Massachusetts Institute of Technology

Department of Computer Science and Electrical Engineering

6.801/6.866 Machine Vision Quiz II Solutions

Handed out: 2004 Dec. 16th

PROBLEM 1 (20 points)

Part a (3 points).

The cost is:

$$E = \sum_{i=1}^{n} \|\mathbf{r_{r,i}} - \mathbf{sR}(\mathbf{r_{l,i}}) - \mathbf{r_0}\|^2$$

Differentiate this with respect to $\mathbf{r_0}$ and set it equal to zero:

$$\frac{\partial E}{\partial \mathbf{r_0}} = \sum_{i=1}^{n} -2\Big(\mathbf{r_{r,i}} - \mathbf{sR}(\mathbf{r_{l,i}}) - \mathbf{r_0}\Big) = \mathbf{0} \quad \Longrightarrow$$

$$\sum_{i=1}^{n}\Big(\mathbf{r_{r,i}}\Big) - \sum_{\mathbf{i=1}}^{\mathbf{n}}\Big(\mathbf{sR}(\mathbf{r_{l,i}})\Big) - \sum_{\mathbf{i=1}}^{\mathbf{n}}\Big(\mathbf{r_0}\Big) = \mathbf{0} \quad \Longrightarrow$$

$$\mathbf{r_0} = \underbrace{\frac{1}{n}\sum_{i=1}^{n}\Big(\mathbf{r_{r,i}}\Big)}_{\mathbf{\overline{r}_{r,i}}} - \mathbf{sR}\Big(\underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathbf{r_{l,i}}}_{\mathbf{\overline{r}_l}}\Big)$$

Part b (3 points).

Thus, we have

$$E = \sum_{i=1}^{n} \|\mathbf{r_{r,i}} - \mathbf{\overline{r}_r} - \mathbf{sR}(\mathbf{r_{l,i}}) - \mathbf{sR\overline{r}_r}\|^2$$

$$= \sum_{i=1}^{n} \|\underbrace{\mathbf{r_{r,i}} - \mathbf{\overline{r}_{r,i}}}_{\mathbf{r'_{r,i}}} - sR(\underbrace{\mathbf{r_{l,i}} - \mathbf{\overline{r}_r}}_{\mathbf{r'_{l,i}}})\|^2$$

Part c (4 points).

We know that $R^T R = 1$. Thus, the sum of squares of errors is

$$\begin{aligned} E &= \sum_{i=1}^{n} \left\| \mathbf{r'_{r,i}} - \mathbf{sRr'_{l,i}} \right\|^2 \\ &= \sum_{i=1}^{n} \left((\mathbf{r'_{r,i}})^{\mathbf{T}} (\mathbf{r'_{r,i}}) - \mathbf{2s} (\mathbf{r'_{r,i}})^{\mathbf{T}} \mathbf{R} (\mathbf{r'_{l,i}}) + \mathbf{s^2} (\mathbf{r'_{l,i}})^{\mathbf{T}} \mathbf{R^T R} (\mathbf{r'_{l,i}}) \right) \\ &= \underbrace{\sum_{i=1}^{n} \left((\mathbf{r'_{r,i}})^{\mathbf{T}} (\mathbf{r'_{r,i}}) \right)}_{S_r} - 2s \underbrace{\sum_{i=1}^{n} \left((\mathbf{r'_{r,i}})^{\mathbf{T}} \mathbf{R} (\mathbf{r'_{l,i}}) \right)}_{D_{rl}} + s^2 \underbrace{\sum_{i=1}^{n} \left((\mathbf{r'_{l,i}})^{\mathbf{T}} (\mathbf{r'_{l,i}}) \right)}_{S_l} \end{aligned}$$

Since $S_l > 0$, E will reach its minimum when

$$s = \sqrt{\frac{D_{rl}}{S_l}}$$

Part d (3 points).

Repeating the similar procedure, we will have

$$\begin{aligned} E &= \sum_{i=1}^{n} \|\mathbf{r'_{l,i}} - \mathbf{s'R'}(\mathbf{r'_{r,i}})\|^{\mathbf{2}} \\ &= \sum_{i=1}^{n} \|\mathbf{r'_{l,i}} - \mathbf{s'R'}(\mathbf{r'_{r,i}})\|^{\mathbf{2}} \\ &= \underbrace{\sum_{i=1}^{n} \left((\mathbf{r'_{l,i}})^{\mathbf{T}} (\mathbf{r'_{l,i}}) \right)}_{S_l} - 2s' \underbrace{\sum_{i=1}^{n} \left((\mathbf{r'_{l,i}})^{\mathbf{T}} \mathbf{R} (\mathbf{r'_{r,i}}) \right)}_{D_{lr}} + (s')^2 \underbrace{\sum_{i=1}^{n} \left((\mathbf{r'_{r,i}})^{\mathbf{T}} (\mathbf{r'_{r,i}}) \right)}_{S_r} \end{aligned}$$

Since $S_r > 0$, E will reach its minimum when

$$s' = \sqrt{\frac{D_{lr}}{S_r}}$$

Typically

$$ss' = \sqrt{\frac{D_{lr}D_{rl}}{S_r S_l}} \neq 1$$

Part e/f (4 points).

