Andrew Dodd

5/1/2016

Computer Vision: HW2

1

1. a) i.

$$V_x = (S \times R) \times (P \times Q)$$

$$= \left(\begin{bmatrix} -1 \\ -.3 \\ 1 \end{bmatrix} \times \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix} \right) \times \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ -.2 \\ 1 \end{bmatrix} \right)$$

$$\boxed{V_x = (-2, .4, 1)}$$

$$V_y = (P \times S) \times (Q \times R)$$

$$= \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \begin{bmatrix} -1 \\ -.3 \\ 1 \end{bmatrix} \right) \times \left(\begin{bmatrix} 1 \\ -.2 \\ 1 \end{bmatrix} \times \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix} \right)$$

$$\boxed{V_y = (2, .6, 1)}$$

ii. form is $y - y_1 = m(x - x_1)$ $\quad m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{.6 - .4}{2 - (-2)} = .05$

$$y - .4 = (.05)(x + 2)$$

$$\boxed{y = 0.05x + 0.5} \quad (z = 1)$$

b) i. We first notice that the line is vertical, thus $\overline{rb} \Rightarrow x = .2$

Setting this line equal to the vanishing line...

$$y = (.05)(.2) + .5 = .51$$

$$\boxed{t = (.2, .51, 1)}$$

ii. $H = \frac{5}{1} \left(.4 - (-.6) + (.51 - .4) \right)$

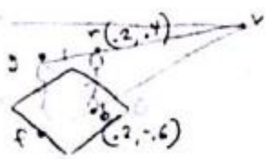
$$= \frac{r_z}{1} \left(H_{pillar} + H_{extra} \right) = 5.55$$

image coor

$$\boxed{\text{Camera is at 5.55 unit height}}$$

1 c) i) $g = (-.2, .35)$ (at $z = 5$) $r = (.2, .4)$ at $z = 5$

$f = (-.2, \ )$



$$v \cong (b \times b_0) \times (v_x \times v_y) = \left(\begin{bmatrix} .2 \\ -.6 \\ 1 \end{bmatrix} \times \begin{bmatrix} -.2 \\ f_2 \\ 1 \end{bmatrix} \right) \times \left(\begin{bmatrix} -.2 \\ .4 \\ 1 \end{bmatrix} \times \begin{bmatrix} 2 \\ .6 \\ 1 \end{bmatrix} \right) \quad \text{Eq 1}$$

$$v \cong (r \times g) \times (v_x \times v_y) = \left(\begin{bmatrix} .2 \\ .4 \\ 1 \end{bmatrix} \times \begin{bmatrix} -.2 \\ .35 \\ .1 \end{bmatrix} \right) \times \left(\begin{bmatrix} -.2 \\ .4 \\ 1 \end{bmatrix} \times \begin{bmatrix} 2 \\ .6 \\ 1 \end{bmatrix} \right) = \begin{bmatrix} 1.667 \\ .5833 \\ 1 \end{bmatrix}$$

Using points v and b we get:

$$\text{line } \overline{vb} = y = .8106x - .76212$$

$$x = -.2 \rightarrow y = -.924$$

$$\boxed{f = (-.2, -.924, 1)}$$

---

ii) 4 pts

	2D		3D
P =	(0,0)	$\leftarrow$	(0,0,0)
Q =	(1,-.2)	$\leftarrow$	(0,1,0)
R =	(0,-1)	$\leftarrow$	(1,1,0)
S =	(-1,-.3)	$\leftarrow$	(1,0,0)

$$\begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$S: \begin{bmatrix} -1w \\ -.3w \\ w \end{bmatrix} = \begin{bmatrix} a & & \\ d & & \\ g & & \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\left. \begin{array}{l} a = -w \\ d = -.3w \\ g = w \end{array} \right\} \rightarrow \begin{array}{l} a = -g \\ d = -.3g \\ d = .3a \end{array}$$

$$Q: \begin{bmatrix} w(1) \\ w(-.2) \\ w \end{bmatrix} = \begin{bmatrix} & b & \\ & e & \\ & h & \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\left. \begin{array}{l} w = b \\ -.2w = e \\ w = h \end{array} \right\} \rightarrow \begin{bmatrix} a & b & c \\ .3a & -.2b & f \\ -a & b & 1 \end{bmatrix}$$

$$\downarrow$$

$$R: \begin{bmatrix} 0 \\ -1w \\ w \end{bmatrix} = \begin{bmatrix} a & 1 & 0 \\ .3a & -.2 & 0 \\ -a & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

