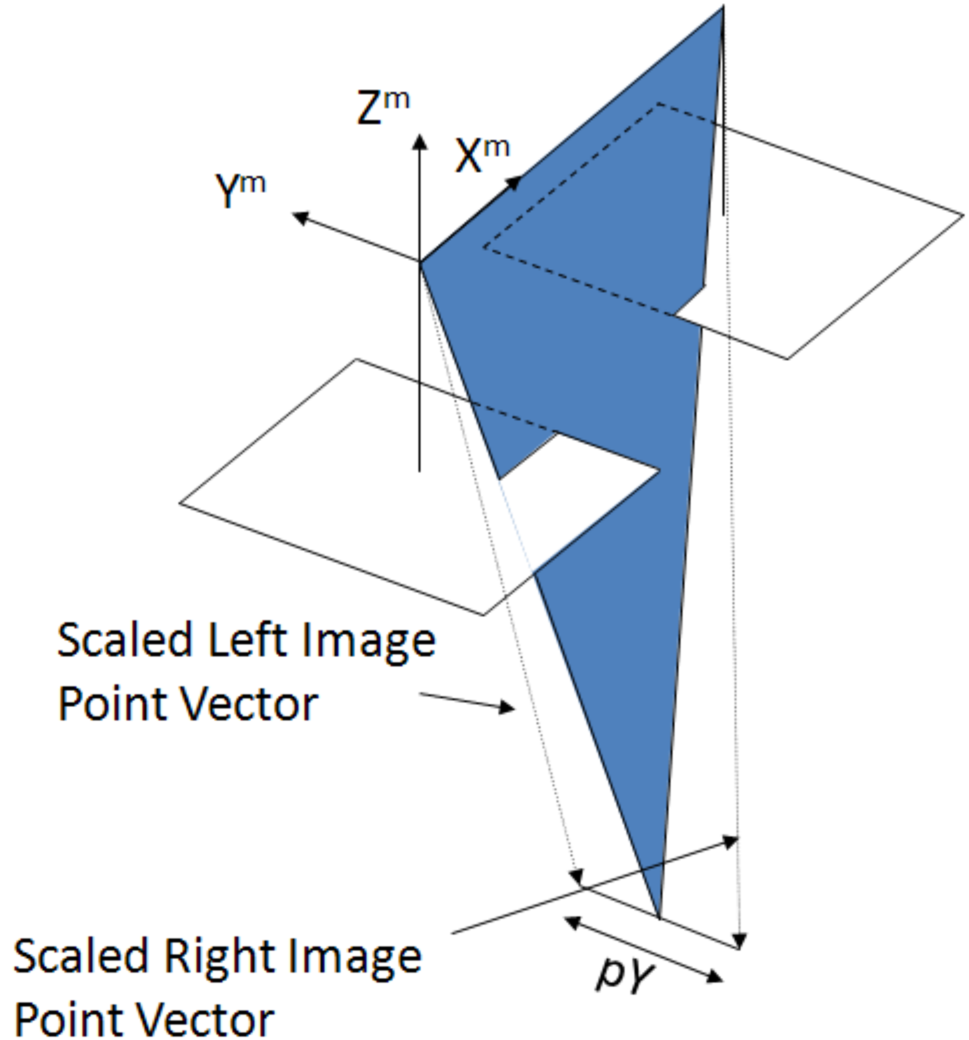
Base Vector

$$\vec{B} = \begin{pmatrix} b_X \\ b_Y \\ b_Z \end{pmatrix}$$

Space Intersection



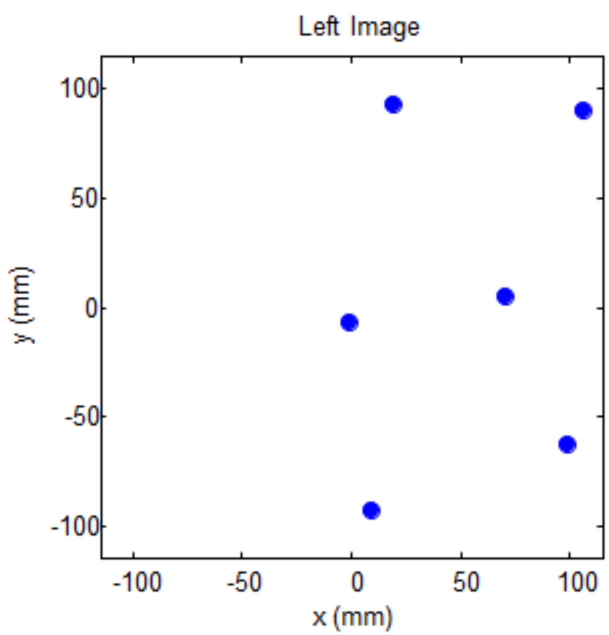
Y-Parallax

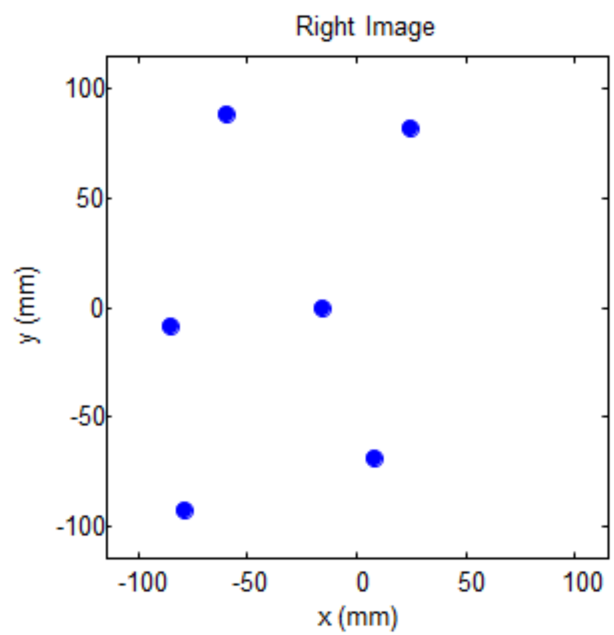


Z^m
X^m
Y^m
Scaled Left Image
Point Vector
Scaled Right Image
Point Vector
pY

Dependent RO Example

- RC8 camera (c = 152.15 mm)
  - 6 points used in the RO
  - IO: affine transformation, pp reduction, distortion, refraction
  - Point locations are indicated below





Dependent RO Example (cont'd)

- Observations, Y-parallaxes and model co-ordinates

id	xl (mm)	yl (mm)	xr (mm)	yr (mm)	pY (mm)	X (mm)	Y (mm)	Z (mm)
