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

# Solutions for exam "Machine Vision" March 18, 2022

Question 1 (5 points)

Constant motion blur should be added to a dataset of images. The movement is expected to run parallel along the $v$-axis in image coordinates. The motion blur reflects a pitch movement of the camera from from top to bottom (German: eine Nickbewegung der Kamera). This should be modeled by a convolution.

(a) Create a convolution mask that models this kind of motion blur as a point-spread-function. The length of the motion blur should extend over 3 pixels. Do not add additional blur in $u$-direction. Fill out the convolution matrix $f$ below and normalize its values. The circle in $f$ indicates its origin.

(b) How can simulated motion blur on an existing dataset be used with respect to deep learning?

$f:$

| | | | | |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

**Solution**

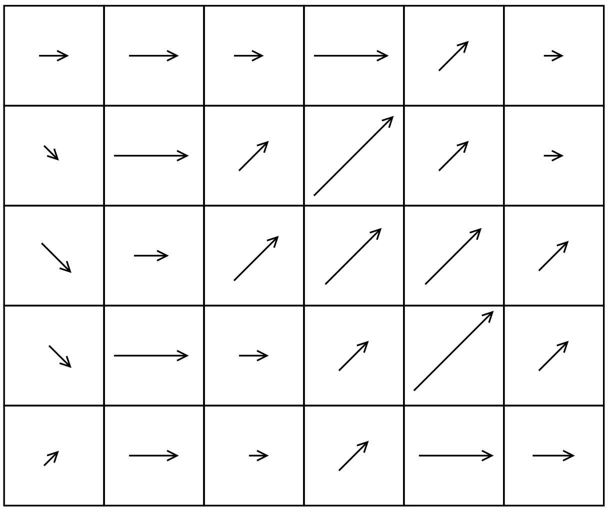
(a)

$$f: \quad \frac{1}{3} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

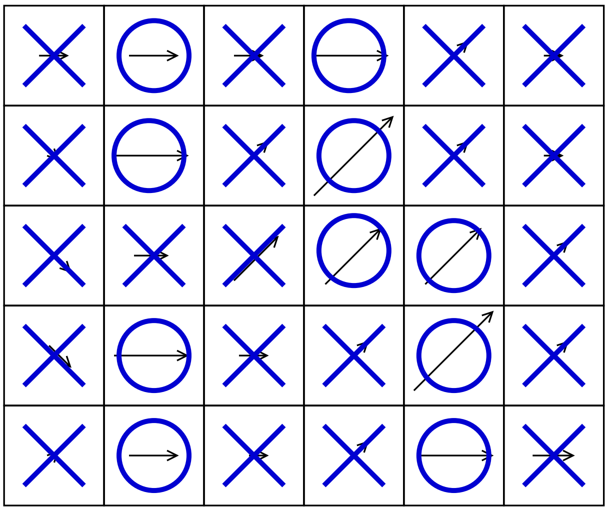
(b) Motion blur can be used as a method for data tuning to extend the training set and increase the variability in the training set. The trained network will be more robust with respect to motion blur. Also, the simulated motion blur could be used to train a neural network in order to compensate motion blur in real data.

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 illustrated as an arrow. Apply non-maxima suppression 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.