If we change definition as followed:

$$\begin{aligned} E &= \sum_{i=1}^{n} \|\frac{1}{\sqrt{s}}\mathbf{r'_{r,i}} - \sqrt{\mathbf{s}}\mathbf{R}(\mathbf{r'_{l,i}})\|^{\mathbf{2}} \\ &= \frac{1}{s}\underbrace{\sum_{i=1}^{n}\left((\mathbf{r'_{r,i}})^{\mathbf{T}}(\mathbf{r'_{r,i}})\right)}_{S_r} - 2\underbrace{\sum_{i=1}^{n}\left((\mathbf{r'_{r,i}})^{\mathbf{T}}\mathbf{R}(\mathbf{r'_{l,i}})\right)}_{D_{rl}} + s\underbrace{\sum_{i=1}^{n}\left((\mathbf{r'_{l,i}})^{\mathbf{T}}(\mathbf{r'_{l,i}})\right)}_{S_l} \\ &\geq 2\sqrt{S_r S_l} - 2D_{rl} \end{aligned}$$

which reaches minimum when

$$s = \sqrt{\frac{S_r}{S_l}}.$$

Similarly we can find the inverse transformation mentioned in part (d).

$$s' = \sqrt{\frac{S_l}{S_r}}.$$

Obviously, the result does not depend on rotation or translation.
We also notice that

$$s * s' = 1$$

i.e., the asymmetry is removed.

Part g (3 points).

To choose best rotation R to minimize the cost functions for part (b), (d), and (e), we only need to maximize

$D_{rl} = (\mathbf{r'_{r,i}})^{\mathbf{T}}\mathbf{R}(\mathbf{r'_{l,i}})$ or $D_{lr} = (\mathbf{r'_{l,i}})^{\mathbf{T}}\mathbf{R'}(\mathbf{r'_{r,i}})$

The selection for rotation matrix R (from left to right) (or R': from left to right) does not depend on the selection of scalar s.

Actually two rotation matrix R and R' than respectively maximize the two above expressions have the following relationship:

$$R' = R^T = R^{-1}$$

i.e., there also exists symmetry for rotation matrix for all three situations.

PROBLEM 2 (15 points)

For each of the following parts, we expect the following important steps. We might ignore many complementary details.

Part a (4 points).

(1) Edge finding to find the edges of the triangular areas — away from the vertices.
(2) Then least squares fitting of straight lines to the edge fragments (ignoring parts near vertices).
(3) Then intersecting the lines to find the vertices (more least squares, since three lines don't intersect in a point generally).

Part b (3 points).

The edges of the triangles are (relatively) independent of distance from camera to target. On the other hand, lines instead of areas have the following disadvantages:
(1) It will not work well from far away when they may be "washed out" because their width is smaller than a pixel,
(2) It will not work well close up when the width of the lines may cover many pixels.
Further, accurate focus is needed to make sure the lines are visible (less of a problem with areas — the borders just have gradual transitions).

Part c (4 points).

Dots are even worse than lines because of the following concerns:
(1) They will disappear when viewed from far enough away (partial pixel effect).
(2) From close up they will cover a large disc in the image (actually, this is not so bad, since we saw that the center of the disc can be found accurately). Same focus issues as (b).

Part d (4 points).

LEDs have similar disadvantages as dots.
(1) From too far away they are invisible. (LEDs can be seen better than black dots.)

(2) From closer up they will saturate the sensor, making it hard to find the exact position of the center.

Further, some sensors will be permanently damaged when exposed to accurately focused.

By the way, additional advantages of this pattern shown are that not only can it be used over a wide range of distances, but also over quite a range of inclinations.

PROBLEM 3 (30 points)

Part a (6 points).

Assume 4 vertices are represented by E, F, G, H.

$$\begin{aligned} E &= (-a, b, 0)^T \\ F &= (-a, -b, 0)^T \\ G &= (a, 0, b)^T \\ H &= (a, 0, -b)^T \end{aligned}$$

The length for all sides are:

$$\begin{aligned} \|EF\| = \|GH\| &= 2b \\ \|FG\| = \|EH\| &= \sqrt{4a^2 + 2b^2} \end{aligned}$$

To construct a regular tetrahedron, we should have:

$$\begin{aligned} \|EF\| = \|GH\| = \|FG\| = \|EH\| \quad &\Longrightarrow \\ 2b^2 = 4a^2 \quad &\Longrightarrow \\ \frac{b}{a} &= \pm\sqrt{2} \end{aligned}$$

Without losing generality we can assume that $\frac{b}{a} = -\sqrt{2}$. Thus, we have:

$$\begin{aligned} E &= -a(1, \sqrt{2}, 0)^T \\ F &= -a(1, -\sqrt{2}, 0)^T \end{aligned}$$

$$\begin{aligned} G &= -a(-1, 0, \sqrt{2})^T \\ H &= -a(-1, 0, -\sqrt{2})^T \end{aligned}$$

1) There will be 4 rotating axis each passing 1 vertice (with 120°, 240° degree rotation) which can bring the tetrahedron back into alignment with itself.

l_1: pass vertice E in the direction of $E - \frac{1}{3}(F + G + H)$, i.e. $-4a/3(1, \sqrt{2}, 0)^T$
(Or in the direction of $GF \times GH$).

$$GF \times GH = a^2(2, -\sqrt{2}, -\sqrt{2})^T \times (0, 0, -2\sqrt{2})^T = 4a^2(1, \sqrt{2}, 0)^T.$$

l_2: pass vertice F in the direction of $F - \frac{1}{3}(E + G + H)$, i.e., $-4a/3(1, -\sqrt{2}, 0)^T$
(Or in the direction of $GH \times GE$).