30	106.399	90.426	24.848	81.824	0.0030	108.9302	92.5786	-155.7695
40	18.989	93.365	-59.653	88.138	-0.0020	19.5304	96.0258	-156.4878
72	70.964	4.907	-15.581	-0.387	-0.0087	71.8751	4.9657	-154.1035
127	-0.931	-7.284	-85.407	-8.351	0.0067	-0.9473	-7.4078	-154.8060
112	9.278	-92.926	-78.81	-92.62	-0.0027	9.6380	-96.5329	-158.0535
50	98.681	-62.769	8.492	-68.873	0.0036	100.4898	-63.9177	-154.9389

- Estimated parameters

RO parameters	
bX (fixed) (mm)	92.0000
bY (mm)	5.0455
bZ (mm)	2.1725
omega (dd)	0.4392
phi (dd)	1.5080
kappa (dd)	3.1575

Dependent RO Example (cont'd)

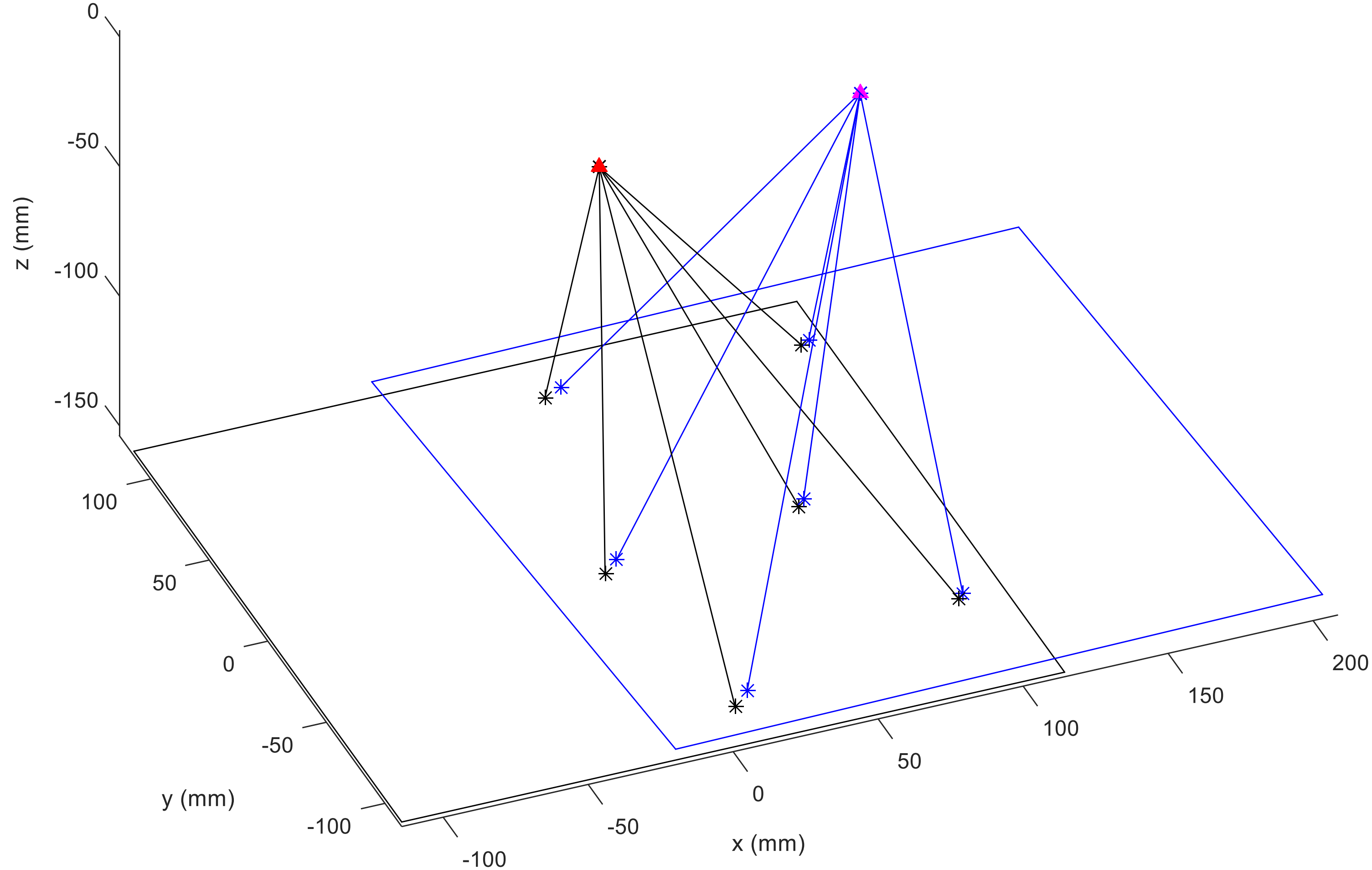
- Correlation coefficient matrix of the parameters