## Solution

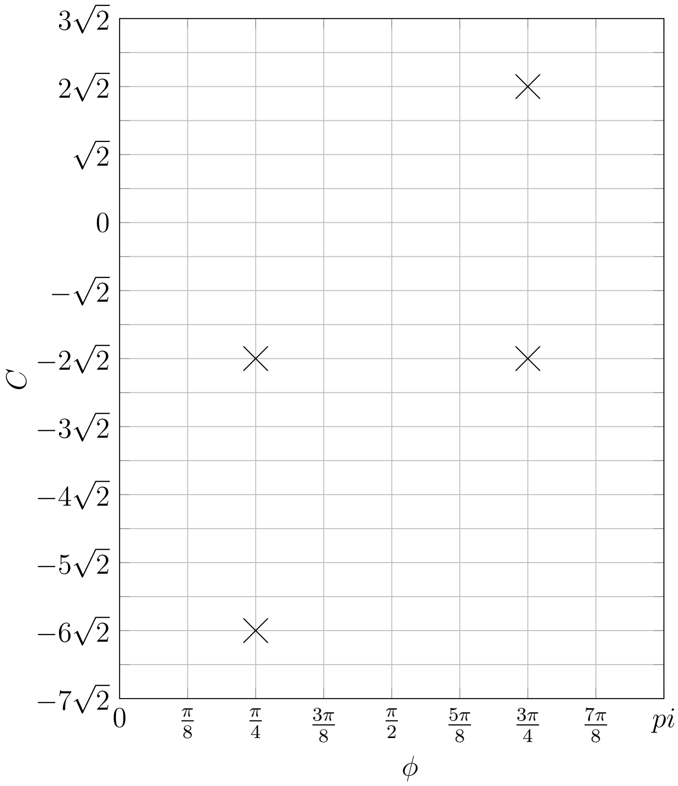


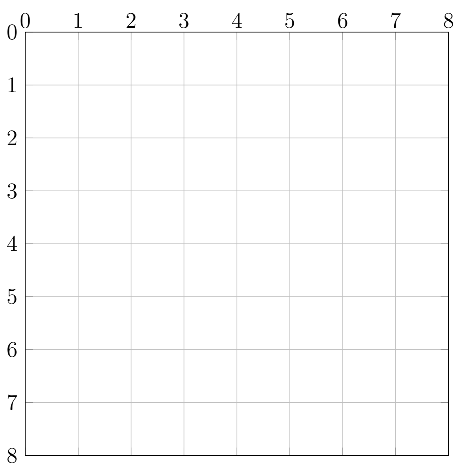
Question 3 (9 points)

In this task the Hough transform has been applied to the edge pixels in a certain image. The graph below shows the Hough space with four maxima (marked by crosses). Transform the maxima back into line equations. Then, draw the lines in the image below. Which geometric shape does the image show? Mark the segment in the image below. Mind that the origin of the image is in its left upper corner, that an angle of 0 points to the right, and an angle of $\frac{\pi}{2}$ points downwards.

*Remark:*

| $\theta$ | $0$ | $\frac{\pi}{8}$ | $\frac{\pi}{4}$ | $\frac{3\pi}{8}$ | $\frac{\pi}{2}$ | $\frac{5\pi}{8}$ | $\frac{3\pi}{4}$ | $\frac{7\pi}{8}$ |
|---|---|---|---|---|---|---|---|---|
| $\sin(\theta)$ | $0$ | $\frac{\sqrt{2-\sqrt{2}}}{2}$ | $\frac{\sqrt{2}}{2}$ | $\frac{\sqrt{2+\sqrt{2}}}{2}$ | $1$ | $\frac{\sqrt{2+\sqrt{2}}}{2}$ | $\frac{\sqrt{2}}{2}$ | $\frac{\sqrt{2-\sqrt{2}}}{2}$ |
| $\cos(\theta)$ | $1$ | $\frac{\sqrt{2+\sqrt{2}}}{2}$ | $\frac{\sqrt{2}}{2}$ | $\frac{\sqrt{2-\sqrt{2}}}{2}$ | $0$ | $-\frac{\sqrt{2-\sqrt{2}}}{2}$ | $-\frac{\sqrt{2}}{2}$ | $-\frac{\sqrt{2+\sqrt{2}}}{2}$ |