$$\Downarrow$$

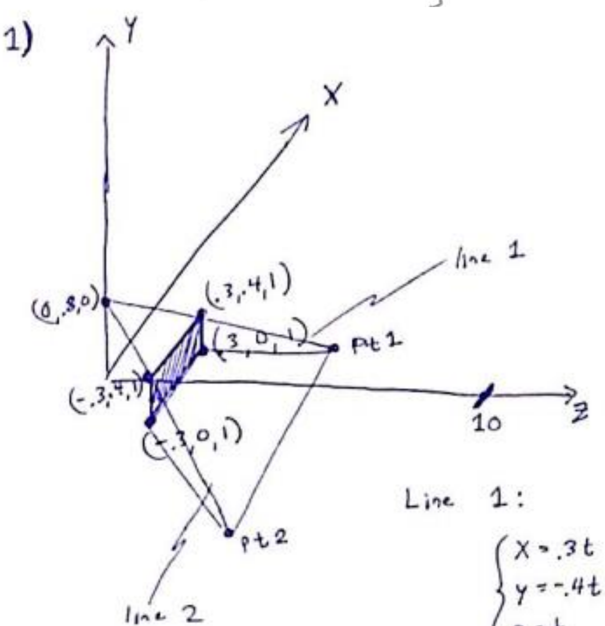
$$\boxed{H = \begin{bmatrix} -1 & 1 & 0 \\ -.3 & -.2 & 0 \\ 1 & 1 & 1 \end{bmatrix}}$$

1)c)iii) $r = \begin{pmatrix} .76 \\ .89 \\ 5 \end{pmatrix}$ $g? \begin{pmatrix} \\ \\ S \end{pmatrix}$

we can find this using the homography from part ii:

$$H^{-1} = \begin{bmatrix} -.4 & -2 & 0 \\ .6 & -2 & 0 \\ -.2 & 4 & 1 \end{bmatrix}, \text{ now taking } g_{3D} = H^{-1} g_{2D} = \boxed{\begin{bmatrix} -.096 \\ -.127 \\ 1 \end{bmatrix}}$$

3. 1)



Line 1:

$$\begin{cases} X = .3t \\ y = -.4t + .8 \\ z = t \end{cases}$$

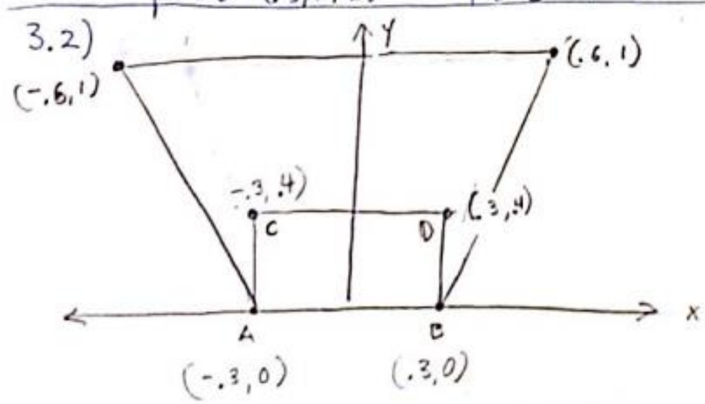
Line 2:

$$\begin{cases} x = -.3t \\ Y = -.4t + .8 \\ z = t \end{cases}$$

$y = 0 \Rightarrow t = 2 \; \therefore$ ~~xxxx~~

Pt. A = (-.3, 0, 1)	Pt D → Pt 1 = (.6, 0, 2)
Pt B = (.3, 0, 1)	Pt C → Pt 2 = (-.6, 0, 2)

3.2)



$$\boxed{H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & -1.25 & 1 \end{bmatrix}}$$

3.3) To compute this transformation we must use a 3 matrix transformation and then fit the image inside the output bounding box. To do this, we can use the same four point homography calculating algorithm from PA3 once to find H and once more to find T. Note that T is found by mapping the top left billboard coordinate to the top left image coordinate. Then the bottom right billboard coordinate cooresponds to the bottom right image coordinate. Another important matter is that the (x,y) coordinate of the image that correspond to the billboard point must be given in terms of (columns, rows) and not (rows, columns) as is often customary for image coordinates. Once the transformation $T^{-1}HT$ has been computed, we must then compute the borders of the output image but checking all four corners image coordinates. Finding the max and min differences will give the bounding box and by translating, we can put the image into this image bounding box so it will properly display. Below is the input image and output projective image.

In this computation, the below matrix was used for T:

.00371	0	-.3
0	-.002	.4
0	-1.25	1

Input **Output**

4

1. By definition, the –z axis will point through the point (0,0) on the image plane. Therefore in the camera's 3D coordinate system, the vector is (0,0,-1,1).
2. This will correspond to the vector (0, 0, -1, 1) from the previous answer that represents the camera's direction of pointing. However, this is in camera 3D coordinates. We know that R converts from world coordinates to camera coordinates so if we take the inverse of R and multiply it by the camera vector (0, 0, -1, 1) we will get the answer, the vector pointing from the origin to the object. Using Matlab:

```
R = [-.6   0 .8   0;
       0   1  0   0;
     -.8   0 -.6  0;
       0   0  0   1];
Ri = inv(R);
cameraV = Ri*[0; 0; -1; 1]; %Answer to 1
```

This gives the vector (.8, 0, .6, 1) in world 3D coordinates.