	bY	bZ	ω	ϕ	κ
bY	1				
bZ	0.25	1			
ω	-0.99	-0.25	1		
ϕ	-0.29	-0.71	0.26	1	
κ	0.07	-0.19	0.03	0.03	1

- Why is there high correlation between
  - b_Y and ω?
  - b_Z and ϕ?

Dependent RO Example (cont'd)

- Analyze the network geometry



Dependent RO Example (cont'd)

- Analyze the design matrix columns to find linear dependencies

$$A = [A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5] =$$

$$\begin{vmatrix} \boxed{\begin{matrix} 13.693 \\ 13.250 \\ 13.737 \\ 13.370 \\ 13.202 \\ 13.835 \end{matrix}} & \begin{matrix} 7.4894 \\ 8.0166 \\ 0.0901 \\ -0.6357 \\ -8.0956 \\ -6.1057 \end{matrix} & \boxed{\begin{matrix} 2818.2 \\ 2831.7 \\ 2146.8 \\ 2106.7 \\ 2937.6 \\ 2594.7 \end{matrix}} & \begin{matrix} 22.15 \\ 719.55 \\ 117.77 \\ 44.07 \\ -590.12 \\ 148.96 \end{matrix} & \begin{matrix} 386.93 \\ -807.64 \\ -220.84 \\ -1196.20 \\ -1059.30 \\ 142.19 \end{matrix} \end{vmatrix}$$

- In this case $A_1 \cong A_3$, but why?

Dependent RO Example (cont'd)

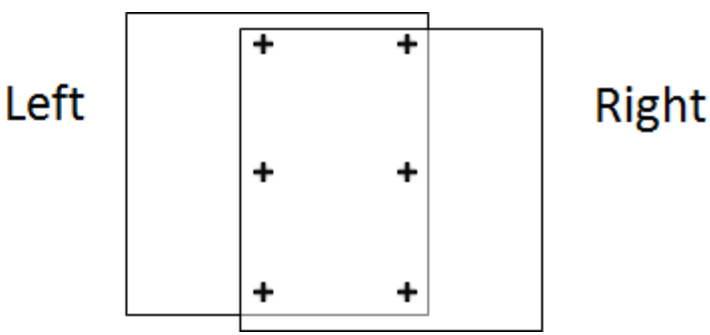
- Analyze the partial derivatives
- Since
  - $\omega \cong \phi \cong \kappa \cong 0$
  - $b_X >> b_Y$ and b_Z
  - $y_L \cong y_R = y$
- The partial derivatives are approximately

$$\frac{\partial \Delta}{\partial b_Y} = \begin{vmatrix} 0 & 1 & 0 \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ \bar{x}'_{R_i} & \bar{y}'_{R_i} & \bar{z}'_{R_i} \end{vmatrix} \approx c p_x \qquad \frac{\partial \Delta}{\partial \omega} = \begin{vmatrix} b_X & b_Y & b_Z \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ 0 & -\bar{z}'_{R_i} & \bar{y}'_{R_i} \end{vmatrix} \approx b_X \left(y^2 + c^2 \right)$$

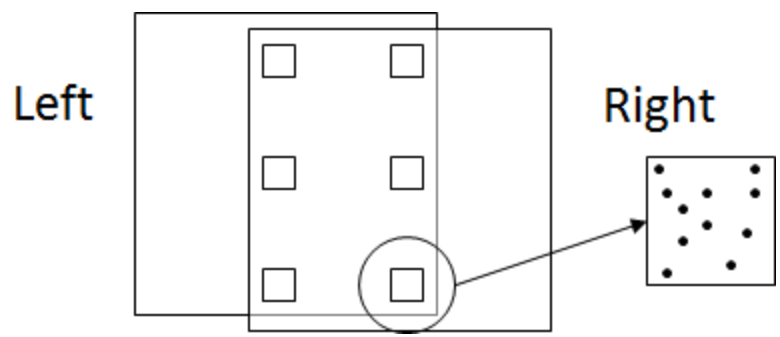
  - The x-parallax, p_x, is approximately constant
  - $c^2 >> y^2$
- So, both derivatives are nearly constant

Autonomous Relative Orientation

- Traditional RO
  - Operator selects and measures points
  - Points selected in standard locations
  - Only a few (e.g. 6) points
  - Points may be measured stereoscopically
  - Points measured very accurately, say ±5 μm

