UNIVERSITY OF MANITOBA
Final Examination

Fall 2003

COMPUTER SCIENCE

Mobile Robotics using Local Vision

Paper No.: 458
Examiners: Jacky Baltes
Date: 08 Dec. 2003
Time: 11:00
Room: Machray Hall 500A

(Time allowed: 180 Minutes)

NOTE: Attempt all questions.
This is an open book examination.
Use of calculators is permitted.
Show your work to receive full marks.

SURNAME: ____________________________
FORENAME(S): _________________________
STUDENT ID: __________________________

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>25</td>
<td>25</td>
<td>25</td>
<td>17</td>
<td>75 + 25B</td>
</tr>
</tbody>
</table>

CONTINUED
Section A: Colour Perception

1. A camera returns 24 bit color values with unsigned 8 bits for the red, green, and blue channel respectively.

The RGB color algorithm uses minimum and maximum thresholds for the individual channels (red, green, and blue). Assuming that the color parameters are given as

<table>
<thead>
<tr>
<th>Channel</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>132</td>
<td>226</td>
</tr>
<tr>
<td>Green</td>
<td>120</td>
<td>200</td>
</tr>
<tr>
<td>Blue</td>
<td>180</td>
<td>240</td>
</tr>
</tbody>
</table>

Show the part of the RGB cube that is being recognized by these color parameters. [3 marks]

![RGB Cube Diagram]

2. The extended RGB color algorithm uses a 12 parameter model which includes minimum and maximum thresholds for the individual channels (red, green, and blue) as well as minimum and maximum thresholds for the difference channels (red - green, red - blue, and green-blue). Assuming that the color parameters are given as
Section B: Image Processing

A convolution matrix is an intuitive way to represent many different image processing and enhancement operators such as edge detection, blurring, and noise cancellation.

A convolution matrix is a $n \times n$ mask that is applied to each pixel of an image.

3. Given the grey scale image $I$ below, show the image resulting from applying the convolution matrix $M$ to $I$. The maximum range of intensity values is from 0 (black) to 255 (white). The divisor of the convolution matrix is 1 and the offset is 0.

You can ignore the borders and some of the correct result values are already entered for you. You only need to show the intensity values for pixels in the grey region.

Image $I =$
Convolution Matrix \( M \) =

\[
\begin{pmatrix}
-2 & 1 & 1 \\
-1 & 2 & 2 \\
1 & -1 & -2
\end{pmatrix}
\]

[8 marks]
4. Show a 3x3 convolution matrix that emphasizes horizontal lines. That is, points that lie on a horizontal line have an increase in brightness whereas all other points are reduced in brightness.

Also indicate the divisor and offset for your convolution mask.

\[ M = \begin{bmatrix}
0 & 0 & 0 \\
1 & 1 & 1 \\
0 & 0 & 0 \\
\end{bmatrix} \]

Divisor = 3, Offset = 0

5. Instead of interpreting convolution matrices as image processing operators, they can also be interpreted as computation. Can you design a convolution matrix that can count from 0 to 1?

Given below are two grey scale images \( I_0, I_1 \). The intensity levels range from 0 (black) to 255 (white). Image \( I_1 \) is obtained from Image \( I_0 \) by applying a \( 3 \times 3 \) convolution matrix \( M_1 \) then thresholding (thres = 128) the image afterwards.
You can ignore the pixels around the edges of the image.

Show one possible convolution matrix $M_1$ that transforms $I_0$ into $I_1$. Also specify the divisor and offset for this convolution matrix. If no such matrix exists, then say so in your answer and explain why it is impossible that such a matrix exists.

[12 marks]

Many such convolution masks exist. The following is a simple one. $M = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

Divisor = 1, Offset = 0
Section C: Optical Flow

The following is a simplified implementation of the Horn Schnuck optical flow algorithm as discussed in class. In particular, the algorithm does not average the velocities $v_x$ and $v_y$.

