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

Solutions for exam "Machine Vision" September 9, 2022

Question 1 (6 points)

Let f, g, and h be arbitrary functions $\mathbb{R} \to \mathbb{C}$. We denote with $f * g$ the convolution of two functions, with $f \cdot g$ the point-wise product of two functions, and with $\mathcal{F}\{g\}$ the Fourier transform of a function. Which of the following terms are equal? Each correct answer yields +2 points, each wrong answer -1 point. The whole task will be awarded with at least 0 points.

(1) $\mathcal{F}\{g\} + \frac{1}{2}\mathcal{F}\{h\}$

(2) $\frac{1}{4}\mathcal{F}\{2g + h\}$

(3) $\mathcal{F}\{3g \cdot f + g \cdot 3h\}$

(4) $f * h + 2g * f$

(5) $f * (2g + h)$

(6) $3\mathcal{F}\{g * (f + h)\}$

(7) $3\mathcal{F}\{g\} * \mathcal{F}\{f + h\}$

(8) $\mathcal{F}\{g + \frac{1}{2}h\}$

(9) $\mathcal{F}\{f \cdot (2g + h)\}$

Solution

To solve the task we just need to consider the properties of convolution and of the Fourier transform introduced in the lecture. So we get (1)=(8) due to the linearity of the Fourier transform, (3)=(7) since the Forier transform replaces convolution by multiplication, and (4)=(5) since convolution is linear and commutative.

Question 2 (6 points)

The picture below shows a small gradient image with 5 by 6 pixels. Each pixel is illustrated as a box. The gradient of each pixel is given by the pair of numbers $\frac{\partial g}{\partial u}, \frac{\partial g}{\partial v}$. Apply double-thresholding with the two thresholds 3 and 4 to find the edge pixels. Mark each edge pixel with a circle ($\circ$) and each non-edge pixel with a cross ($\times$).

You will obtain $+\frac{1}{5}$ point for each correct pixel and $-\frac{1}{5}$ point for each incorrect pixel. Pixels that are not marked will not be considered. The whole task will not be awarded with negative points.

3,0	2,1	3,–1	3,–3	1,–2	0,–4
2,0	1,1	2,–1	2,–2	2,–2	2,–3
3,1	0,1	2,–3	3,–3	3,–3	3,–2
3,2	0,0	0,–2	2,–2	4,–2	2,–1
2,3	1,3	–1,1	2,–1	2,0	2,0