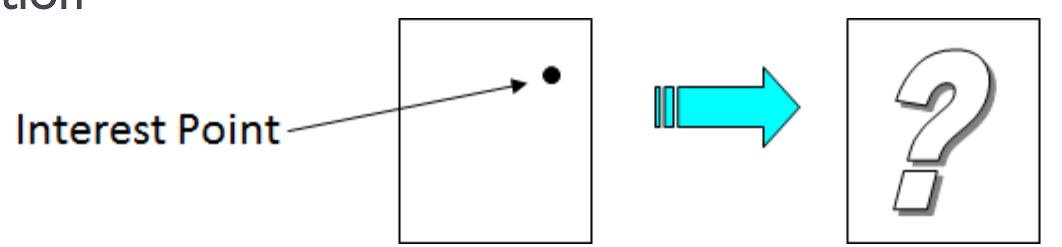


- Autonomous RO
  - Features are extracted automatically
  - Points randomly selected
  - Large number of features and high redundancy
  - Points are automatically matched across images
  - Feature matching accuracy may be low (i.e. ± a few pixels)



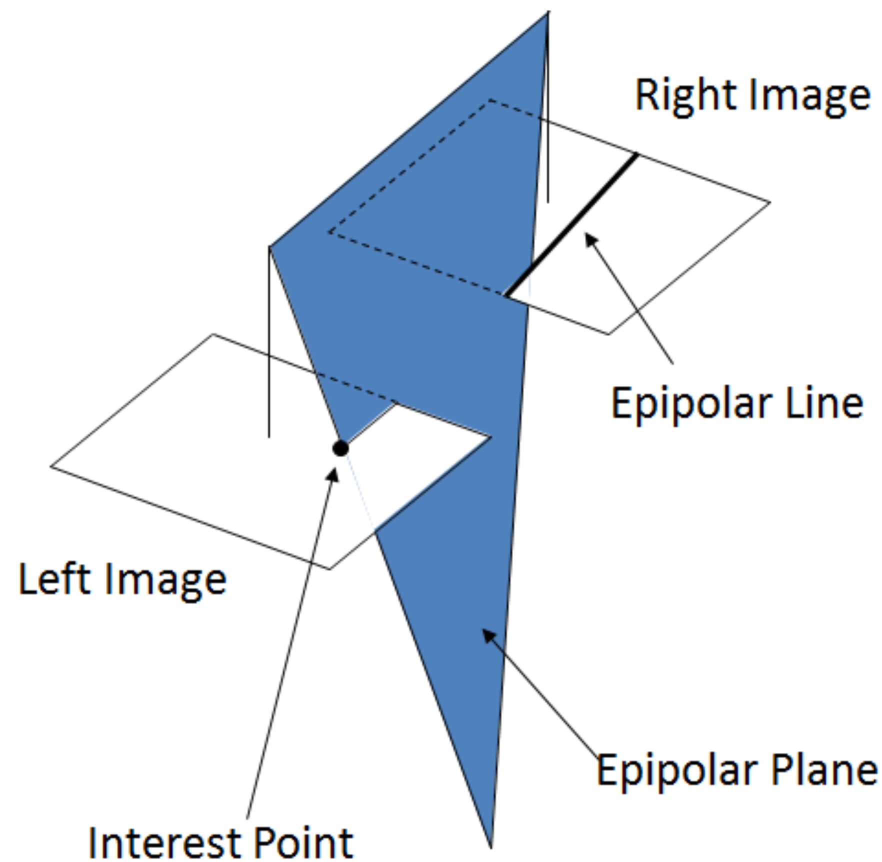
Autonomous Relative Orientation (cont'd)

- Autonomous RO can be performed using
  - Interest points
  - Edge pixels
  - Edge entities
- Interest points
  - Interest point: a point distinct from its surroundings (high radiometric variance)
  - An interest point operator is passed over each image
  - For an interest point in the left image, a ***search space*** is defined in the right image in which a search is performed for the most probable match point location