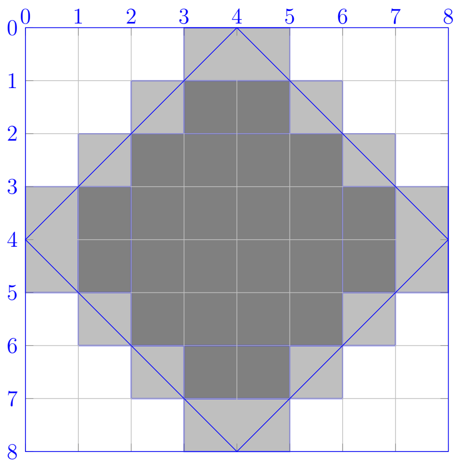
## Solution

The four line equations are

$$\begin{aligned}
\frac{1}{2}\sqrt{2}\cdot u+\frac{1}{2}\sqrt{2}\cdot v-6\sqrt{2} &= 0\\
\frac{1}{2}\sqrt{2}\cdot u+\frac{1}{2}\sqrt{2}\cdot v-2\sqrt{2} &= 0\\
-\frac{1}{2}\sqrt{2}\cdot u+\frac{1}{2}\sqrt{2}\cdot v-2\sqrt{2} &= 0\\
-\frac{1}{2}\sqrt{2}\cdot u+\frac{1}{2}\sqrt{2}\cdot v+2\sqrt{2} &= 0
\end{aligned}$$

which can be simplified by multiplying each equation with $\sqrt{2}$. We obtain

$$\begin{aligned}
u+v &= 12\\
u+v &= 4\\
-u+v &= 4\\
-u+v &= -4
\end{aligned}$$




The image shows a diamond.

Question 4 (8 points)

We want to use the random field approach to segment an image into foreground and background. The object in the foreground should be assigned to $l = 1$ and the background to $l = 0$. We assume that the foreground object is located in the bottom left corner of the image.

(a) Model the unary potential function $\phi_{prior}$ as a function of the image coordinates ($u$,$v$) and the width and height of the input image in pixels.

(b) Implement the unary potential function $\phi_{prior}$ in the MATLAB function template below to model this general preference. The input variables $u$ and $v$ denote the pixel coordinate, $l$ denotes the label, that is assigned to this pixel. In addition, the variables *width* and *height* denote the width and height of the image. The output variable *phi_prior* should return the result of the potential function $\phi_{prior}$.

```
function [ phi_prior ] = general_preference( u, v, l, width, height )
```

**Solution**

```
function [ phi_prior ] = general_preference( u, v, l, width, height )
   phi_prior = (( 1 - u ).^2 + ( height - v ).^2 ) / ...
               (( 1 - width ).^2 + ( 1 - height ).^2 );

   if ( l ~= 1 )
     phi_prior = 1 - phi_prior;
   end
end
```

Question 5 (8 points)

Assume a pinhole camera that is described by its matrix of intrinsic parameters

$$A = \begin{pmatrix} 600 & 0 & 500 \\ 0 & 600 & 400 \\ 0 & 0 & 1 \end{pmatrix}$$

As stated in the lecture, a set of parallel lines, that is observed by the camera is mapped onto a set of lines in the image that intersect at a single point, the so-called *vanishing point*. Calculate the vanishing point for the given camera and a set of observed parallel lines that are described in the camera coordinate system by their direction vector $(\Delta x, \Delta y, \Delta z) = (2, -2, 4)$. Explain your calculations briefly.

*Remark:* Mind that one of the parallel lines is mapped to a single point in the image.

## Solution

- *First and simplest way to solve the task:*
  Among the set of parallel lines we use the line that passes through the focal point of the camera. It is mapped in the image onto a single point which necessarily is equal to the vanishing point. The point can be obtained by mapping an arbitrary point on that line into the image, e.g. the point $(x,y,z) = (2,-2,4)$. Hence, we obtain

$$4 \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = A \cdot \begin{pmatrix} 2 \\ -2 \\ 4 \end{pmatrix} = \begin{pmatrix} 3200 \\ 400 \\ 4 \end{pmatrix} \quad \Rightarrow \quad u = 800, v = 100$$

- *Another way to solve the task:*
  The points on one of the parallel lines can be represented as

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + \tau \begin{pmatrix} 2 \\ -2 \\ 4 \end{pmatrix}$$

  with $(x_1, y_1, z_1)$ an arbitrary point on the line and $\tau \in \mathbb{R}$. Such a point is mapped into the image by

$$\lambda \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = A \cdot \left( \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + \tau \begin{pmatrix} 2 \\ -2 \\ 4 \end{pmatrix} \right) = A \cdot \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + \tau \cdot A \cdot \begin{pmatrix} 2 \\ -2 \\ 4 \end{pmatrix}$$