$$GH \times GE = a^2(0, 0, -2\sqrt{2})^T \times (2, \sqrt{2}, -\sqrt{2})^T = 4a^2(1, -\sqrt{2}, 0)^T.$$

l_3: pass vertice G in the direction of $G - \frac{1}{3}(E + F + H)$, i.e., $-4a/3(1, 0, -\sqrt{2})^T$
(Or in the direction of $FH \times FE$).

$$FH \times FE = a^2(-2, \sqrt{2}, -\sqrt{2})^T \times (0, 2\sqrt{2}, 0)^T = 4(1, 0, -\sqrt{2})^T.$$

l_4: pass vertice H in the direction of $H - \frac{1}{3}(E + F + G)$. i.e., $-4a/3(1, 0, \sqrt{2})^T$
(Or in the direction of $EG \times EF$).

$$EG \times EF = a^2(-2, -\sqrt{2}, \sqrt{2})^T \times (0, -2\sqrt{2}, 0)^T = 4(1, 0, \sqrt{2})^T.$$

For simplicity, let

$$\begin{aligned} l_1 &= (1, \sqrt{2}, 0)^T \\ l_2 &= (1, -\sqrt{2}, 0)^T \\ l_3 &= (1, 0, -\sqrt{2})^T \\ l_4 &= (1, 0, \sqrt{2})^T \end{aligned}$$

Note:

For this specific example, $E+F+G+H = (0, 0, 0)$, therefore the center of the tetrahedron is at the origin and we can find the rotation axis directly from four vertices. i.e., $l_1 = E$, $l_2 = F$, $l_3 = G$ and $l_4 = H$. However it is not the general case. (The solution we offered applies in more general situations.) Students who use the vertices coordinates as rotating axis without explanation do not receive full credit.

2) There will be 3 rotating axis each passing the midpoints of two opposite edges (with 180° degree rotation) which can bring the tetrahedron back into alignment with itself.

l_5: pass the midpoint of edge EF $[-a(1,0,0)^T]$ and the midpoint of GH $[-a(-1,0,0)^T]$ (i.e., in the direction of $\frac{E+F}{2} - \frac{G+H}{2}$).

l_6: pass the midpoint of edge EH $[-a(0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})^T]$ and the midpoint of FG $[-a(0, -\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T]$ (i.e., in the direction of $\frac{E+H}{2} - \frac{G+F}{2}$).

l_7: pass the midpoint of edge EG $[-a(0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T]$ and the midpoint of FH $[-a(0, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})^T]$ (i.e., in the direction of $\frac{E+G}{2} - \frac{F+H}{2}$).

i.e.,

$$\begin{aligned} l_5 &= (1,0,0)^T \\ l_6 &= (0,1,-1)^T \\ l_7 &= (0,1,1)^T \end{aligned}$$

In summary, there are 12 rotations:

$(1,0,0,0)$	the identity
$\left(\frac{1}{2}, \frac{1}{2}, 0, \frac{1}{\sqrt{2}}\right)$	120° rotation around $(1,0,\sqrt{2})$ axis
$\left(-\frac{1}{2}, \frac{1}{2}, 0, \frac{1}{\sqrt{2}}\right)$	240° rotation around $(1,0,\sqrt{2})$ axis
$\left(\frac{1}{2}, \frac{1}{2}, 0, -\frac{1}{\sqrt{2}}\right)$	120° rotation around $(1,0,-\sqrt{2})$ axis
$\left(-\frac{1}{2}, \frac{1}{2}, 0, -\frac{1}{\sqrt{2}}\right)$	240° rotation around $(1,0,-\sqrt{2})$ axis
$\left(\frac{1}{2}, \frac{1}{2}, \frac{1}{\sqrt{2}}, 0\right)$	120° rotation around $(1,\sqrt{2},0)$ axis
$\left(-\frac{1}{2}, \frac{1}{2}, \frac{1}{\sqrt{2}}, 0\right)$	240° rotation around $(1,\sqrt{2},0)$ axis
$\left(\frac{1}{2}, \frac{1}{2}, -\frac{1}{\sqrt{2}}, 0\right)$	120° rotation around $(1,-\sqrt{2},0)$ axis
$\left(-\frac{1}{2}, \frac{1}{2}, -\frac{1}{\sqrt{2}}, 0\right)$	240° rotation around $(1,-\sqrt{2},0)$ axis
$\left(0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$	180° rotation around $(0,1,1)$ axis
$\left(0, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right)$	180° rotation around $(0,1,-1)$ axis
$(0,1,0,0)$	180° rotation around $(1,0,0)$ axis

We could also write simple matlab code to cross-multiply all the resulting quaternions and stop if no new ones are produced. However, we should aware that quaternions q, q_x, q_y, q_z and $-q, -q_x, -q_y, -q_z$ represent the same rotations.

Part b (6 points).

Consider the orientation tetrahedron described in part(a) the standard orientation. We define $\overset{\circ}{q}$ to be the set of quaternions that bring it back to alignment with itself.

Quaternion $\overset{\circ}{p}$ describes the transformation that rotates a tetrahedron from the *standard* orientation (T) to an *arbitrary* alignment (T').

Quaternion $\overset{\circ}{p}*$ describes the inverse transformation that rotates the tetrahedron with the mentioned *arbitrary* alignment (T') to the *standard* orientation (T).