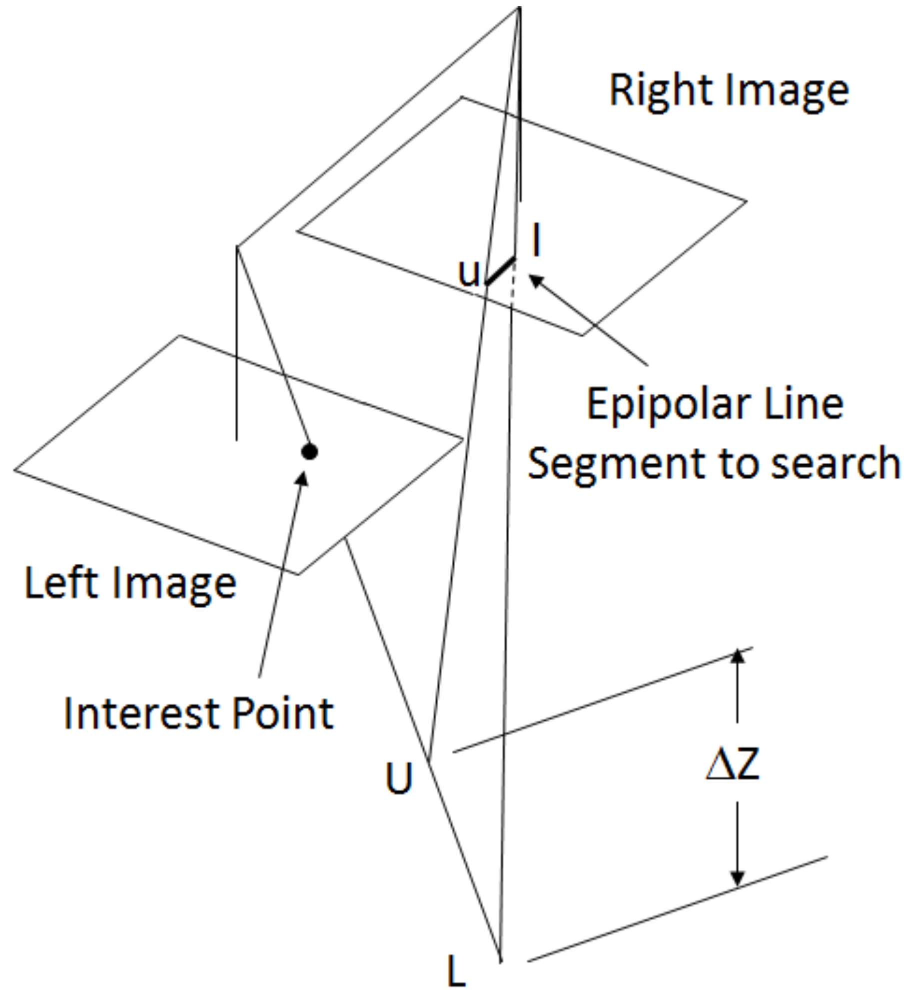
Reducing the Search Space

- Epipolar line constraint
  - Reduces the problem from a 2D to a 1D search—do not need to search the entire image
  - Requires known RO: can be iteratively refined
- Epipolar plane
  - Contains the 2 PCs and the object point
- Epipolar line
  - Line of intersection of epipolar and image planes



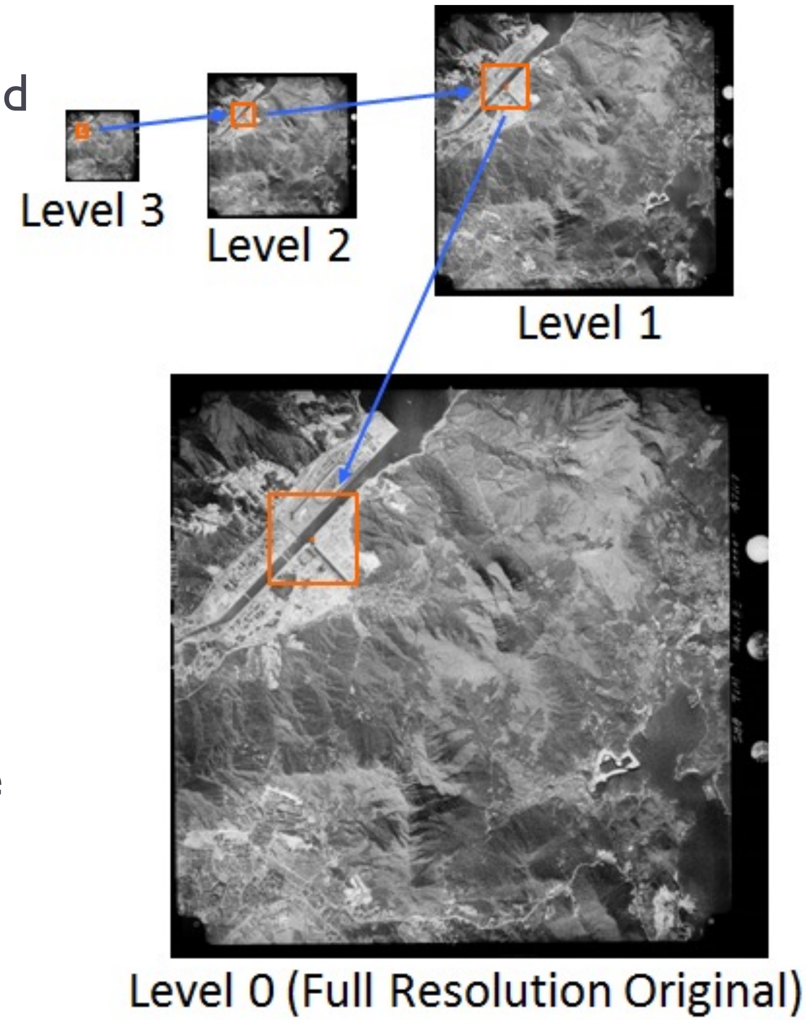
Reducing the Search Space (cont'd)

- Elevation range constraint
  - The linear search space (epiploar line) can be further reduced by approximating the feature height
  - The approximate height range, ΔZ, can be back projected into the image, resulting in a smaller search range on the epipolar line