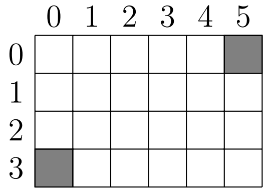
Solution

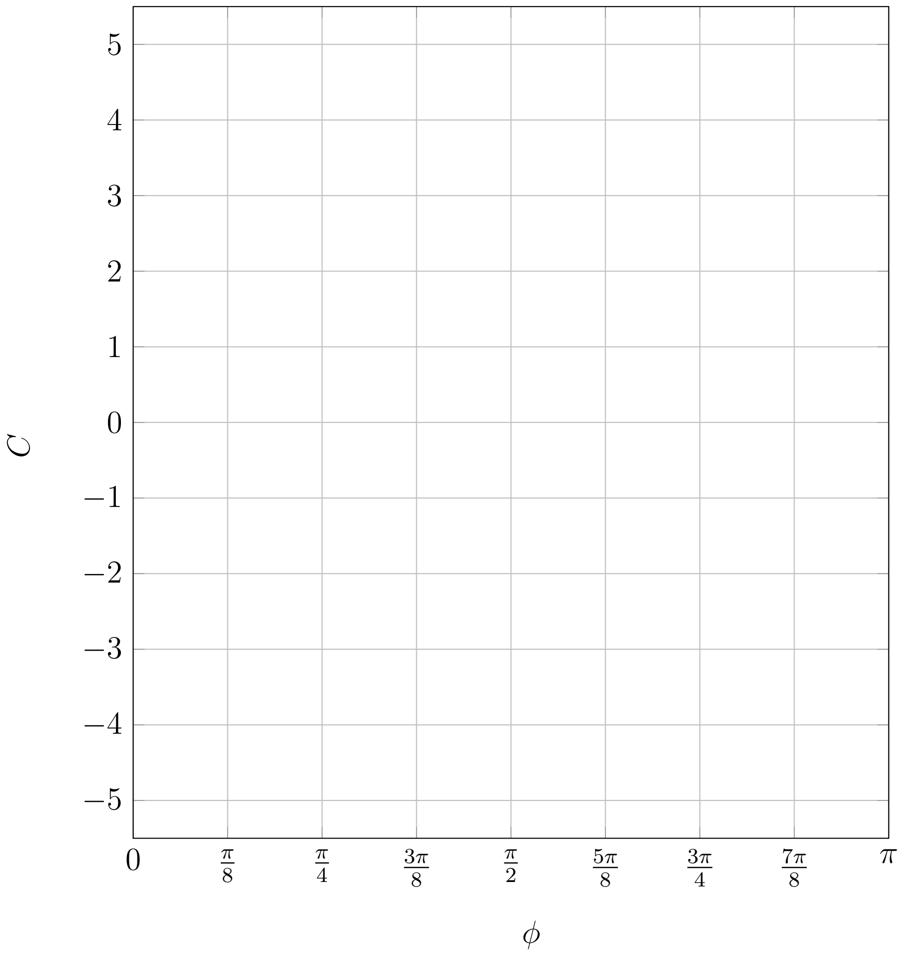
~~3,0~~	~~2,1~~	3,–1	3,–3	~~1,–2~~	0,–4
~~2,0~~	~~1,1~~	~~2,–1~~	~~2,–2~~	~~2,–2~~	2,–3
~~3,1~~	~~0,1~~	2,–3	3,–3	3,–3	3,–2
~~3,2~~	~~0,0~~	~~0,–2~~	~~2,–2~~	4,–2	~~2,–1~~
~~2,3~~	~~1,3~~	~~–1,1~~	~~2,–1~~	~~2,0~~	~~2,0~~

Question 3 (8 points)

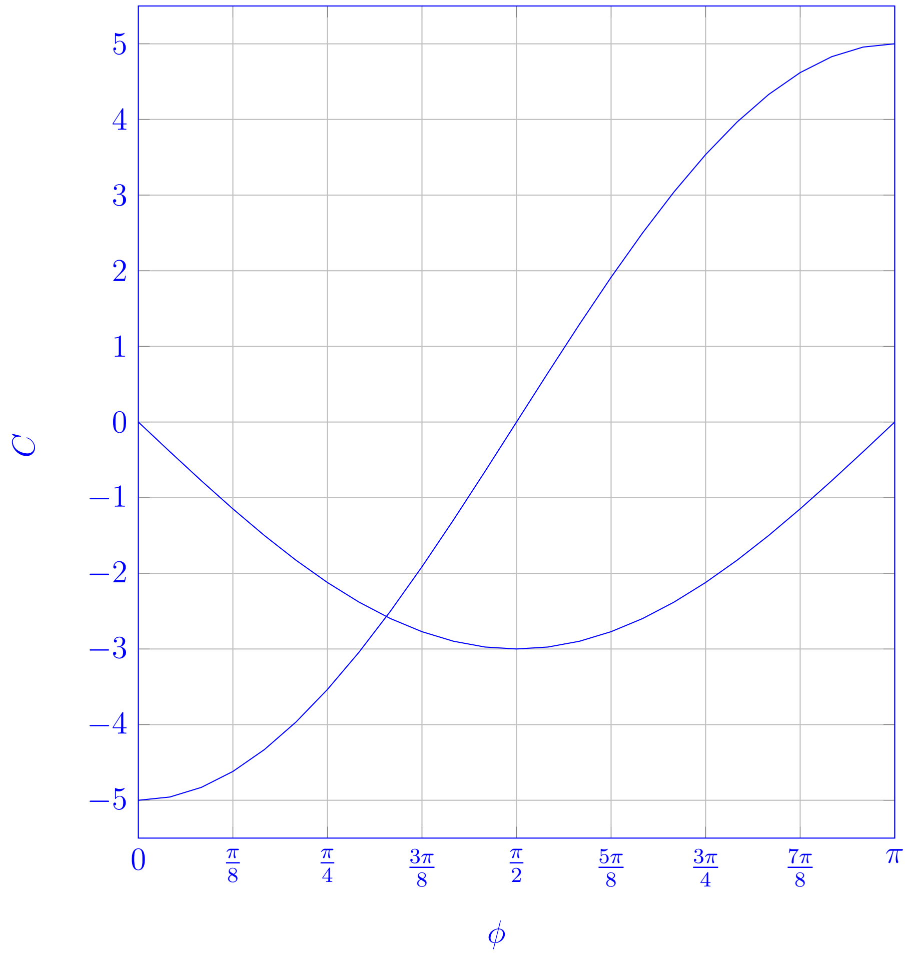
The image I shows a 4-by-6 pixel image. The two pixels highlighted in gray are edge pixels. Transform the two edge pixels into Hough space and draw the corresponding curves in the graph below. For each edge pixel provide a formula that describes the curve in Hough space.