Then, for the arbitrary tetrahedron, the quaternion $\overset{\circ}{p}\overset{\circ}{q}\overset{\circ}{p}*$ describes the following composed rotation:

1) It first rotates (described by $\overset{\circ}{p}*$) the *arbitrary* tetrahedron (T') with other alignment to the *standard* orientation (T).

2) Next, it rotates (described by $\overset{\circ}{q}$) the tetrahedron from its *standard* orientation (T) and brings it back to alignment with *itself* (T).

3) Finally, it rotates (described by $\overset{\circ}{p}$) the tetrahedron from its *standard* orientation to the initial arbitrary orientation (T').

Thus, quaternion $\overset{\circ}{p}\overset{\circ}{q}\overset{\circ}{p}*$ describes the transformation that rotate the arbitrary tetrahedron with other alignment back to the alignment with itself. And the set of quaternions we find are all we need because of one-to-one relationship we present here.

Part c (6 points).

The smallest angles of rotation that brings platonic solids back to alignment with themselves are:

Tetrahedron :	120°
Hexahedron :	90°
Octahedron :	90°
Dodecahedron :	72°
Icosahedron :	72°

The smallest rotation is 72° for dodecahedron and icosahedron.

Part d (6 points).

$$\begin{aligned}
(\overset{\circ}{a}\overset{\circ}{q})\cdot\overset{\circ}{b} &= (aq-\mathbf{q}\cdot\mathbf{a}, a\mathbf{q}+q\mathbf{a}+\mathbf{a}\times\mathbf{q})\cdot\overset{\circ}{b}\\
&= (baq-b\mathbf{q}\cdot\mathbf{a})+(a\mathbf{q}\cdot\mathbf{b}+q\mathbf{a}\cdot\mathbf{b})+(\mathbf{a}\times\mathbf{q})\cdot\mathbf{b}\\
\overset{\circ}{a}\cdot(\overset{\circ}{b}\overset{\circ}{q}*) &= \overset{\circ}{a}\cdot(bq+\mathbf{b}\cdot\mathbf{q}, -b\mathbf{q}+q\mathbf{b}-\mathbf{b}\times\mathbf{q})\\
&= (abq+a\mathbf{b}\cdot\mathbf{q})+(-b\mathbf{a}\cdot\mathbf{q}+q\mathbf{a}\cdot\mathbf{b})-\mathbf{a}\cdot(\mathbf{b}\times\mathbf{q})
\end{aligned}$$

Since

$$(\mathbf{a}\times\mathbf{q})\cdot\mathbf{b}=\mathbf{a}\cdot(\mathbf{q}\times\mathbf{b})=-\mathbf{a}\cdot(\mathbf{b}\times\mathbf{q})$$

We show that

$$(\overset{\circ}{a}\overset{\circ}{q})\cdot\overset{\circ}{b}=\overset{\circ}{a}\cdot(\overset{\circ}{b}\overset{\circ}{q}*)$$

Using the conclusion, we have

$$(\overset{\circ}{a}\overset{\circ}{b})\cdot(\overset{\circ}{a}\overset{\circ}{b}) = \overset{\circ}{a}\cdot(\overset{\circ}{a}\overset{\circ}{b}\overset{\circ}{b}*)=\overset{\circ}{a}\cdot(\overset{\circ}{a}\,(\overset{\circ}{b}\cdot\overset{\circ}{b}))=(\overset{\circ}{a}\cdot\overset{\circ}{a})(\overset{\circ}{b}\cdot\overset{\circ}{b})$$

If $\overset{\circ}{r}=(0,\mathbf{r})$ and $\overset{\circ}{s}=(0,\mathbf{s})$, we have:

$$\begin{aligned}
\overset{\circ}{r}\overset{\circ}{s} &= (0-\mathbf{r}\cdot\mathbf{s}, 0\mathbf{s}+0\mathbf{r}+\mathbf{r}\times\mathbf{s})=(-\mathbf{r}\cdot\mathbf{s},\mathbf{r}\times\mathbf{s})\\
(\overset{\circ}{r}\overset{\circ}{s})\cdot\overset{\circ}{t} &= (-\mathbf{r}\cdot\mathbf{s},\mathbf{r}\times\mathbf{s})\cdot\overset{\circ}{t}=(\mathbf{r}\times\mathbf{s})\cdot\mathbf{t}=[\mathbf{rst}]
\end{aligned}$$

Part e (6 points).

For $\overset{\circ}{r}=(0,\mathbf{r})$

$$\begin{aligned}
\overset{\circ}{r'} &= \overset{\circ}{q}\overset{\circ}{r}\overset{\circ}{q}*=\overset{\circ}{q}\,(\mathbf{r}\cdot\mathbf{q}, q\mathbf{r}-\mathbf{r}\times\mathbf{q})\\
&= (q\mathbf{r}\cdot\mathbf{q}-q\mathbf{q}\cdot\mathbf{r}+\mathbf{q}\cdot(\mathbf{r}\times\mathbf{q}), q(q\mathbf{r}-\mathbf{r}\times\mathbf{q})+(\mathbf{r}\cdot\mathbf{q})\mathbf{q}+\mathbf{q}\times(q\mathbf{r}-\mathbf{r}\times\mathbf{q})\\
&= (0, q(q\mathbf{r}-\mathbf{r}\times\mathbf{q})+(\mathbf{r}\cdot\mathbf{q})\mathbf{q}+\mathbf{q}\times(q\mathbf{r}-\mathbf{r}\times\mathbf{q})
\end{aligned}$$