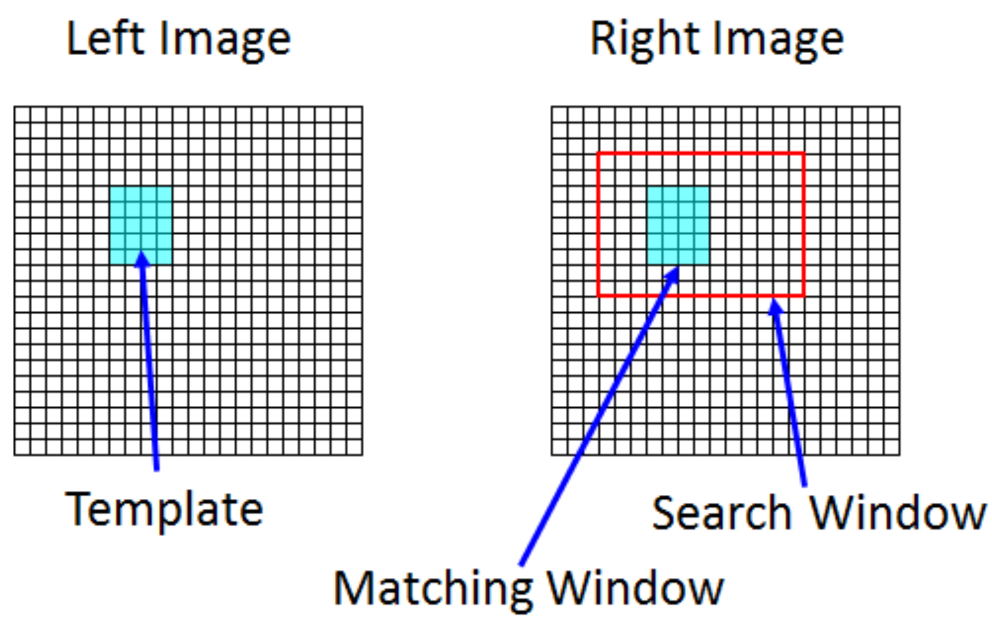
Reducing the Search Space (cont'd)

- Image pyramid
  - The search space can be further reduced by reducing the image resolution in a pyramid structure
  - The search is first performed at the lowest resolution
  - The match location is projected onto the next highest resolution level of the pyramid at the centre of the search window
  - The match location is improved and projected to the next highest level
  - Process repeats until the match is made at the original resolution