I :





Solution



The formula for the two edge pixels are:

$$\begin{aligned} 0 &= 3 \cdot \sin\phi + c \\ 0 &= 5 \cdot \cos\phi + c \end{aligned}$$

Question 4 (8 points)

We want to apply the mean-shift algorithm for image segmentation to a small gray value image with 12 pixels. The gray values of the 12 pixels are shown in the drawing below. Apply the mean-shift algorithm for the pixel with the gray value 15. Consider for the set S all pixels which have gray values that deviate no more than at most 5.1. For each iteration of the mean-shift algorithm provide the previous reference color/gray value c, the set S, and the new reference c. State explicitely when the mean-shift algorithm terminates.

65	56	21	7
89	72	15	4
8	2	10	3

Solution

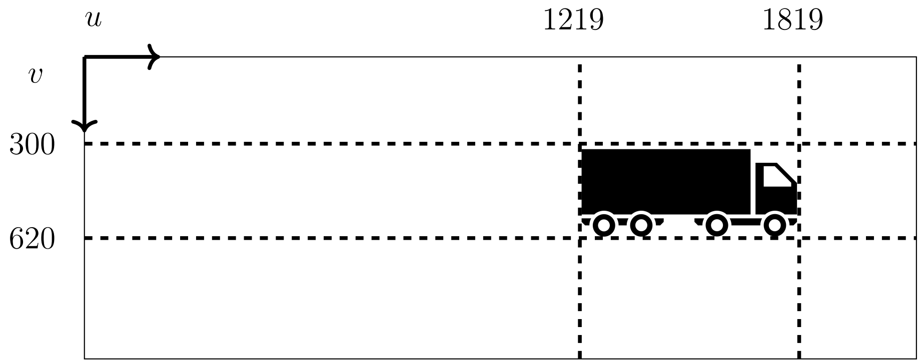
Iteration	c (before)	S	c (new)
1	15	$\{15, 10\}$	12.5
2	12.5	$\{15, 10, 8\}$	11
3	11	$\{15, 10, 8, 7\}$	10
4	10	$\{15, 10, 8, 7\}$	10

Since the set S does not change from iteration 3 to iteration 4, the mean-shift algorithm stops after iteration 4.

Question 5 (6+4 points)

Assume a setup in which a camera is mounted on the roof of an automated vehicle. The camera can be modeled using the pinhole camera model with the parameters given by matrix A. The camera is mounted such that the x- and z-axes of the camera coordinate system are parallel to the ground plane. Instance segmentation on a certain camera image yields a bounding box of a truck as given in the picture below.

$$A = \begin{pmatrix} 550 & 0 & 1040 \\ 0 & 550 & 510 \\ 0 & 0 & 1 \end{pmatrix}$$



(a) Assume that the ground plane is given as $y = 2\,\mathrm{m}$ in the camera coordinate system. Determine the distance to the truck.

Remark: You can assume that the wheels of the truck are in contact with the ground plane.

(b) Another, very precise lidar sensor is sensing a distance to the truck as $d = 11\,\mathrm{m}$. We assume that the mismatch between the lidar measurement and the result from part (a) is caused by an error in the given height of the ground plane. Recalculate the height y of the ground plane such that the distance measurement of the lidar and the content of the image are consistent.