Since

$$\mathbf{q} \times (\mathbf{r} \times \mathbf{q}) = \mathbf{r}(\mathbf{q} \cdot \mathbf{q}) - \mathbf{q}(\mathbf{q} \cdot \mathbf{r})$$
$$\mathbf{q} \times (\mathbf{q} \times \mathbf{r}) = -\mathbf{r}(\mathbf{q} \cdot \mathbf{q}) + \mathbf{q}(\mathbf{q} \cdot \mathbf{r})$$

We have

$$\mathbf{r}' = \underbrace{(q^2 - \mathbf{q} \cdot \mathbf{q})\mathbf{r}}_{\mathbf{r}_1} + \underbrace{2q(\mathbf{q} \times \mathbf{r})}_{\mathbf{r}_2} + \underbrace{2(\mathbf{r} \cdot \mathbf{q})\mathbf{q}}_{\mathbf{r}_3}$$

and

$$\begin{aligned}\mathbf{r}' &= (q^2 - \mathbf{q} \times \mathbf{q})\mathbf{r} + 2q(\mathbf{q} \times \mathbf{r}) + 2(\mathbf{r} \cdot \mathbf{q})\mathbf{q} + 2\mathbf{q} \times (\mathbf{q} \times \mathbf{r}) + 2\Big(\mathbf{r}(\mathbf{q} \cdot \mathbf{q}) - \mathbf{q}(\mathbf{q} \cdot \mathbf{r})\Big) \\ &= \underbrace{(q^2 + \mathbf{q} \cdot \mathbf{q})\mathbf{r}}_{\mathbf{r}_4} + \underbrace{2q(\mathbf{q} \times \mathbf{r})}_{\mathbf{r}_2} + \underbrace{2\mathbf{q} \times (\mathbf{r} \times \mathbf{q})}_{\mathbf{r}_5}\Big)\end{aligned}$$

Now we compare the computational loads for two equations when $\overset{\circ}{q}$ is a unit quaternion. The computation load for typical operations are listed below.

Operation	Computational load
dot product	**3 muls, 2 adds**
cross product	**6 muls, 3 adds**
scalar multiplication of a vector	**3 muls, 0 add**
scalar multiplication	**1 muls, 0 add**

For the first expression:

1. Computational load for $(q^2 - \mathbf{q} \cdot \mathbf{q})$ and for $\mathbf{r}_1$ is (**4 muls, 3 adds**) and (**7 muls, 3 adds**).

2. Computational load for $(\mathbf{q} \times \mathbf{r})$ and for $\mathbf{r}_2$ is (**6 muls, 3 adds**) and (**10 muls, 3 adds**).

3. Computational load for $2(\mathbf{r} \cdot \mathbf{q})$ and for $\mathbf{r}_3$ is (**4 muls, 2 adds**) and (**7 muls, 2 adds**).

Total: (**24 muls, 14 adds**)

For the second expression, we notice that $\mathbf{r}_4 = \mathbf{r}$ and we can reuse the result of $(\mathbf{q} \times \mathbf{r})$.

1. Computational load for $\mathbf{r}_4$ is 0, which save us the computational load for $\mathbf{r}_1$, i.e., (**7 muls, 3 adds**).

2. Computational load for $2\mathbf{q}$ and for $\mathbf{r}_5$ is (**3 muls**) and (**9 muls, 3 adds**), which needs extra (**2muls**, **1adds**) than for $\mathbf{r}_3$.

Total: (**19 muls, 12 adds**)

In total, the computational load for the second expression saves us **5 muls, 2 adds**.

PROBLEM 4a (7 points)

For simplicity, we can assume we are given (x_c, y_c, z_c) and (x_I, y_I).

Since

$$\begin{aligned} \frac{x_{pi} - x_o}{f} &= \frac{x_{ci}}{z_{ci}} \\ \frac{y_{pi} - y_o}{f} &= \frac{y_{ci}}{z_{ci}} \end{aligned}$$

we have,

$$\begin{aligned} x_{pi} &= x_o + f\frac{x_{ci}}{z_{ci}} \\ y_{pi} &= y_o + f\frac{y_{ci}}{z_{ci}} \end{aligned}$$

and

$$E = \sum_{i=1}^{N} (x_{Ii} - x_o - f\frac{x_{ci}}{z_{ci}})^2 + (y_{Ii} - y_o - f\frac{y_{ci}}{z_{ci}})^2$$

We need to find (x_{ci}, y_{ci}, z_{ci}) which minimize E.

We notice that the choices

$$\begin{aligned} x_{ci} &= \frac{k_i}{f}(x_{Ii} - x_o) \\ y_{ci} &= \frac{k_i}{f}(y_{Ii} - y_o) \\ z_{ci} &= k_i \end{aligned}$$

can make E to be zero. Because there are three degree of freedom in choosing 3D points while there are only two constraints, we will have infinite number of solutions. It means that we could not uniquely decided the shape of 3D object if we are given only one image. The result does not change for $N = 1$ or $N > 1$.