$$
\begin{align*}
 v^{k+1}_x &= v^k_x - \frac{\left[ \frac{\delta I}{\delta x} v^k_x + \left( \frac{\delta I}{\delta y} \right) v^k_y + \frac{\delta I}{\delta t} \right]}{\lambda + \left( \frac{\delta I}{\delta x} \right)^2 + \left( \frac{\delta I}{\delta y} \right)^2} \frac{\delta I}{\delta x} \\
 v^{k+1}_y &= v^k_y - \frac{\left[ \frac{\delta I}{\delta x} v^k_x + \left( \frac{\delta I}{\delta y} \right) v^k_y + \frac{\delta I}{\delta t} \right]}{\lambda + \left( \frac{\delta I}{\delta x} \right)^2 + \left( \frac{\delta I}{\delta y} \right)^2} \frac{\delta I}{\delta y}
\end{align*}
$$

The following equations are used to compute the spatio temporal derivatives $\frac{\delta I}{\delta x}$, $\frac{\delta I}{\delta y}$, and $\frac{\delta I}{\delta t}$

$$
\begin{align*}
 \frac{\delta I}{\delta x} &= M_x \ast (I_1 + I_2) \\
 \frac{\delta I}{\delta y} &= M_y \ast (I_1 + I_2) \\
 \frac{\delta I}{\delta t} &= M_t \ast (I_2 - I_1)
\end{align*}
$$

where the convolution masks $M_x$, $M_y$, and $M_t$ are given as follows

$$
\begin{align*}
 M_x &= \frac{1}{4} \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, \\
 M_y &= \frac{1}{4} \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}, \\
 M_t &= \frac{1}{4} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\end{align*}
$$

6. You are also given two $4 \times 4$ images $I_0$ and $I_1$. The intensity levels range from 0 (black) to 255 (white).
Compute the partial temporal derivative $\frac{\delta I}{\delta t}$ for the image. You can ignore the border of the image.

\[
\frac{\delta I}{\delta t} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
0 & 63 & 63 & 0 & 0 \\
0 & 0 & 63 & 63 & 0 \\
0 & -63 & -63 & 0 & 0 \\
0 & 0 & -63 & -63 & 0 \\
\end{bmatrix}
\]

7. Compute the partial derivative in the x direction $\frac{\delta I}{\delta x}$. You can ignore the border of the image.

\[
\frac{\delta I}{\delta x} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
-127 & 63 & 63 & 0 & 0 \\
-255 & 127 & 63 & 63 & 0 \\
-255 & 63 & 63 & 127 & 0 \\
-127 & 0 & 63 & 63 & 0 \\
\end{bmatrix}
\]

8. Compute the partial derivative in the y direction $\frac{\delta I}{\delta y}$. You can ignore the border of the image.

\[
\frac{\delta I}{\delta y} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
-127 & 63 & 63 & 0 & 0 \\
-255 & 127 & 63 & 63 & 0 \\
-255 & 63 & 63 & 127 & 0 \\
-127 & 0 & 63 & 63 & 0 \\
\end{bmatrix}
\]
9. Assume that the coefficient $\lambda$ is set to 0.7. Show the output (i.e., the optical flow velocities $v_x$ and $v_y$) of the first iteration of the Horn Schnuck algorithm for the area $(1,1)$ to $(3,3)$ of the images.

$$\frac{\delta I}{\delta y} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
-127 & -191 & -63 & 0 & 0 \\
0 & 0 & -63 & -63 & 0 \\
0 & -63 & -63 & 0 & 0 \\
127 & 255 & 191 & 63 & 0 \\
\end{bmatrix}$$

$$v_x^1 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
0 & -0.09 & -0.49 & 0 & 0 \\
0 & 0 & -0.49 & -0.49 & 0 \\
0 & 0.49 & 0.49 & 0 & 0 \\
0 & 0 & 0.09 & 0.49 & 0 \\
\end{bmatrix}$$

$$v_y^1 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
0 & 0.26 & 0.25 & 0 & 0 \\
0 & 0 & 0.25 & 0.25 & 0 \\
0 & 0 & 0.25 & 0 & 0 \\
0 & 0 & 0.26 & 0.25 & 0 \\
\end{bmatrix}$$

10. Assume that the coefficient $\lambda$ is set to 0.7. Show the output (i.e., the optical flow velocities $v_x$ and $v_y$) of the second iteration of the Horn Schnuck algorithm for the area $(1,1)$ to $(3,3)$ of the images.