Solution

(a)

$$z \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = A \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

$$zv = 550y + 510z$$

$$z = \frac{550y}{v - 510}$$

$$z = \frac{550 \cdot 2\,\mathrm{m}}{620 - 510}$$

$$z = 10\,\mathrm{m}$$

(b)

$$zv = 550y + 510z$$

$$y = \frac{zv - 510z}{550}$$

$$y = \frac{11\,\mathrm{m} \cdot (620 - 510)}{550}$$

$$y = 2.2\,\mathrm{m}$$

Question 6 (4 points)

Explain why it is impossible to calibrate all intrinsic and extrinsic parameters of a pinhole camera model from a single image of a flat checkerboard.

Remark: Compare the number of parameters that need to be estimated and the number of equations that you obtain from a single checkerboard image.

Solution

A mapping from a flat planar structure like the checkerboard to the image plane is a homography which itself can be described by a 3×3 matrix H with 9 entries. However, there is still one degree of freedom in H, i.e. H can be multiplied with a positive number and still represents the same homography, so that the real number of parameters of H is at most 8. H can be determined from a single image of a checkerboard. However, the number of intrinsic parameters (between 3 and 5 depending on the number of additional constraints) and extrinsic parameters (6) is larger than 8 so that it is impossible to determine all of them from a single homography.

Question 7 (5 points)

Assume you have trained a non-linear support vector machine (SVM). Which pieces of information do you need to store that you can apply the non-linear SVM later on new examples? Correct answers yield +1 point, incorrect answers -1 point. The whole task will be awarded with at least 0 points.

Solution

- suport vectors
- bias weight
- class labels of support vectors
- Lagrange multipliers
- kernel type
- kernel parameters

Question 8 (7 points)

Implement a MATLAB-function that calculates a histogram of oriented gradients (HOG) from the gradients of a gray level image. The gradients are provided in terms of two matrices. The first matrix `dir` contains the direction of the gradient in each pixel. The directions are encoded $\phi \in (-\pi, \pi]$. The second matrix `len` contains the length of the gradient in each pixel. The histogram of oriented gradients should discretize the gradient direction to eight equal sectors as given in the table below. The histogram should be represented as vector with eight entries. Use the gradient length to weight each gradient direction. You don't need to normalize the histogram.

index	bin
1	$(-\pi, -\frac{3\pi}{4}]$
2	$(-\frac{3\pi}{4}, -\frac{\pi}{2}]$
3	$(-\frac{\pi}{2}, -\frac{\pi}{4}]$
4	$(-\frac{\pi}{4}, 0]$
5	$(0, \frac{\pi}{4}]$
6	$(\frac{\pi}{4}, \frac{\pi}{2}]$
7	$(\frac{\pi}{2}, \frac{3\pi}{4}]$
8	$(\frac{3\pi}{4}, \pi]$

Remark: MATLAB-function `ceil(x)` rounds the variable x to the next integer greater or equal x, whereas `floor(x)` rounds the variable x to the next integer less or equal x.

```
function histogram = hog(dir, len)
```

Solution

```
function histogram = hog(dir, len)
   histogram = zeros (8, 1);
   for v = 1:size (dir, 1)
     for u = 1:size (dir, 2)
       i = ceil(dir(v, u) * 4 / pi);
       histogram(i + 4) = histogram(i + 4) + len(v, u);
     end
   end
 end
end
```

Question 9 (2+2+2 points)

(a) Why is it impossible to approximate the identity function $x \mapsto x$ (for all $x \in \mathbb{R}$) with a multi layer perceptron with hyperbolic tangent activation function ($tanh$)?

(b) Why is it impossible to approximate the exponential function $x \mapsto e^x$ (for all $x \in \mathbb{R}$) with a multi layer perceptron with ReLU activation function?

(c) How can we implement the logical NOT-function with a single perceptron with ReLU activation function? The NOT-function maps input $x = 0$ to output $y = 1$ and input $x = 1$ to output $y = 0$. Provide the weights of the perceptron.

Solution

(a) The hyperbolic tangent function is bounded by -1 and 1. Hence, the output of the neural network is also bounded. However, the identity function is unbounded.

(b) The derivative of the ReLU is bounded by 0 and 1. Hence the derivative of the neural network is also bounded. However, the derivative of the exponential function is unbounded.

(c) The perceptron has two weights w_0 (bias weight) and w_1 (input from x). To obtain $y = 1$ for $x = 0$ we need $w_0 = 1$. To obtain $y = 0$ for $x = 1$ we need $w_1 = -1$.

Gesamtpunkte: 60