Institut für Mess- und Regelungstechnik
mit Maschinenlaboratorium
Karlsruher Institut für Technologie
(KIT)
Prof. Dr.-Ing. C. Stiller

Solutions for exam „Machine Vision“ August 7, 2018

Question 1 (8 points)

Let $f : \mathbb{R} \to \mathbb{R}$ be a function. Let $g(t) = f(-t)$. Prove the following property of the Fourier transform:

$$\mathcal{F}\{f\}(x) = \mathcal{F}^{-1}\{g\}(x) \quad \text{for all } x \in \mathbb{R}$$

Note, that $\mathcal{F}\{f\}(k)$ denotes the Fourier transform of the function $f(t)$.

Solution

$$\begin{aligned}
\mathcal{F}^{-1}\{g\}(x) &= \int_{-\infty}^{+\infty} f(-t) \cdot e^{2\pi i x t} dt \\
&\underset{u=-t}{=} \int_{+\infty}^{-\infty} f(u) \cdot e^{-2\pi i x u} (-du) \\
&= \int_{-\infty}^{+\infty} f(u) \cdot e^{-2\pi i x u} du = \mathcal{F}\{f\}(x)
\end{aligned}$$

Question 2 (2+3+3 points)

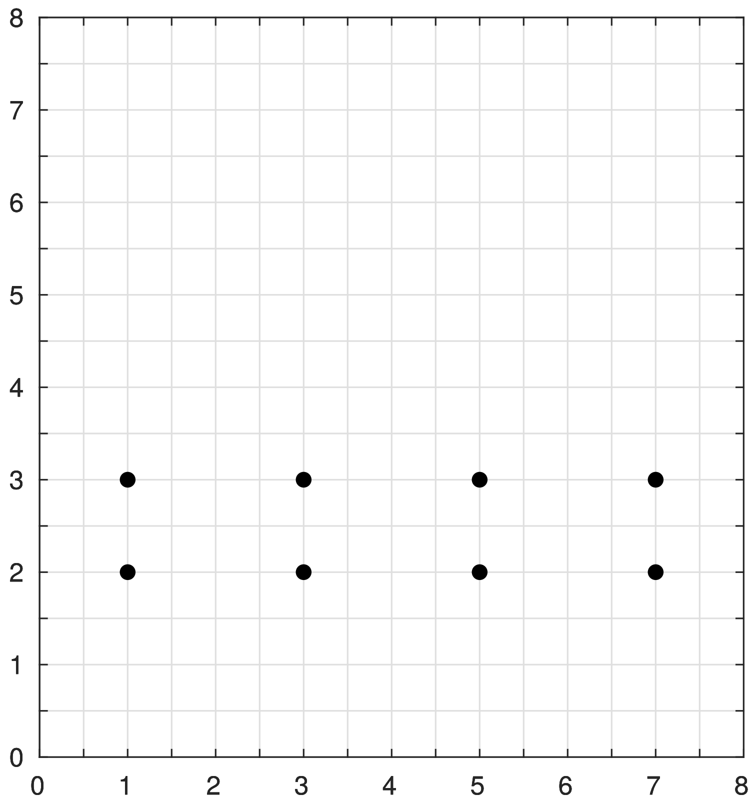
In the figure below, eight points are given that should be approximated using a line fitting algorithm.

(a) Draw the optimal solution of a total least sum of squares line fitting approximation into the figure below.

(b) Is it possible to obtain the line from part (a) using the RANSAC algorithm as introduced in the lecture? Either calculate the probability that RANSAC finds that solution in one trial or justify briefly why it is impossible.

(c) Draw all optimal RANSAC solutions with $\theta = \frac{1}{2}$. How many outliers do they have? Mark your solution of part (c) clearly so that it can be distinguished from your solution of part (a).