PROBLEM 4b (8 points)

This problem was phrased in a slightly tricky way: The answer is that the 2D-version of the relative orientation problem is unsolvable, regardless of how many correspondences you have. In other words, for any number n of points $x_1, ..., x_n$ seen by the left camera (i.e. $p_1, ..., p_n$ are points on the "image line" of the left camera, a 2D-equivalent to the image plane of a regular camera. These points are the images of the n points on the plane that are in the field of view of this camera), if the right camera sees the same n points as points $p'_1, ..., p'_n$ on *its* image line, then the two cameras can lie on the plane almost anywhere in relation to one another.

Here is why we say they can lie *almost* anywhere: Points $p_1, ..., p_n$ define n rays coming out of the left camera, and points $p'_1, ...p'_n$ define n rays coming out of the right camera. The only constraint on the relative position of the two cameras is that for each k, the k-th rays of the two cameras must intersect. This is not much of a constraint ... On the figure 1 below we mark these intersections with circles. You can think of them as rings binding the two rays together: Imagine the two set of rays as two set of infinite straight wires fixed to their image lines: The rings binding each two corresponding rays together leave a lot of freedom for the two cameras.

Some intuition for why the problem cannot be solved for any n correspondences can be also gained from the algebra: For each point seen by the two cameras, its positions (x_i, y_i) and (x'_i, y'_i) in the coordinate systems of, correspondingly, the left and the right camera, are bound by equation

$$[x'_i, y'_i]^T = R * [x_i, y_i]^T + T$$

where R is a rotation matrix (determined by one variable, the rotation angle α), and T is the translation vector (two variables t_x, t_y). This equation gives us two constraints per each point: $x'_i = x_i \cos\alpha + y_i \sin\alpha + t_x$ and $y'_i = -x_i \sin\alpha + y_i \cos\alpha + t_y$. Furthermore, the *four* unknowns x_i, y_i, x'_i, y'_i are bound by the equations $p_i/f = x_i/y_i$ and $p'_i/f' = x'_i/y'_i$, where f and f' are principal distances of the two cameras. Therefore, we get the total of four constraints for each point visible by the two cameras (i.e. per each "correspondence"). However, each point gives us also, as we pointed out, *four* unknowns. Hence, no amount of points can give us enough constraints to figure out the three main unknowns we want to determine: α, t_x, t_y.

PROBLEM 5 (20 points)

The point of this problem was to show that (almost) each movement of a camera in relation to some plane can be interpreted in two ways. Let's say the real movement of the camera is defined by $(\mathbf{n}, \mathbf{t}, \omega, h) = (\mathbf{b}, \mathbf{a}, \mathbf{c}, \mathbf{R}_0\mathbf{n})$ where h is a constant such that $\mathbf{R} \cdot \mathbf{n} = h$ for all $\mathbf{R}$ (we know that's constant because all the camera sees is some plane with normal

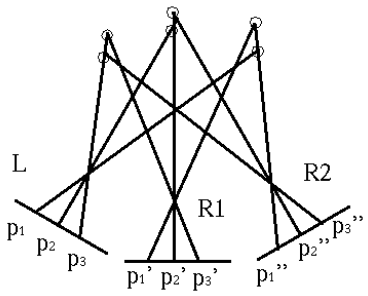

Figure 1: The freedom of relative orientation in 2D (Prob.2)

$\mathbf{n}$). It turns out that there exists k, h' s.t. movement $(\mathbf{b}, \mathbf{a}, \mathbf{c}, \mathbf{R}_0\mathbf{n})$ and movement $(\mathbf{a}, \mathbf{b}, \mathbf{c} + k\mathbf{a} \times \mathbf{b}, h')$ produce the same motion field $\mathbf{r}'$ (note that notion $\mathbf{r}'$ is a shortcut for a function which defines a motion $\mathbf{r}'$ for each point $\mathbf{r}$ on the image plane). In other words, whatever your real movement $(\mathbf{n}, \mathbf{t}, \omega, h) = (\mathbf{b}, \mathbf{a}, \mathbf{c}, \mathbf{R}_0\mathbf{n})$ is, you can (but only for a split second...) believe that your movement is rather described by $(\mathbf{n}, \mathbf{t}, \omega, h) = (\mathbf{a}, \mathbf{b}, \mathbf{c} + k\mathbf{a} \times \mathbf{b}, h')$ where k and h' are variables that depend on $\mathbf{b}, \mathbf{a}, \mathbf{R}_0$ in a way that we will show below.

We will proceed as follows: First we transform the equation for the motion field $\mathbf{r}'$ so that it depends only on the "movement descriptors" $(\mathbf{n}, \mathbf{t}, \omega, h)$ and the pixel position $\mathbf{r}$. To do that, we first find $\mathbf{r}'$ by differentiating

$$(1/f)\mathbf{r} = \frac{\mathbf{R}}{\mathbf{R} \cdot \hat{\mathbf{z}}}$$

and so we get

$$(1/f)\mathbf{r}' = \frac{\mathbf{R}'}{\mathbf{R} \cdot \hat{\mathbf{z}}} - \frac{(\mathbf{R}' \cdot \hat{\mathbf{z}})\mathbf{R}}{(\mathbf{R} \cdot \hat{\mathbf{z}})^2} = \frac{(\mathbf{R} \cdot \hat{\mathbf{z}})\mathbf{R}' - (\mathbf{R}' \cdot \hat{\mathbf{z}})\mathbf{R}}{(\mathbf{R} \cdot \hat{\mathbf{z}})^2} = \frac{\hat{\mathbf{z}} \times (\mathbf{R} \times (-\mathbf{R}'))}{(\mathbf{R} \cdot \hat{\mathbf{z}})^2}$$

This last form is the simplest to work with: It shows that $\mathbf{r}'$ must be perpendicular to $\hat{\mathbf{z}}$, which it how it should be, because any change in the image position of a certain point in the image (that's what the motion field $\mathbf{r}'$ describes) must be within the image plane, and so it is always perpendicular to the camera axis $\hat{\mathbf{z}}$.