R_i (the inverse of R) was calculated to be:

```
Ri =

   -0.6000         0   -0.8000         0
         0    1.0000         0         0
    0.8000         0   -0.6000         0
         0         0         0    1.0000
```

3. Because this line passes through the camera at point (0,0,0), the line equation is fairly straightforward:
   $x = .8\, t_1$
   $y = 0$
   $z = .6\, t_1$
4. We know that the point in camera coordinates is (3, 0, -4, 1). Note that we use negative four instead of positive because the image is on the –z axis. Using Matlab:

   ```
   vector = Ri*[-3; 0; 4; 1];
   ```

   We find the vector is (1.4, 0, 4.8, 1). Using this we find the equation to be
   $x = 1.4\, t_2 + 5$
   $y = 0$
   $z = 4.8\, t_2$
5. If we solve these two equations simultaneously using algebra, we find that the object world coordinates are {x,y,z} = {6.4, 0, 4.8}

5

a.

$M = L^T N$ (6x3)

$L = [n \times 3]$ $N = [3 \times p]$

p = # pixels (3)

n = # images (6)

Our next step is to compute svd(M) to solve for L vectors, N vectors and albedos of each pixel.

b. Computing SVD and finding L_hat = U*sqrt(sigma) and S_hat = sqrt(sigma)*V_T
We get:

```
L_hat [[-0.0326092  -0.0297122  -0.27669062]
 [-0.14895025 -0.33678949  0.13440337]
 [-0.39727583  0.3060619   0.09102049]
 [-0.3341988  -0.0348913  -0.02961118]
 [-0.24961558 -0.03680966 -0.13389011]
 [-0.24303651 -0.20411884 -0.01579977]]
S_hat [[-0.13504809 -0.30843212 -0.54842755]
 [-0.07494033 -0.42472642  0.25731717]
 [-0.33745108  0.08195898  0.03700278]
 [ 0.          0.          0.        ]
 [ 0.          0.          0.        ]
 [ 0.          0.          0.        ]]
```

c. When we multiply these 3 matrices out, we need to note that B is a symmetric matrix across the diagonal, so we only have 6 unknowns.

$$l_i = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad l_i^T = \begin{bmatrix} x & y & z \end{bmatrix} \qquad B = \begin{bmatrix} B_{00} & B_{01} & B_{02} \\ B_{10} & B_{11} & B_{12} \\ B_{20} & B_{21} & B_{22} \end{bmatrix}$$

$$\begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} B_{00} & B_{01} & B_{02} \\ B_{10} & B_{11} & B_{12} \\ B_{20} & B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = 1$$

$$\begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} B_{00}x + B_{01}y + B_{02}z \\ B_{01}x + B_{11}y + B_{12}z \\ B_{02}x + B_{12}y + B_{22}z \end{bmatrix} = 1 \qquad \begin{matrix} B_{01} = B_{10} \\ B_{02} = B_{20} \\ B_{12} = B_{21} \end{matrix}$$

$$(B_{00}x^2 + B_{01}xy + B_{02}xz) + (B_{01}xy + B_{11}y^2 + B_{12}yz) + (B_{02}xz + B_{12}yz + B_{22}z^2) = 1$$

$$\begin{bmatrix} x^2 & 2xy & 2xz & y^2 & 2yz & z^2 \end{bmatrix} \begin{bmatrix} B_{00} \\ B_{01} \\ B_{02} \\ B_{11} \\ B_{12} \\ B_{22} \end{bmatrix} = 1$$

d.

```
B [[ 7.54315929   2.71885886   3.70062158]
 [  2.71885886   5.80831911   1.76507694]
 [  3.70062158   1.76507694  11.57012434]]
```

e.

```
A [[-2.01788295  1.35759808  1.27602316]
 [-1.21359485  1.52816515 -1.41429061]
 [-3.03225312 -1.51506065 -0.28311937]]
```

f. Note, the S vector for each pixel are each column. The light vector for each image is each row.

```
S_final [[ 0.094082    0.06033597  0.04628345]
 [ 0.03296327 -0.1841202  -0.06292817]
 [ 0.00787407  0.0495916  -0.28965133]]
L [[ 0.94085611  0.32952773  0.07874808]
 [ 0.3017451  -0.92051379  0.24820205]
 [ 0.15422382 -0.20952934 -0.96556329]
 [ 0.80650654 -0.46216459 -0.36871547]
 [ 0.95435575 -0.19227723 -0.22854883]
 [ 0.78604569 -0.6179356  -0.01696363]]
```

g.

The normal vectors are the columns.

```
albedos [[ 0.1       ]
 [ 0.20000001]
 [ 0.30000001]]
normals [[ 0.94081997  0.30167985  0.15427815]
 [ 0.32963267 -0.92060095 -0.20976057]
 [ 0.0787407   0.24795799 -0.96550441]]
```

h. Yes it is possible, because we are assuming that the sigma matrix is simply being broken up via a square root. However, this is an assumption, because it could be –sqrt(sigma) on both sides instead of +sqrt(sigma) on both sides. Or it could be sigma$^{1/3}$ and sigma$^{2/3}$, it is not known. In addition, we are using a least squares approach for part of the problem and this is not perfectly accurate.

6

1. T
2. T
3. T
4. F
5. F
6. T
7. T
8. F
9. F
10. T