Remark: Note that the line $y = 3$ is not an optimal RANSAC solution for $\theta = \frac{1}{2}$

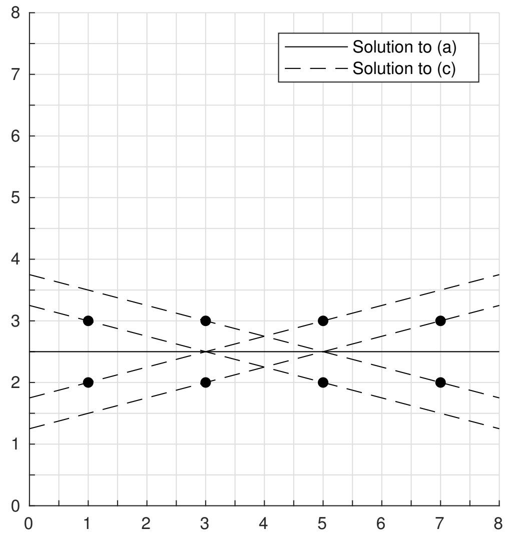


Solution

(a) see figure below.

(b) No it is not. All RANSAC line-fit solutions go through at least two data points. Therefore the optimal solution cannot be found by RANSAC

(c) see figure below. There are 4 optimal solutions with each 3 outliers and 5 inliers. All other possible RANSAC solutions have more than 3 outliers for $\theta = \frac{1}{2}$.



Best least-square-fit and best RANSAC fits.

Question 3 (8 points)

In the lecture we introduced a white balance procedure for RGB encoded color images that used two correction factors c_R and c_B to modify the R- and B-channels while the G-channel was preserved. The disadvantage of this method is that the average image brightness changes. Modify the method in such a way that the average image brightness is preserved. For that purpose introduce a third correction factor c_G to modify the G-channel. Derive values for the three correction factors c_R, c_G, and c_B as functions of the average R-, G-, and B-values within a gray region of interest denoted as $\bar{R}$, $\bar{G}$, and $\bar{B}$, respectively, and the average brightness $\bar{L}$ of that area.

Remark: The brightness L of an RGB encoded color pixel (R,G,B) is calculated as $L = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$. Note that $0.299 + 0.587 + 0.114 = 1$.

Solution

We want to calculate factors c_R, c_G, and c_B such that

$$c_R \cdot \bar{R} = c_G \cdot \bar{G} = c_B \cdot \bar{B}$$

and

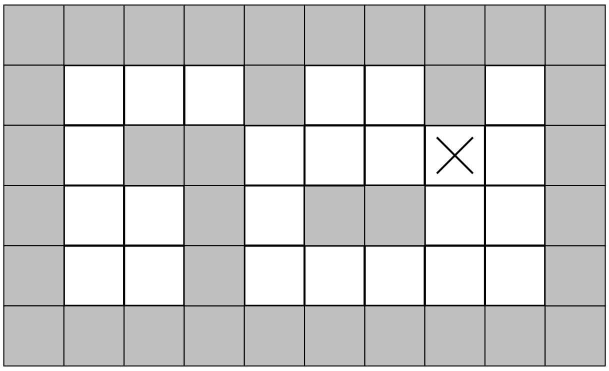
$$\bar{L} = 0.299 \cdot c_R \cdot \bar{R} + 0.587 \cdot c_G \cdot \bar{G} + 0.114 \cdot c_B \cdot \bar{B}$$

By combining these equations we obtain

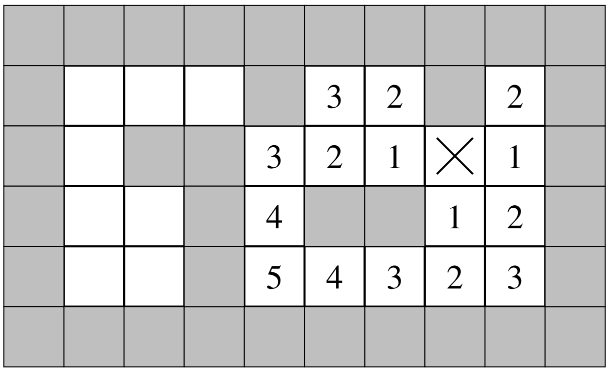
$$c_R = \frac{\bar{L}}{\bar{R}}, \quad c_G = \frac{\bar{L}}{\bar{G}}, \quad c_B = \frac{\bar{L}}{\bar{B}}$$

