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

Solutions for exam "Machine Vision" August 3, 2020

Question 1 (4 points)

Create a 4-by-4 binomial filter mask.

Solution

The binomial coefficients are $\binom{3}{0} = \binom{3}{3} = 1$, $\binom{3}{1} = \binom{3}{2} = 3$. The sum of all entries is 8. Hence we obtain

$$\frac{1}{8}\begin{bmatrix}1\\3\\3\\1\end{bmatrix} * \frac{1}{8}\begin{bmatrix}1 & 3 & 3 & 1\end{bmatrix} = \frac{1}{64}\begin{bmatrix}1 & 3 & 3 & 1\\3 & 9 & 9 & 3\\3 & 9 & 9 & 3\\1 & 3 & 3 & 1\end{bmatrix}$$

Question 2 (3+2+2 points)

We want to follow the ideas of the Hough transform to detect circles from a set of edge pixels. For each edge pixel we want to create the set of all circles that contain this pixel on its boundary and increment accumulator cells in paramter space accordingly. We want to represent a circle by its center and radius.

(a) Assume, we consider an edge pixel at image position (u,v). At which positions might the corresponding circle center $\vec{m} = (m_u, m_v)$ be located if we know the true radius r? Describe the positions as precise as possible.

(b) Extend your solution from part (a) and describe the three-dimensional geometric shape of those parameter sets (r, m_u, m_v) which are supported by edge pixel (u,v).

(c) How many accumulator cells do we need if we want to represent radii $r \in [0,100)$, and circle centers $\vec{m} \in [0,1000) \times [0, 800)$ with cell sizes of $10 \times 10 \times 10$ pixels?

Remark: Part (c) can be solved independently of part (a) and (b)

Solution

(a) Since point (u,v) must be located on a circle with radius r, the circle center $\vec{m}$ must be located on a circle around (u,v) with radius r.

(b) Since r can take on positive numbers, the parameter set (r, m_u, m_v) has the shape of a cone. Its axis of symmetry is parallel to the r-axis and its apex is at $(r,m_u,m_v) = (0,u,v)$.

(c) In r-direction we need $\frac{100}{10} = 10$ bins, in m_u-direction $\frac{1000}{10} = 100$ bins, in m_v-direction $\frac{800}{10} = 80$ bins. Hence, we need $10 \cdot 100 \cdot 80 = 80000$ accumulator cells.

Question 3 (5 points)

The message in the image below is hard to read due to low contrast. Implement a function in MATLAB that increases the contrast by brightening the graylevels in the image. The function should take an 8-bit encoded graylevel image as input and return an 8-bit encoded graylevel image of the same size. Black pixels should remain black while the gray values of gray pixels should be doubled.

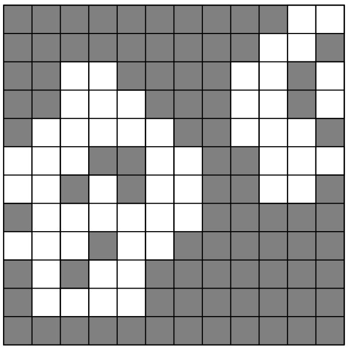
Solution

```
function light_text = recolor(dark_text)
   light_text = 2*dark_text;
end
```

Question 4 (9 points)

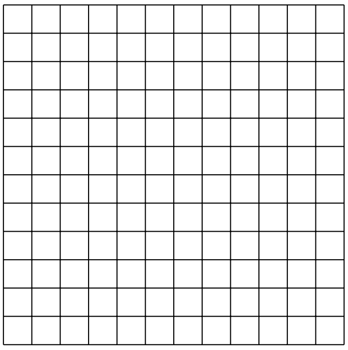
The Image I shows two white segments in front of darker background pixels. Assume that the image consists only of graylevels between 0 (= black) to 255 (= white). Name the two operations which, consecutively applied, can be used to close holes and simultaneously preserve the dimensions of segments. Then, execute the operations consecutively on the image I by filling out the resulting images I^* and I^{**}. Use a 4-neighborhood of pixels. How is this specific combination of morphological operations named? Why does image I^{**} differ from the original image I apart from the fact that the holes are filled?

I :



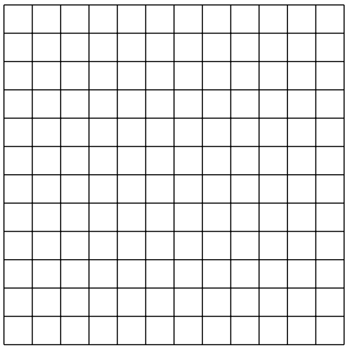
(a) input image

I^* :



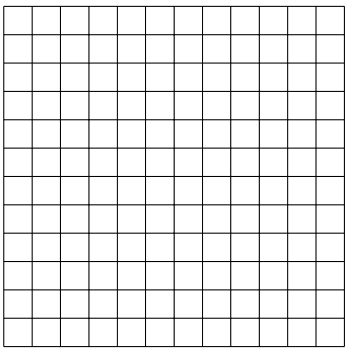
(b) first operation. Enter your solution here!