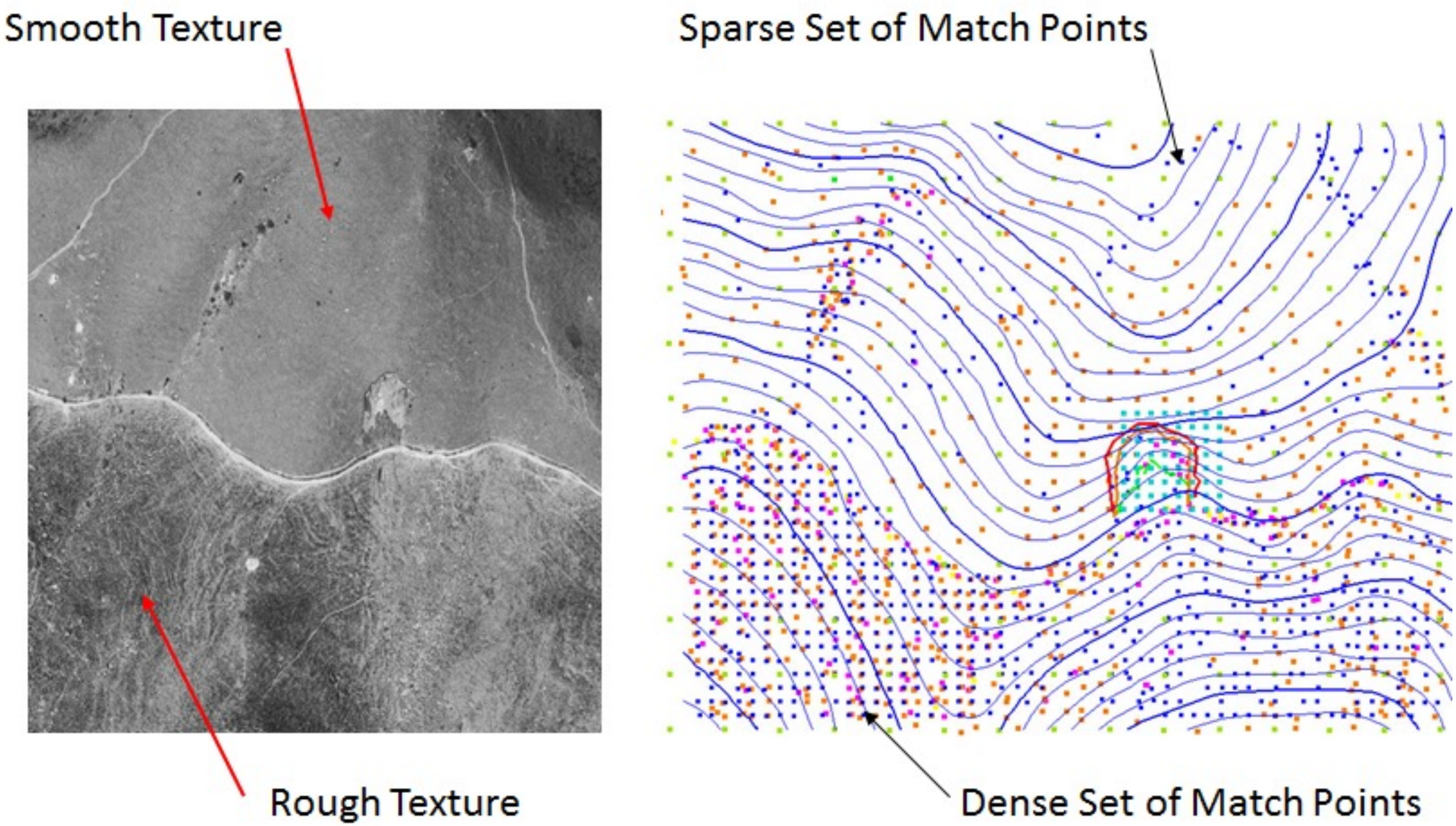


Area-Based Matching

- The matching entity location is determined by comparing the brightness values of a **template** patch (e.g. from the left image) with those of a **search window** in the corresponding image (e.g. the right)
- Essentially, the template is translated through the search window
- Methods of comparison
  - Cross-correlation
  - Least-squares matching
  - Many, many more…



The Influence of Texture



Figures courtesy Andy Hansen

Dense Image Matching

- Sparse image matching:
  - Produces a set of high-quality matches for a stereo pair
  - Used to compute the RO parameters, from which the epipolar lines can be computed
- Dense image matching
  - Produces (lower-quality) matches for every pixel of a stereo pair
  - Used to produce a dense 3D point cloud for a scene

Dense Image Matching (cont'd)

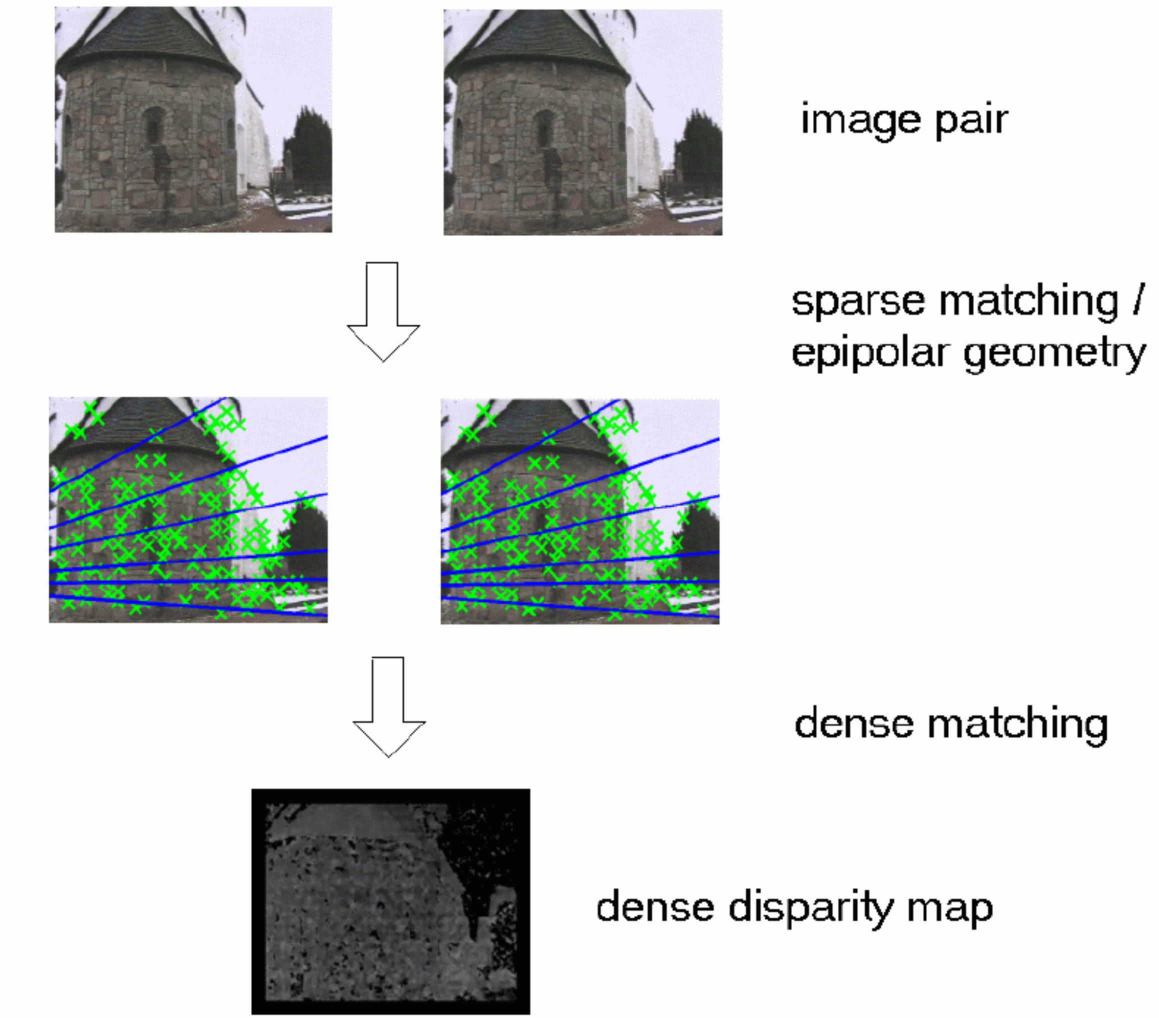


Image source: http://homepages.inf.ed.ac.uk/rbf/CAMERA/RESULTS/YEAR3/node8.html

Dense Image Matching (cont'd)