  If we devide both sides of the equation by $\tau$ we obtain

$$\lambda' \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \frac{1}{\tau} \cdot A \cdot \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + A \cdot \begin{pmatrix} 2 \\ -2 \\ 4 \end{pmatrix}$$

  where $\lambda' = \frac{\lambda}{\tau}$. Obviously, the influence of the point $(x_1,y_1,z_1)$ vanishes for $\tau \to \pm\infty$. In the limit, we obtain

$$\lambda' \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = A \cdot \begin{pmatrix} 2 \\ -2 \\ 4 \end{pmatrix} = \begin{pmatrix} 3200 \\ 400 \\ 4 \end{pmatrix} \quad \Rightarrow \quad u = 800, v = 100$$

- *A third way:*
  We construct the image of two of those parallel lines and afterwards calculate the point of intersection. The first line is given by the points $\vec{p}_1 = (1,0,1)^T$ and $\vec{p}_2 = (3,-2,5)^T$. The second line is given by the points $\vec{p}_3 = (0,1,1)^T$ and $\vec{p}_4 = (2,-1,5)^T$. The image coordinates of the projected points are

  $$(u_1,v_1) = (1100, 400),\ (u_2,v_2) = (860, 160), (u_3,v_3) = (500,1000), (u_4,v_4) = (740, 280)$$

  Now, we calculate the point of intersection $\vec{q}$ of the lines in image coordinates

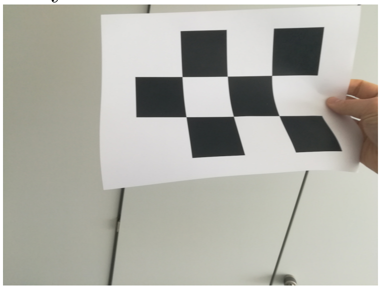
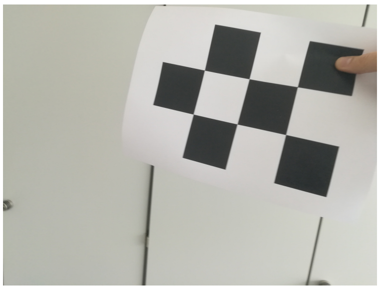
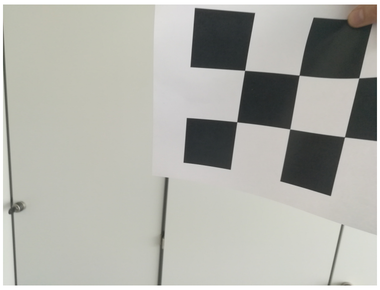
  $$\vec{q} = (1100, 400) + \tau \cdot (-240, -240) = (500, 1000) + \rho \cdot (240, -720)$$

  Hence, we obtain a system of linear equations for $\tau$ and $\rho$ which can be solved easily, e.g. with Cramer's rule or Gaussian elimination. We obtain

  $$\tau = \rho = \frac{5}{4} \quad \Rightarrow \quad \vec{q} = (1100, 400) + \frac{5}{4} \cdot (-240, -240) = (800, 100)$$

Question 6 (6 points)

Assume a new collaborator of yours wants to make his first camera calibration with Zhang's approach. For that, he printed out a chessboard pattern on a sheet of paper, holds the sheet in front of the camera and makes the three pictures that are shown below. Then, he applies a calibration software to the three pictures and checks the results.
Unfortunately, the calibration results are of bad quality. Suggest three improvements in the calibration process. If you suggest more than three improvements only the first three count for your answer.

## Solution

Possible improvements could be:

- keep chessboard sheet planar (required by Zhang)
- make more than 3 pictures (3 is the absolute minimum)
- show the chessboard at different poses and distances
- show the chessboard in different areas of the image, not just in the right upper corner
- don't let the chessboard be cropped by the image boundaries
- don't occlude the chessboard with your fingers
- use a chessboard with more patches

Question 7 (2+2+2 points)

For two training patterns $\vec{x} = (8, 2, 7, 3)^T$ and $\vec{y} = (1, 6, 0, 4)^T$ calculate the output of the Kernel function $K(\vec{x}, \vec{y})$ for the following three kernel functions $K$. In part (c) you may provide your result in the form $e^z$ with $z \in \mathbb{R}$.