I^* :



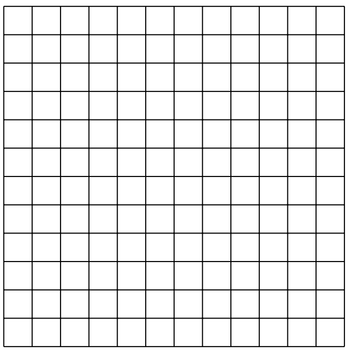
(c) spare image just in case you need it (e.g. if you want to revise your solution)

I^{**} :



(d) second operation. Enter your solution here!

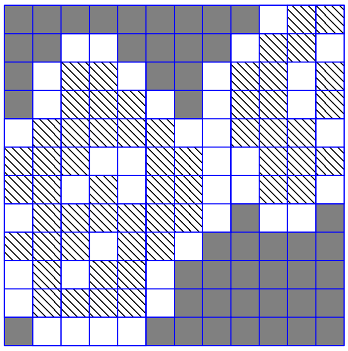
I^{**} :



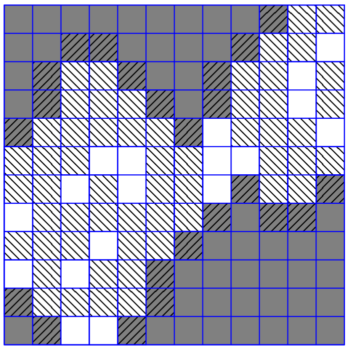
(e) spare image just in case you need it (e.g. if you want to revise your solution)

Solution

$I^* :$



$I^{**} :$



$$I \xrightarrow{\text{dilation}} I^*$$

$$I^* \xrightarrow{\text{erosion}} I^{**}$$

$$dilation + erosion \equiv closing$$

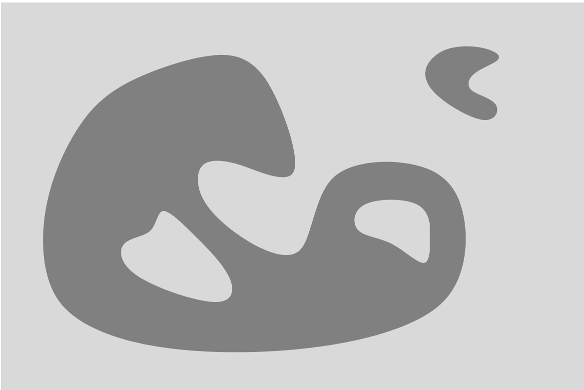
Due to the dilation both segments were fused to a single segment. This cannot be undone by the erosion.
Both segments are located at the image border. This leads to an extension of the segments area.

Question 5 (7 points)

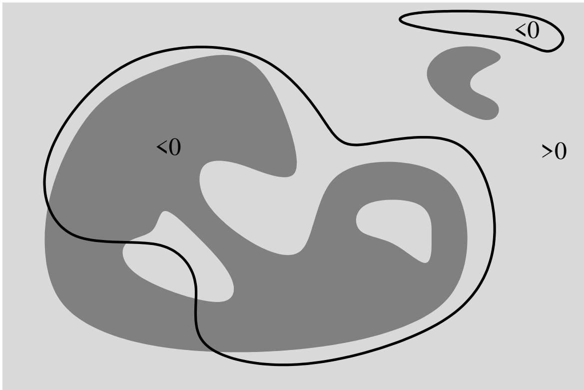
Picture (a) below shows a graylevel image with dark gray foreground objects on light gray background. Picture (b) shows an initial level set segmentation for image (a). The black line shows all pixels with $\phi(u,v) = 0$ where ϕ denotes the signed distance function. The labels `<0` and `>0` indicate the sign of ϕ in the respective areas.

We apply level set evolution using the Mumford-Shah approach without boundary rectification ($\beta = 0$), without general shrinking/expanding ($\alpha = 0$) and with balanced parameters for foreground and background ($\lambda_1 = \lambda_2 > 0$). Enter in picture (c) the boundary of the level set segmentation after convergence. Use a solid line to mark the boundary. Enter labels `>0` and `<0` to mark the sign of the signed distance function after convergence.