We get rid of $-\mathbf{R}'$ by subsituting $-\mathbf{R}' = \mathbf{t} + \omega \times \mathbf{R}$:

$$\mathbf{r}' = f\frac{\hat{\mathbf{z}} \times \Big[\mathbf{R} \times (\mathbf{t} + \omega \times \mathbf{R})\Big]}{(\mathbf{R} \cdot \hat{\mathbf{z}})^2}$$

Now we use the fact that $\frac{\mathbf{r}}{f} = \frac{\mathbf{R}}{\mathbf{R} \cdot \hat{\mathbf{z}}}$, and hence $\mathbf{R} = \frac{\mathbf{R} \cdot \hat{\mathbf{z}}}{f}\mathbf{r}$, to replace the occurences of $\mathbf{R}$ with formulas involving only $\mathbf{r}$ and $(\mathbf{R} \cdot \hat{\mathbf{z}})$:

$$f\mathbf{r}' \quad = \quad f^2\frac{1}{(\mathbf{R} \cdot \hat{\mathbf{z}})^2}\left(\hat{\mathbf{z}} \times \Big[\frac{\mathbf{R} \cdot \hat{\mathbf{z}}}{f}\mathbf{r} \times (\mathbf{t} + \frac{\mathbf{R} \cdot \hat{\mathbf{z}}}{f}\omega \times \mathbf{r})\Big]\right)$$

$$= \quad \hat{\mathbf{z}} \times \left[\mathbf{r} \times \left(\frac{f}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{t} + \omega \times \mathbf{r}\right)\right]$$

Finally we get rid of $\mathbf{R}$ in the $\mathbf{R}\cdot\hat{\mathbf{z}}$ term using the fact that

$$(\mathbf{r}\cdot\mathbf{n}) = f(\frac{\mathbf{R}}{\mathbf{R}\cdot\hat{\mathbf{z}}})\cdot\mathbf{n} = f\frac{\mathbf{R}\cdot\mathbf{n}}{\mathbf{R}\cdot\hat{\mathbf{z}}}$$

and therefore

$$\frac{f}{\mathbf{R}\cdot\hat{\mathbf{z}}} = \frac{\mathbf{r}\cdot\mathbf{n}}{\mathbf{R}\cdot\mathbf{n}} = \frac{\mathbf{r}\cdot\mathbf{n}}{h},$$

because $\mathbf{R}\cdot\mathbf{n} = h$ for all $\mathbf{R}$ (Note that $\mathbf{n}$ is not necessarily unit vector and h is not necessarily distance.) We thus have:

$$f\mathbf{r}' = \hat{\mathbf{z}} \times \left[\mathbf{r} \times \left(\frac{\mathbf{r}\cdot\mathbf{n}}{h}\mathbf{t} + \omega \times \mathbf{r}\right)\right]$$

Now that we have arrived at a satisfactory formula for $\mathbf{r}'$, which describes it purely in terms of the "movement descriptor" variables $(\mathbf{n}, \mathbf{t}, \omega, h)$ and the pixel variable $\mathbf{r}$, we can plug in the variables describing the two movements $(\mathbf{n}, \mathbf{t}, \omega, h) = (\mathbf{b}, \mathbf{a}, \mathbf{c}, h_1)$ and $(\mathbf{n}, \mathbf{t}, \omega, h) = (\mathbf{a}, \mathbf{b}, \mathbf{c} + k\mathbf{a}\times\mathbf{b}, h_2)$, which will give us two motion field functions $f\mathbf{r}'_1$ and $f\mathbf{r}'_2$ (both are functions of pixel position $\mathbf{r}$). And then we can see when these two motion fields can be the same, i.e. when function $f\Delta\mathbf{r}' = f\mathbf{r}'_2 - f\mathbf{r}'_1$ is equal to zero for the whole image plane (i.e. for all $\mathbf{r}$):

$$\begin{aligned}
f\mathbf{r}'_1 &= \hat{\mathbf{z}} \times \left[\mathbf{r} \times \left(\frac{\mathbf{r}\cdot\mathbf{b}}{h_1}\mathbf{a} + \mathbf{c} \times \mathbf{r}\right)\right] \\
f\mathbf{r}'_2 &= \hat{\mathbf{z}} \times \left[\mathbf{r} \times \left(\frac{\mathbf{r}\cdot\mathbf{a}}{h_2}\mathbf{b} + (\mathbf{c} + k\mathbf{a}\times\mathbf{b}) \times \mathbf{r}\right)\right] \\
f\Delta\mathbf{r}' &= \hat{\mathbf{z}} \times \left[\mathbf{r} \times \left(\frac{\mathbf{r}\cdot\mathbf{a}}{h_2}\mathbf{b} - \frac{\mathbf{r}\cdot\mathbf{b}}{h_1}\mathbf{a} + k(\mathbf{a}\times\mathbf{b}) \times \mathbf{r}\right)\right] \\
&= \hat{\mathbf{z}} \times \left[\mathbf{r} \times \left(\frac{\mathbf{r}\cdot\mathbf{a}}{h_2}\mathbf{b} - \frac{\mathbf{r}\cdot\mathbf{b}}{h_1}\mathbf{a} + k(\mathbf{r}\cdot\mathbf{a})\mathbf{b} - k(\mathbf{r}\cdot\mathbf{b})\mathbf{a}\right)\right] \\
&= \hat{\mathbf{z}} \times \left[\mathbf{r} \times \left(\left(\frac{1}{h_1} + k\right)(\mathbf{r}\cdot\mathbf{b})\mathbf{a} - \left(\frac{1}{h_2} + k\right)(\mathbf{r}\cdot\mathbf{a})\mathbf{b}\right)\right]
\end{aligned}$$