- Semi-global dense matching from oblique aerial imagery

Image source: http://www.ifp.uni-stuttgart.de/forschung/Image_Processing/Dense_Matching/index.en.html

Summary of Equations

- Coplanarity condition

$$\vec{B} = \begin{pmatrix} b_X & b_Y & b_Z \end{pmatrix}^T$$

$$\vec{v}_{L_i} = \begin{pmatrix} \bar{x}_{L_i} & \bar{y}_{L_i} & -c \end{pmatrix}^T$$

$$\vec{v}_{R_i} = \begin{pmatrix} \bar{x}_{R_i} & \bar{y}_{R_i} & -c \end{pmatrix}^T$$

$$M_R^T \vec{v}_{R_i} = M_R^T \begin{pmatrix} \bar{x}_{R_i} \\ \bar{y}_{R_i} \\ -c \end{pmatrix} = \begin{pmatrix} \bar{x}'_{R_i} \\ \bar{y}'_{R_i} \\ \bar{z}'_{R_i} \end{pmatrix}$$

$$\vec{B} \bullet \left(\vec{v}_L \times M_R^T \vec{v}_R \right) = \begin{vmatrix} b_X & b_Y & b_Z \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ \bar{x}'_{R_i} & \bar{y}'_{R_i} & \bar{z}'_{R_i} \end{vmatrix} = 0$$

- Space intersection

$$X_{L_i}^m = \lambda_i \bar{x}_{L_i} \qquad X_{R_i}^m = \mu_i \bar{x}'_{R_i} + b_X$$

$$Y_{L_i}^m = \lambda_i \bar{y}_{L_i} \qquad Y_{R_i}^m = \mu_i \bar{y}'_{R_i} + b_Y$$

$$Z_{L_i}^m = -\lambda_i c \qquad Z_{R_i}^m = \mu_i \bar{z}'_{R_i} + b_Z$$

$$\lambda_i = \frac{b_X \bar{z}'_{R_i} - b_Z \bar{x}'_{R_i}}{\bar{x}_{L_i} \bar{z}'_{R_i} + c \bar{x}'_{R_i}} \qquad \mu_i = \frac{-b_X c - b_Z \bar{x}_{L_i}}{\bar{x}_{L_i} \bar{z}'_{R_i} + c \bar{x}'_{R_i}}$$

$$\begin{pmatrix} X_i^m \\ Y_i^m \\ Z_i^m \end{pmatrix} = \begin{pmatrix} X_{L_i}^m \\ \dfrac{Y_{L_i}^m + Y_{R_i}^m}{2} \\ Z_{L_i}^m \end{pmatrix} = \begin{pmatrix} X_{R_i}^m \\ \dfrac{Y_{L_i}^m + Y_{R_i}^m}{2} \\ Z_{R_i}^m \end{pmatrix}$$

$$pY_i = Y_{R_i}^m - Y_{L_i}^m$$

Coplanarity Condition Partial Derivatives

- (Dependent RO)
- The partial derivatives can be expressed as determinants

$$\frac{\partial\Delta}{\partial b_Y} = \begin{vmatrix} 0 & 1 & 0 \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ \bar{x}'_{R_i} & \bar{y}'_{R_i} & \bar{z}'_{R_i} \end{vmatrix} \qquad \frac{\partial\Delta}{\partial b_Z} = \begin{vmatrix} 0 & 0 & 1 \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ \bar{x}'_{R_i} & \bar{y}'_{R_i} & \bar{z}'_{R_i} \end{vmatrix}$$

$$\frac{\partial\Delta}{\partial \omega} = \begin{vmatrix} b_X & b_Y & b_Z \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ 0 & -\bar{z}'_{R_i} & \bar{y}'_{R_i} \end{vmatrix} \qquad \frac{\partial\Delta}{\partial \phi} = \begin{vmatrix} b_X & b_Y & b_Z \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ A & B & C \end{vmatrix} \qquad \frac{\partial\Delta}{\partial \kappa} = \begin{vmatrix} b_X & b_Y & b_Z \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ D & E & F \end{vmatrix}$$

$$A = -\bar{y}'_{R_i} \sin\omega^0 + \bar{z}'_{R_i} \cos\omega^0$$

$$B = \bar{x}'_{R_i} \sin\omega^0$$

$$C = -\bar{x}'_{R_i} \cos\omega^0$$

$$D = -\bar{y}'_{R_i} \cos\omega^0 \cos\phi^0 - \bar{z}'_{R_i} \sin\omega^0 \cos\phi^0$$

$$E = \bar{x}'_{R_i} \cos\omega^0 \cos\phi^0 - \bar{z}'_{R_i} \sin\phi^0$$

$$F = \bar{x}'_{R_i} \sin\omega^0 \cos\phi^0 + \bar{y}'_{R_i} \sin\phi^0$$

- Misclosure

$$\Delta_i^0 = f\left(b_Y^0, b_Z^0, \omega^0, \phi^0, \kappa^0, \vec{v}_{L_i}, M_R^T \vec{v}_{R_i}\right)$$

$$= \begin{vmatrix} b_X & b_Y^0 & b_Z^0 \\ \bar{x}_{L_i} & \bar{y}_{L_i} & -c \\ \bar{x}'_{R_i} & \bar{y}'_{R_i} & \bar{z}'_{R_i} \end{vmatrix}$$