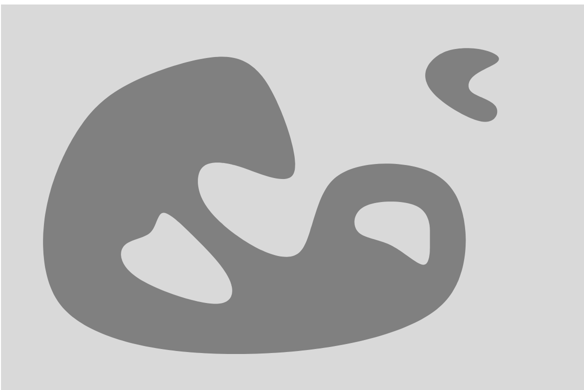
Draw carefully! If you want to revise your solution, cross out picture (c) and use (d) instead.



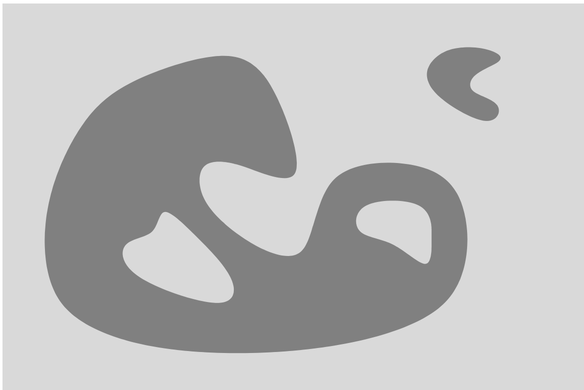
(a) input image



(b) ininitial segmentation

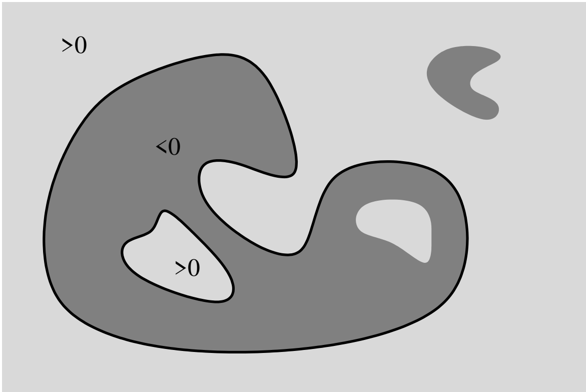


(c) segmentation after convergence. Enter your solution here!



(d) spare image just in case you need it (e.g. if you want to revise your solution)

Solution


>0
<0
>0

Question 6 (7 points)

A camera calibration yielded intrinsic parameters which are given in camera matrix $\mathbf{A}$ below as well as extrinsic parameters that describe the pose of the camera relative to the world coordinate system by rotation matrix $\mathbf{R}$ and translation vector $\vec{t}$.

For one of the calibration markers we know its image coordinates $\vec{p}$ as well as its 3-dimensional world coordinates $\vec{q}$. Calculate the reprojection error in image space using the known position of the calibration markers in image and world coordinates. Use the Euclidean norm to calculate the reprojection error.

$$\mathbf{A} = \begin{pmatrix} 600 & 0 & 300 \\ 0 & 600 & 150 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}, \quad \vec{t} = \begin{pmatrix} -0.5 \\ 1 \\ 1.5 \end{pmatrix}$$

$$\vec{p} = \begin{pmatrix} 184 \\ 387 \end{pmatrix}, \quad \vec{q} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

Solution

$$\begin{pmatrix} x & y & z \end{pmatrix}^\intercal = \begin{pmatrix} \mathbf{R} & | & \vec{t} \end{pmatrix} \begin{pmatrix} \xi & \eta & \zeta & 1 \end{pmatrix}^\intercal = \begin{pmatrix} \mathbf{R} & | & \vec{t} \end{pmatrix} \begin{pmatrix} \vec{q_n} & 1 \end{pmatrix}^\intercal$$

$$\tilde{\vec{p}} = \begin{pmatrix} u & v & 1 \end{pmatrix}^\intercal = \frac{1}{z} \mathbf{A} \begin{pmatrix} x & y & z \end{pmatrix}^\intercal$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & -0.5 \\ 0 & 0 & -1 & 1 \\ 0 & 1 & 0 & 1.5 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -0.5 \\ 1 \\ 2.5 \end{pmatrix}$$

$$\tilde{\vec{p}} = \begin{pmatrix} \tilde{u} \\ \tilde{v} \\ 1 \end{pmatrix} = \frac{1}{2.5} \begin{pmatrix} 600 & 0 & 300 \\ 0 & 600 & 150 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} -0.5 \\ 1 \\ 2.5 \end{pmatrix} = \begin{pmatrix} 180 \\ 390 \\ 1 \end{pmatrix}$$

$$error = \|\vec{p} - \tilde{\vec{p}}\| = \sqrt{4^2 + 3^2} = 5$$

Question 7 (9 points)

Assume a hard margin support vector machine (SVM) with kernel function $k(x,y) = x^2 \cdot y^2$, and the support vectors provided in the table below.

support vector x	class label d	Lagrange multiplier α
-3	$+1$	$\frac{2}{50}$
2	-1	$\frac{4}{50}$
3	$+1$	$\frac{2}{50}$

Calculate the output of the support vector machine for input pattern $x = 1$. Write down intermediate results as well. A drawing is not sufficient in this task to obtain full points.

