3D Lines from Left:Right 2D Line Pairs

Robert B. Fisher
School of Informatics
University of Edinburgh

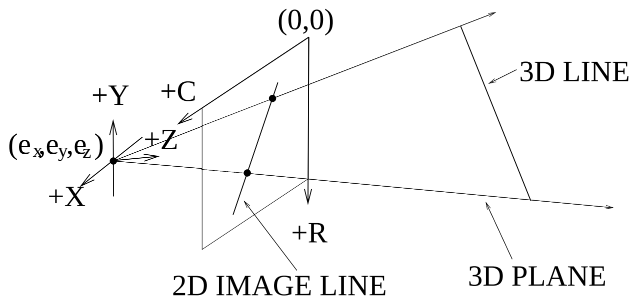
3D Line Calculation

Aim: recovery of 3D line positions

Assume: line successfully matched in L & R images

1. Compute 3D plane that goes through image line and camera origin
2. Compute intersection of 3D planes from 2 cameras (which gives a line)

3D plane passing thru 2D image line

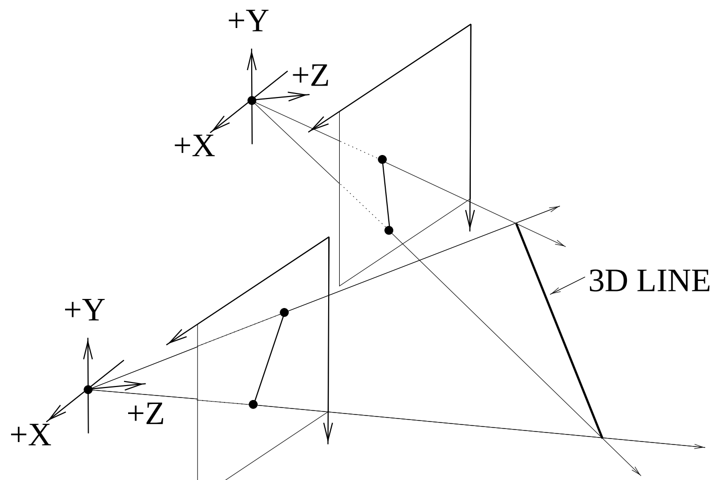


2D image line $l = [a, b, c]'$ is $a * col + b * row + c = 0$

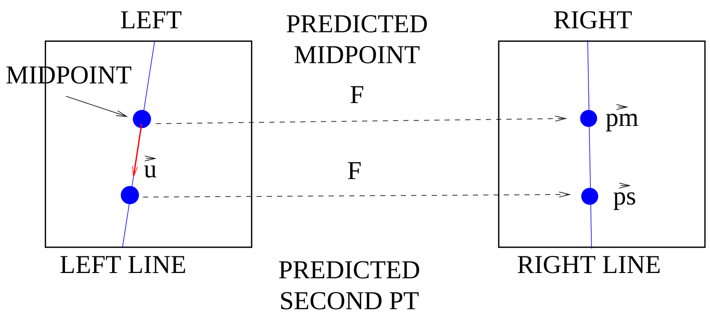
Then plane is l'P

Compute for left and right images

3D Plane Intersection → 3D Line



Computing 3D Line Positions I



Computing paired points on left and right line

Computing 3D Line Positions II

Given: paired lines (l,r) with midpoints $\vec{m}_l = (m_{lx}, m_{ly})$ and $\vec{m}_r$ and directions $\vec{a}_l = (a_{lx}, a_{ly})$ and $\vec{a}_r = (a_{rx}, a_{ry})$
Fundamental matrix $\mathbf{F}$ that maps left to right image

1) Define homogeneous left midpoint: $\vec{c}_l = (m_{lx}, m_{ly}, 1)'$ and line direction $\vec{u} = (a_{lx}, a_{ly}, 0)'$
2) Define projective right line: $\vec{v} = (a_{ry}, -a_{rx}, -(a_{ry}, -a_{rx}) \cdot \vec{m}_r)$
3) Define skew matrix version of projective right line (for algebraic line intersection):

$$\mathrm{M} = \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix}$$

Computing 3D Line Positions III

4) Predict left midpoint position in right image on paired line (so that exact triangulation works): $\vec{p} = \mathrm{M} * \mathbf{F}\vec{c}_l$, $\vec{pm}_r = (p_x/p_z, p_y/p_z)'$
5) Predict second left point on line in right image: $\vec{q} = \mathrm{M} * \mathbf{F}(\vec{c}_l + 10 * \vec{u})$, $\vec{ps}_r = (q_x/q_z, q_y/q_z)'$
6) Triangulate pairs $(\vec{c}_l, \vec{pm}_r)$ and $(\vec{c}_l + 10 * \vec{u}, \vec{ps}_r)$ to get 3D points $\vec{g}$ and $\vec{h}$
7) Compute matched line 3D midpoint $\vec{p}_3 = \vec{g}$ and 3D line direction $\vec{d}_3 = (\vec{h} - \vec{g}) / \mid\mid \vec{h} - \vec{g} \mid\mid$