Question 4 (6 points)

Apply the *region growing* algorithm to the image shown in the figure below. Each pixel is either gray or white. The seed pixel is marked with a cross. Add pixels only if they share the same gray value. Use the 4-neighborhoodship (including the left, right, lower, and upper neighbor) for the growing step. Mark the added pixel with a number that describes in which region growing step the pixel has been added (start with 1 for the first region growing step). Leave all other pixels blank.



Solution



Question 5 (8 points)

Assume that we used Tsai's calibration approach to calibrate a camera. The approach returns as result for the 3×4 projection matrix $M = A \cdot (R|\vec{t})$

$$M = \begin{pmatrix} 200 & -200\sqrt{3} & 500 & 5000 \\ 200\sqrt{3} & 200 & 300 & 2600 \\ 0 & 0 & 1 & 6 \end{pmatrix}$$

where A denotes the matrix of intrinsic parameters, R the rotation matrix and $\vec{t}$ the translation vector between the origin of the world coordinate system and the camera coordinate system.

Calculate the translation vector $\vec{t}$.

Solution

We follow the approach described on slide 41 of chapter 7.

$$
\begin{aligned}
t_3 &= m_{3,4} = 6 \\
v_0 &= \langle \vec{m}_{3,1:3}, \vec{m}_{2,1:3} \rangle = 300 \\
u_0 &= \langle \vec{m}_{3,1:3}, \vec{m}_{1,1:3} \rangle = 500 \\
\frac{\beta'}{\sin\theta} &= \sqrt{||\vec{m}_{2,1:3}||^2 - v_0^2} = \sqrt{12 \cdot 10^4 + 4 \cdot 10^4 + 9 \cdot 10^4 - 9 \cdot 10^4} = 400 \\
t_2 &= \frac{1}{\frac{\beta'}{\sin\theta}} (m_{2,4} - v_0 t_3) = \frac{2600 - 300 \cdot 6}{400} = 2 \\
\beta' \cot\theta &= \frac{1}{\frac{\beta'}{\sin\theta}} (u_0 v_0 - \langle \vec{m}_{1,1:3}, \vec{m}_{2,1:3} \rangle) = \frac{15 \cdot 10^4 - 4 \cdot 10^4 \sqrt{3} + 4 \cdot 10^4 \sqrt{3} - 15 \cdot 10^4}{400} = 0 \\
\alpha' &= \sqrt{||\vec{m}_{1,1:3}||^2 + (\beta' \cot\theta)^2 - u_0^2} = \sqrt{4 \cdot 10^4 + 12 \cdot 10^4 + 25 \cdot 10^4 + 0 - 25 \cdot 10^4} = 400 \\
t_1 &= \frac{1}{\alpha'} (m_{1,4} + \beta' \cot\theta t_2 - u_0 t_3) = \frac{5000 + 0 - 500 \cdot 6}{400} = 5
\end{aligned}
$$

Hence, $\vec{t} = (5,2,6)^T$

Question 6 (2+2+2 points)

Which of the following functions can be used as kernel of a support vector machine? Justify your answers briefly.

(a) $f(x,y) = \sin x \cdot \sin y + \cos x \cdot \cos y$

(b) $g(x,y,z) = x \cdot y \cdot z$

(c) $h(x,y) = e^{-(x-y)^2}$

Solution

(a) f can be used as kernel function since $f(x,y) = \langle (\sin x, \cos x), (\sin y, \cos y) \rangle$, i.e. f is a composition of the nonlinear mapping $z \mapsto (\sin z, \cos z)$ and the dot product in feature space.