[7 marks]
Section D: Localization using Computer Vision

The following equations show the kinematics of a differential drive robot given the left and right wheel velocities ($v_l$ and $v_r$ respectively). The width of the robot is given as $w$.

\[
\dot{x} = \frac{v_r + v_l}{2} \cos \theta \\
\dot{y} = \frac{v_r + v_l}{2} \sin \theta \\
\dot{\theta} = \frac{v_r - v_l}{w}
\]

11. The state of the robot at time $t_0 = 0.00$ s is $x = 15.00$ cm, $y = 10.00$ cm, $\theta = 30.00^\circ$. The robot has a width of 20 cm.

How long will it take the robot to turn by 25 degrees, assuming that the left and right wheel velocities are 0.2 m/s and $-0.6$ m/s respectively?

[5 marks]
It takes 0.10 sec. to turn by 25 degrees.

\[ td = \frac{-0.6 - 0.2}{0.2} = -4. \]
\[ \frac{25}{180} \times \pi \]
\[ ans = 0.4363323 \]
\[ \frac{ans}{td} \]
\[ ans = -0.1090831 \]

12. Given is a differential drive robot \( R \) with a width of 200mm and a camera. In the view of the robot is a line. The image processing system of the robot is able to recognize the angle \( \theta' \) and distance \( d \) of the line with respect to the robot.

Assume that the estimate of the robot’s position in the world coordinate system \( W \) at time \( t_0 \) is \( P(t_0) = (x(t_0), y(t_0)) = (10.0, 20.0) \) mm, \( \theta(t_0) = 60^\circ \). From this position, the robot detects the line and determines that the angle and distance of the robot to the line are \( \theta'(t_0) = 30^\circ \) and \( d(t_0) = 22.0 \) mm respectively.

At time \( t_1 = 0.05 \) s, the robot detects the line and determines that the angle and distance of the robot to the line are now \( \theta'(t_1) = 38^\circ \) and \( d(t_0) = 27.0 \) mm.

Compute the angular velocity \( \dot{\theta} \) of the robot in degrees per second.

[5 marks]
\[ \dot{\theta} = 160^\circ/\text{sec} \]

13. Compute the right \(v_r\) and left \(v_l\) wheel velocities of the robot R.  
[10 marks]

\[ v_r = \ldots, \quad v_l = \ldots \]

14. Assume that a robot knows its current position in the world coordinate system and that your robot can only determine the angle \(\theta'\) of the robot and the line, but not the distance \(d\). Which ones of the following quantities can be determined through this information. One of the answers is already shown.  
[5 marks]

- Angular velocity \(\dot{\theta}\): Yes
- Left wheel velocity \(v_l\): No
- Right wheel velocity \(v_r\): No
- New orientation \(\theta\) in world coordinates: Yes
- Linear velocity of the robot \(v\): No
- New X coordinate in world coordinates: No
- New Y coordinate in world coordinates: No
Section E: Hough Transform

The Hough transform uses the following equation for a line

\[ x \cos \theta + y \sin \theta = \rho \]

15. The Hough transform is applied to an image and results in the Hough space below. By analyzing the Hough space, a student suggests that the image must have contained an approximate square.
Is this conclusion correct? If the student is correct then explain why the image must contain a square and the coordinates of the square. If the student is incorrect then give a counter example of an image that does not contain a square, but that will result in the Hough space shown below.

This conclusion is incorrect since the Hough transform detected strong maximas at (90, 150), (90, 350), (0, 0), (0, 200). The maximas have a value of 7 or 8, which means that the length of the detected lines is at most 7 or 8 pixels long. Therefore, the lines do not necessarily need to form a square, since the diagonal of the picture is about 250, which means that the image is about 15 by 15 pixels.

16. The Generalized Hough Transform is an extension of the standard Hough Transform to detect arbitrary shapes. Each pixel votes for in this case possible centers of a shape.

You are applying the Generalized Hough Transform to detect the following shape in the image. Given that your system finds an edge pixel at the x,y position 6,10, what are the possible centers of the shape that are supported by an edge pixel.

[8 marks]

CONTINUED
You do not have to worry about possible rotations of the shape, only translations (i.e., shifts up/down or right/left). [7 marks]

A pixel at the x,y position (6,10) supports the following center positions:
(6,11),(5,11),(4,11),(3,11),(2,11),(3,10),(3,12),(4,12)

17. You are trying to determine the center of the shape above. The shape can be translated, but not rotated. What is the dimensionality of the associated Hough Space? [2 marks]

The associated Hough Space has 2 dimensions
Additional work pages
Additional work pages