Triangulating 2 points $(\vec{a}, \vec{b}) \rightarrow \vec{x}$

Given: Left/right projection matrices: $\mathbf{P}_l$, $\mathbf{P}_r$
Left/right intrinsic parameter matrices: $\mathbf{K}_l$, $\mathbf{K}_r$
Left/right matched points: $\vec{a} = (a_x, a_y)'$ and $\vec{b} = (b_x, b_y)'$

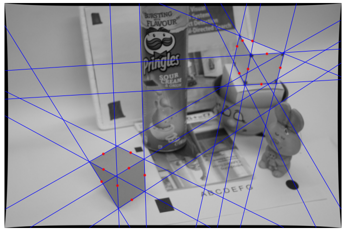
Compute:
$\vec{r} = (\mathbf{K}_l)^{-1}(a_x, a_y, 1)'$ and $\vec{s} = (\mathbf{K}_r)^{-1}(b_x, b_y, 1)'$

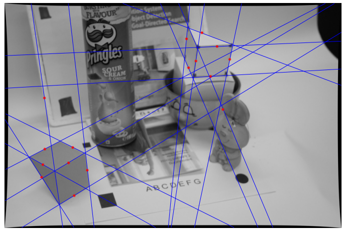
$\vec{a}_1 = r_1 * \mathbf{P}_l(3,:) - \mathbf{P}_l(1,:)$ and $\vec{a}_2 = r_2 * \mathbf{P}_l(3,:) - \mathbf{P}_l(2,:)$
$\vec{a}_3 = s_1 * \mathbf{P}_r(3,:) - \mathbf{P}_r(1,:)$ and $\vec{a}_4 = s_2 * \mathbf{P}_r(3,:) - \mathbf{P}_r(2,:)$

$[USV] = svd(\frac{\vec{a}_1}{||\vec{a}_1||}; \frac{\vec{a}_2}{||\vec{a}_2||}; \frac{\vec{a}_3}{||\vec{a}_3||}; \frac{\vec{a}_4}{||\vec{a}_4||})$

$\vec{x} = V(1:3,4)'/V(4,4)$

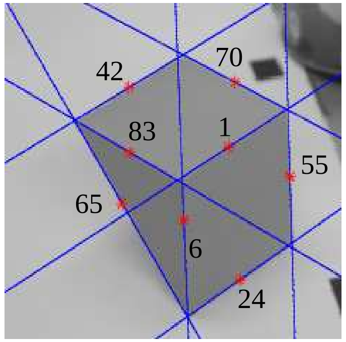
Found Valid Line Pairs

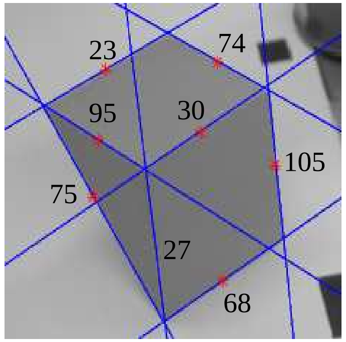




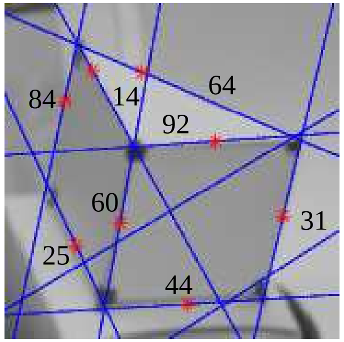
All lines present and all but one midpoint still misplaced

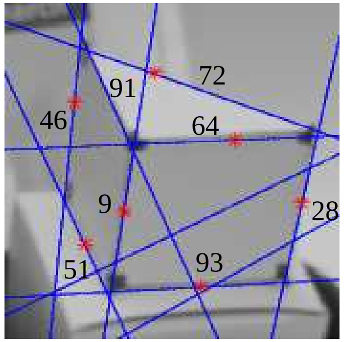
Block 1 2D Line Labels





Block 2 2D Line Labels





Block 1 3D Line Relative Orientations I

Left 1	Left 2	True	Computed
1	6	1.57	1.52
1	24	0.00	0.20
1	42	0.00	0.14
1	55	1.57	1.50
1	65	1.57	1.55
1	70	1.57	1.48
1	83	1.57	1.43
6	24	1.57	1.45
6	42	1.57	1.50
6	55	0.00	0.07

Block 1 3D Line Relative Orientations II

Left 1	Left 2	True	Computed
6	65	0.78	0.84
6	70	1.57	1.44
6	83	1.57	1.52
24	42	0.00	0.06
24	55	1.57	1.49
24	65	1.57	.153
24	70	1.57	1.52
24	83	1.57	1.56
42	55	1.57	1.54
42	65	1.57	1.54

Block 1 3D Line Relative Orientations III

Left 1	Left 2	True	Computed
42	70	1.57	1.56
42	83	1.57	1.52
55	65	0.78	0.78
55	70	1.57	1.51
55	83	1.57	1.56
65	70	0.78	0.86
65	83	0.78	0.78
70	83	0.00	0.09

Clearly reasonably accurate in 3D

What We Have Learned

- Computing 3D line by intersecting backprojected 2D lines
- Backprojection geometric calculations, including triangulation
- Backprojection is reasonably accurate, but not perfect