(b) g cannot be used as kernel function since a kernel function takes two arguments instead of three.

(c) h is a radial basis function kernel as introduced in the lecture.

Question 7 (8 points)

Implement a MATLAB function that takes a graylevel image and returns the integral image. Do not use the MATLAB built-in function *integralImage*.

Solution

```
function [ J ] = integral_image ( I )
    [ rows cols ] = size(I);
    J = I;
```

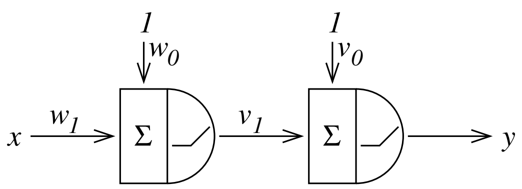
```
   for u=2:cols
      J(1,u)=J(1,u-1)+I(1,u);
   end
   for v=2:rows
      J(v,1)=J(v-1,1)+I(v,1);
      for u=2:cols
         J(v,u) = J(v-1,u)+J(v,u-1)-J(v-1,u-1)+I(v,u);
      end
   end
end
```

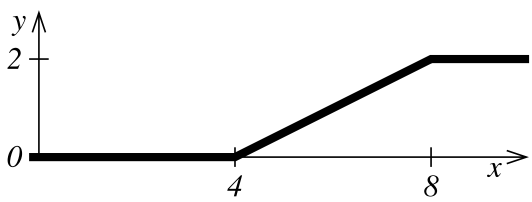
Question 8 (8 points)

Assume a small multi layer perceptron with two concatenated perceptrons. Both perceptrons use the ReLU activation function. Perceptron 1 takes the network input x which consists out of a single real number. The output of perceptron 1 serves as the only input for perceptron 2. The output of perceptron 2 is used as network output y. The weights of perceptron 1 are denoted as w_0 (bias weight) and w_1 (weight for input x), the weights for perceptron 2 are denoted as v_0 (bias weight) and v_1. The figure below on the left illustrates the network architecture. Derive weights w_0, w_1, v_0, and v_1 such that the multi layer perceptron implements the function

$$f : x \mapsto \begin{cases} 0 & \text{if } x \leq 4 \\ \frac{x}{2} - 2 & \text{if } 4 < x \leq 8 \\ 2 & \text{if } 8 < x \end{cases}$$

which is illustrated in the figure below on the right.





Solution

The multi layer perceptron implements the function

$$\begin{aligned} y(x) &= \max\{0, v_0 + v_1 \cdot \max\{0, w_0 + w_1 x\}\} \\ &= \begin{cases} \max\{0, v_0, v_0 + v_1 \cdot (w_0 + w_1 x)\} & \text{if } v_1 \geq 0 \\ \max\{0, \min\{v_0, v_0 + v_1 \cdot (w_0 + w_1 x)\}\} & \text{if } v_1 < 0 \end{cases} \end{aligned}$$

Apparently, the second case is providing a lower bound of 0 and an upper bound of v_0 which can be used to model the asymptotic behavior of f for a choice of $v_0 = 2$. The term $v_0 + v_1 \cdot (w_0 + w_1 x)$ can be used to model the ramp. With $2 + v_1 \cdot (w_0 + w_1 \cdot 4) = 0$ and $2 + v_1 \cdot (w_0 + w_1 \cdot 8) = 2$ we obtain two equations that describe the two endpoints of the ramp. Resolving those equations with respect to w_0 and w_1 yields $w_0 = -\frac{4}{v_1}$ and $w_1 = \frac{1}{2v_1}$. We can choose an arbitrary negative value for v_1, e.g. $v_1 = -1$ and obtain in total: $v_0 = 2$, $v_1 = -1$, $w_0 = 4$, $w_1 = -\frac{1}{2}$.

Gesamtpunkte: 60