(a) histogram intersection kernel

(b) dot product kernel

(c) radial basis function kernel with $\sigma = 5$

**Solution**

(a)

$$K(\vec{x}, \vec{y}) = \sum_{i=1}^{4} \min\{x_i, y_i\} = 1 + 2 + 0 + 3 = 6$$

(b)

$$K(\vec{x}, \vec{y}) = \sum_{i=1}^{4} x_i \cdot y_i = 8 + 12 + 0 + 12 = 32$$

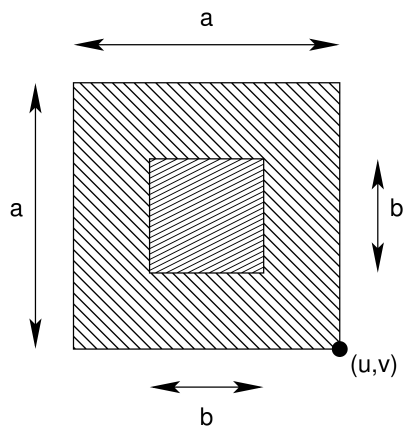
(c)

$$K(\vec{x}, \vec{y}) = e^{-\frac{||\vec{x} - \vec{y}||^2}{2\sigma^2}} = e^{-\frac{49+16+49+1}{50}} = e^{-\frac{115}{50}} = e^{-2.3}$$

Question 8 (8 points)

Consider the center-surround Haar feature shown below with edge length $a$ of the outer box and edge length $b < a$ of the inner box. $a$ and $b$ are both assumed to be even, positive numbers. The surrounding contributes positive to the Haar feature, the inner box contributes negative. The right lower pixel of the outer box is located at image position $(u,v)$.

Derive a formula that describes how to calculate the Haar feature value based on the integral image $I$. For that, first provide a formula based on the integral image that calculates the average gray value of the inner box. Then, provide a formula based on the integral image that calculates the average gray value of the surrounding. Finally, provide the formular to calculate the whole Haar feature. You may always reuse your previous results. At how many different positions do we have to evaluate the integral image?




**Solution**

To simplify the notation let $c = \frac{a-b}{2}$.

The inner box has area $b^2$. We obtain its sum of gray values as

$$B = I(u-c, v-c) + I(u-c-b, v-c-b) - I(u-c-b, v-c) - I(u-c, v-c-b)$$

Hence, the average gray value of the inner box is $\frac{B}{b^2}$.

The surrounding has area $a^2 - b^2$ since the inner box does not contribute to the surrounding. We obtain the sum of gray values of the large box as

$$A = I(u,v) + I(u-a, v-a) - I(u-a, v) - I(u, v-a)$$

Hence the average gray value of the surrounding is $\frac{A-B}{a^2-b^2}$.

Hence, the overall value of the Haar feature is

$$\frac{A-B}{a^2-b^2} - \frac{B}{b^2} = \frac{1}{a^2-b^2}A - \frac{a^2}{(a^2-b^2)b^2}B$$

We have to evaluate $I$ at 8 different positions.

Question 9 (4 points)

Review the following four statements and determine whether they are correct or incorrect. Correct the statements if necessary.

(a) A neural network trained on semantic labeling provides the contours of each object instance from the input image.

(b) ResNet layers enable deeper network architectures by preserving information of the input layers.

(c) In a generative adversarial network (GAN) the generative network tries to replicate real images, while the discriminative network shows the differences between the real and generated images.

(d) A gated recurrent unit has fewer parameters than the long short term memory unit.

**Solution**

(a) Incorrect. Semantic labeling does not provides the contours of each object instance.

(b) Correct.

(c) Incorrect. The generative network creates new images, while the discriminative network distinguished if images are real or generated.

(d) Correct.

Gesamtpunkte: 60