Solution

There are two possible ways to solve the task.

Version 1:

We can easily derive the nonlinear mapping Φ from which the kernel is generated. It is $\Phi(x) = x^2$. Hence, we can directly do all calculations in feature space. Hence we get:

$$\begin{aligned}
w &= \sum_i \alpha_i d^{(i)} \Phi(x^{(i)}) = \frac{2}{50} \cdot 9 - \frac{4}{50} \cdot 4 + \frac{2}{50} \cdot 9 = \frac{2}{5} \\
b &= d^{(1)} - \Phi(x^{(1)}) \cdot w = +1 - 9 \cdot \frac{2}{5} = -\frac{13}{5} \\
y(1) &= sign(1 \cdot w + b) = sign(\frac{2}{5} - \frac{13}{5}) = -1
\end{aligned}$$

Version 2:

We apply the kernel instead of deriving Φ explicitly. We obtain:

$$\begin{aligned}
b &= d^{(1)} - (\alpha_1 d^{(1)} k(x^{(1)}, x^{(1)}) + \alpha_2 d^{(2)} k(x^{(2)}, x^{(1)}) + \alpha_3 d^{(3)} k(x^{(3)}, x^{(1)})) \\
&= +1 - (\frac{2}{50} \cdot 81 - \frac{4}{50} \cdot 36 + \frac{2}{50} \cdot 81) = +1 - \frac{18}{5} = -\frac{13}{5} \\
y(1) &= sign(\alpha_1 d^{(1)} k(1, x^{(1)}) + \alpha_2 d^{(2)} k(1, x^{(2)}) + \alpha_3 d^{(3)} k(1, x^{(3)}) + b) \\
&= sign(\frac{2}{50} \cdot 9 - \frac{4}{50} \cdot 4 + \frac{2}{50} \cdot 9 - \frac{13}{5}) = sign(\frac{4}{50} \cdot 5 - \frac{13}{5}) = sign(-\frac{11}{5}) = -1
\end{aligned}$$

Question 8 (6 points)

Assume we apply AdaBoost on a classification task and obtain an ensemble of six classifiers, $c_1, \ldots, c_6$. The classifier weights β_i are all equal. For eight training examples we observed the classification results of the six individual classifiers given in the table below. For which of the eight training examples are the pattern weights γ_i small after six iterations of AdaBoost, for which training examples are the pattern weights large, and for which are the pattern weights medium size?

example	class label	output classifier					
		c_1	c_2	c_3	c_4	c_5	c_6
1	+1	+1	+1	+1	+1	+1	+1
2	+1	+1	+1	-1	-1	+1	-1
3	+1	-1	-1	-1	-1	-1	-1
4	+1	-1	+1	+1	-1	+1	-1
5	-1	+1	-1	-1	+1	-1	+1
6	-1	-1	-1	-1	-1	-1	-1
7	-1	+1	+1	+1	+1	+1	+1
8	-1	+1	-1	-1	+1	+1	-1

Solution

Large pattern weights: examples 3 and 7 since they are misclassified by all classifiers: Their pattern weight is proportional to $e^{6\beta_1}$

Small pattern weights: examples 1 and 6 since they are classified correctly by all classifiers. Their pattern weight is proportional to $e^{-6\beta_1}$

Median size pattern weights: examples 2, 4, 5, 8 since they are misclassified by three and classified correctly by another three classifiers. Their pattern weight is proportional to e^0.

Question 9 (3+3 points)

In deep learning a *region proposal network* is used to predict the position and size of a bounding box and the class of the object that is inside of the bouning box. Assume that the region propsal network should return the output values $u \in \mathbb{R}$ and $v \in \mathbb{R}$ (pixel position of the left upper corner of the bounding box), $w > 0$ and $h > 0$ (width and height of the bounding box), $c_1, \ldots, c_q$ (class probabilities for q different classes).

(a) Which activation function should the output neurons of the region propsal network have? You may distinguish in your answer between the different output neurons. Justify your answer briefly.

(b) Which loss function should be used to train the region proposal network? You may distinguish in your answer between the different output neurons. Justify your answer briefly.

Solution

(a) w, h, u, and v are real numbers that might be larger than 1. Therefore, only unbounded activation functions (linear activation or ReLU) are possible. The c_i should be interpreted as class probabilities. Hence, the softmax activation function is suitable.

(b) Since w, h, u, and v provide real numbers and the task to estimate those values can be interpreted as a regression task the squared error is appropriate. The c_i outputs are probabilities so that the cross entropy error is appropriate.

Gesamtpunkte: 60