In other words $\Delta\mathbf{r}' = \hat{\mathbf{z}} \times [\mathbf{r} \times \mathbf{v}]$ where $\mathbf{v} = c_1\mathbf{a} + c_2\mathbf{b}$, where c_1, c_2 are scalar functions of $\mathbf{r}$.

First observe that to make $\Delta\mathbf{r}'$ zero for all $\mathbf{r}$, $\mathbf{v}$ has to be zero for all $\mathbf{r}$. That is because $\mathbf{r}$ is always non-zero and never perpendicular to $\hat{\mathbf{z}}$, and hence for every non-zero vector $\mathbf{v}$, $\mathbf{r}\times\mathbf{v}$ will be always non-zero and never parallel to $\hat{\mathbf{z}}$, and hence $\hat{\mathbf{z}} \times [\mathbf{r} \times \mathbf{v}]$ will never be zero.

Now, $\mathbf{v} = c_1\mathbf{a} + c_2\mathbf{b}$, where c_1, c_2 are scalar functions of $\mathbf{r}$, and so there are two cases to consider if we want $\mathbf{v}$ to be always zero:

1. Let's consider first a special case when $\mathbf{a}$ and $\mathbf{b}$ are parallel, i.e. if $\mathbf{b} = C\mathbf{a}$, then $\mathbf{v} = c_1\mathbf{a} + c_2\mathbf{b} = (c_1 + Cc_2)\mathbf{a}$ is always zero if $c_1 + Cc_2 = 0$ for all $\mathbf{r}$, i.e. if

$$\left(\frac{1}{h_1} + k\right)(\mathbf{r} \cdot (C\mathbf{a})) - C\left(\frac{1}{h_2} + k\right)(\mathbf{r} \cdot \mathbf{a}) = C\left(\frac{1}{h_1} - \frac{1}{h_2}\right)(\mathbf{r} \cdot \mathbf{a}) = 0$$

   This will be always zero if $h_1 = h_2$, which we can simply denote by some constant h.

   Notice, however, that if $\mathbf{a}$ and $\mathbf{b}$ are parallel, then $k(\mathbf{a} \times \mathbf{b}) = 0$, and so the two movements scenarioes, $(\mathbf{a}, \mathbf{b}, \mathbf{c}, h)$ and $(\mathbf{b}, \mathbf{a}, \mathbf{c}, h)$, describe really the same movement of the camera, only with regards to a different unviersal coordinate system: but from the camera's point of view these two movements are the same.

2. Now take a general case when $\mathbf{a}$ and $\mathbf{b}$ are not parallel. Then $\mathbf{v}$ will be always zero only if c_1 and c_2 are zero for all $\mathbf{r}$. But that happens only if $\frac{1}{h_1} + k = 0$ and $\frac{1}{h_2} + k = 0$, i.e. when $h_1 = h_2 = h$ for some h, and if $k = \frac{-1}{h}$. See figure 2 below for an example.

   For the example scenario, we have two image planes with normal vector a and b. We can see that effect of extra rotation is to make the angles between the translational direction and the plane normal be the same so that we can easily show that the motion field is the same under two translational motion. It makes sense since to adjust same amount of motion differences caused by pure translation, the closer the camera, the less rotation angle we need. The constraint $h_1 = h_2$ must be mentioned. It is also acceptable to use the situation where a and b are parallel as an example which is easier and does not require the constraints.

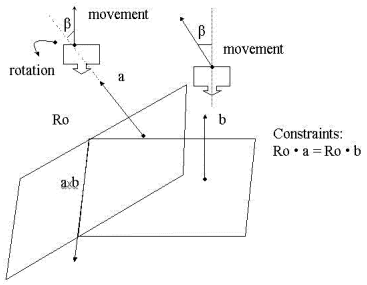

Figure 2: Ambiguity in detecting motion(Prob.5)

Note: It is important to understand the assumption and the definition of variables involved in equations.

(1) We can only assume image coordinates $r_1 = r_2$, while we should not assume $R_1 = R_2$, which is not necessarily true for all image points. It is true that the two planes have intersection, i.e., $R_1 = R_2$, which, however, holds only for a few points.

(2) We cannot assume $\mathbf{R} \cdot \mathbf{n} = \mathbf{1}$ or $\mathbf{R} \cdot \mathbf{n}$=constant, which actually assumes $\mathbf{R_1} \cdot \mathbf{n_1} = \mathbf{R_2} \cdot \mathbf{n_2}$ and is not necessarily true. Furthermore, $\mathbf{a}$ and $\mathbf{b}$ are not necessarily unit vectors.