

CS 675 – Computer Vision – Fall 2007

Instructor: Marc Pomplun

Practice Exam Solutions

Duration: 2 hours and 30 minutes

Notice that in the actual exam, no calculators, no books, and no notes are allowed.

Question 1: ____ out of ____ points

Question 2: ____ out of ____ points

Question 3: ____ out of ____ points

Question 4: ____ out of ____ points

Question 5: ____ out of ____ points

Question 6: ____ out of ____ points

Question 7: ____ out of ____ points

Total Score:

Grade:

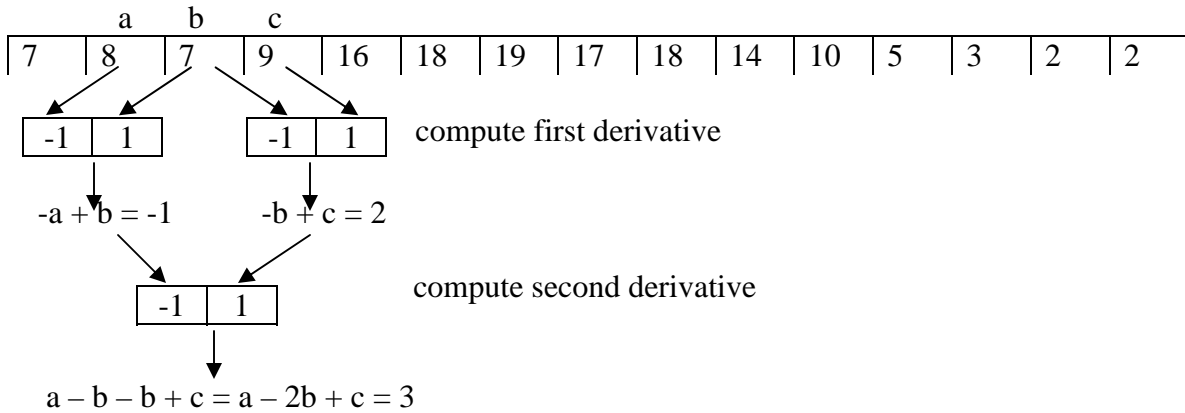
Question 1: True or False?

Tell whether each of the following statements is true or false by checking the appropriate box. Do not check any box if you do not know the right answer, because you will lose points for incorrect answers.

Statement	True	False
a) Image understanding is the highest-level problem in computer vision.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
b) Let us apply a size filter to an image A, resulting in image A', and then apply the same size filter to A', resulting in image A''. Then it is possible that $A' \neq A''$.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
c) The thinning algorithm can also be applied to three-dimensional structures.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
d) The k-means algorithm will always yield the same clustering result when started on the same dataset.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
e) Self-organizing maps perform unsupervised learning.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
f) When we move forward, the direction of optical flow in our visual field is always perpendicular to the gradient of its amplitude.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
g) Two different shapes, even if they cannot be turned into each other by rotation, can have identical signatures.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
h) In the Canny Edge Detector, non-maxima suppression has the effect of thinning the detected edges to a width of one pixel.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
i) Support Vector Machines can learn to classify objects based on a set of given sample objects for each class.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
j) In artificial neural networks, supervised learning is biologically more plausible than unsupervised learning.	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Question 2: Laplace and His Filter

(a) Let's just pick three consecutive pixels and call them a, b, and c:



So if we wanted to do this computation in one step, we could simply use the following filter – which is the desired Laplacian filter:

1	-2	1
---	----	---

(b + c)

0	-2	3	5	-5	-1	-3	3	-5	0	-1	3	1	1	0
		↑		↑		↑	↑			↑				

The edges are located at those positions where the curvature of the original function, i.e., its second derivative changes from positive to negative or vice versa. So, strictly speaking, the sequence -5, 0, -1 above does not contain an edge at the 0.

Question 3: Customizing the Hough Transform

First of all, you should smooth your camera image using a Gaussian convolution filter. Afterwards, you apply (for example) the Canny Edge Detector to derive an edge image. The hamburgers should now look like circles. You can therefore use a Hough transform that detects the centers of circles with radius r . The input space of this Hough transform is the two-dimensional edge image with parameters x and y , and the output space is also a two-dimensional space with parameters x' and y' , the center positions of circles in the input image.

Your transform works as follows: It iterates through all pixels (x, y) in the input image, and whenever it finds a pixel value greater than zero, it increases all relevant counters in the output space. The relevant ones are those that stand for centers (x', y') that are r pixels away from (x, y) , because if there were a circle centered at any of these (x', y') , we would expect an edge located at (x, y) . To be precise, the set of all counters that need to be increased is given by:

$$\{ (x', y') \mid (x' - x)^2 + (y' - y)^2 = r^2 \}$$

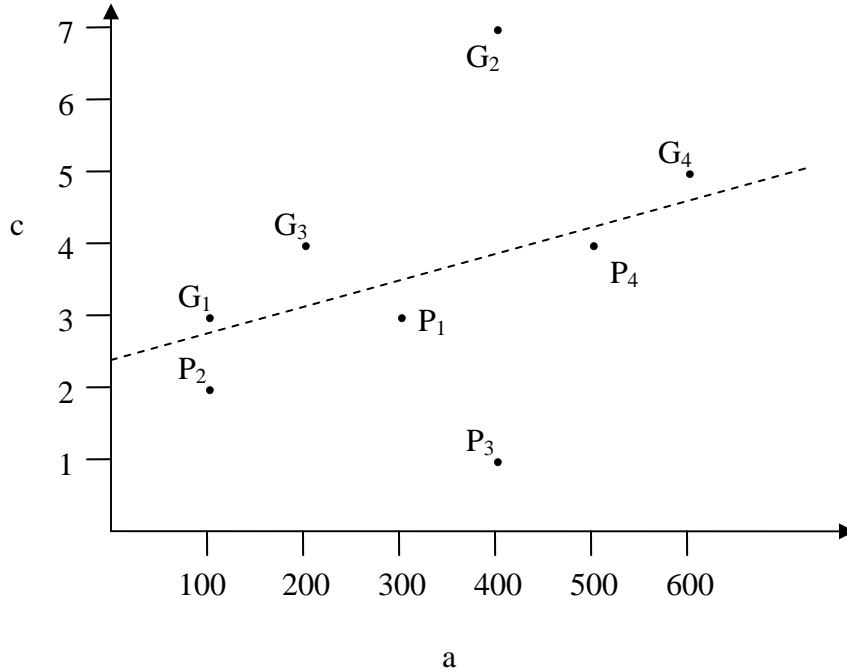
This increase could be simply a “+1” operation, or you could add the pixel value at (x, y) . Either way, the most likely hamburger centers should now be indicated by local peaks in the output space. So you could basically just count the local maxima to find the number of hamburgers in the original image. Local maxima are those pixels whose values are greater than or equal to all their neighbors’ values (usually 8-neighbors).

However, there are several possible problems here. First of all there could be substantial noise in the output data, which could cause you to find many more local maxima than there are hamburgers. One operation you definitely need to do to get rid of that noise is Gaussian smoothing with a rather large filter. In addition, you should set a detection threshold for local maxima, i.e., all local maxima whose values are below this threshold are disregarded. However, even then it could happen that two neighboring pixels have identical values and are counted as the centers of two hamburgers. This, of course, could only be correct if two hamburgers almost completely overlapped, which we do not assume to be possible. To avoid this problem, you can define a minimum distance between any two hamburger centers. This means that once you detected a center, there cannot be another center within that distance.

Another thing that needs to be considered is hamburgers whose centers are not in the image, but that are still partially visible. If you want to detect such partial hamburgers, you can simply expand your output space by r pixels in all four directions. Then, finally, you should be able to count hamburgers reliably well.

Question 4: Recognizing Objects

$G_1 = (100, 3)$, $G_2 = (400, 7)$, $G_3 = (200, 4)$, $G_4 = (600, 5)$,
 $P_1 = (300, 3)$, $P_2 = (100, 2)$, $P_3 = (400, 1)$, $P_4 = (500, 4)$.



The dashed line indicates a reasonable estimate for the optimal division between the two object classes. It is given by the following equation:

$$c = 0.004a + 2.3$$

$$f(a, c) = 0, \text{ if } c > 0.004a + 2.3 \\ = 1, \text{ otherwise}$$

Question 5: Errors in Depth Perception

Having two eyes, we are able to perceive the depth (z-distance) of an object through binocular disparity information. As you know, we can also give our computer vision system two cameras and let it do the same thing. The question is: How accurate is its estimation of depth, and on what factors does this accuracy depend?

The main problem here is the limited accuracy and resolution of the cameras. Let us say that the actual position of an object in the camera image may be up to one millimeter to the left or to the right of its actual position. For example, if the left camera measures $x_l = 5.3$ mm, it means that the objects' actual position could be anywhere between $x_l = 5.2$ mm and $x_l = 5.4$ mm.

- (a) Given this camera accuracy, what is the z-range that an object could have (i.e., the minimum and maximum z-distance possible), if the cameras with baseline $b = 10$ cm and focal length $f = 20$ cm measure positions $x_l = 6.1$ and $x_r = 5.1$? If you do not remember the formula, try to derive it; it is not very difficult.

The most extreme cases are $x_l = 6.0$ and $x_r = 5.2$ and $x_l = 6.2$ and $x_r = 5.0$. In the first case, we derive distance z as:

$$z = 10 \text{ cm} \cdot 20 \text{ cm} / 1.2 \text{ cm} = 166.67 \text{ cm}$$

In the second case, we get:

$$z = 10 \text{ cm} \cdot 20 \text{ cm} / 0.8 \text{ cm} = 250 \text{ cm}$$

The z-distance ranges from 166.67 to 250 cm.

- (b) What do you think will happen if the object is much further away from the system? Will the error in z-distance measurement (i.e., the z-range) increase or decrease? Why?

The z-distance error will increase, because will greater distance the same variation in the camera image corresponds to greater variation in the z-distance measurement (it can also be seen from the equations above).

- (c) What do you think will happen if we keep the object in the same place as in (a), but increase the distance between the cameras, i.e., the baseline b ? Will the error in z-distance measurement (i.e., the z-range) increase or decrease? Why?

Now the z-distance error will decrease, because the same variation in the camera image corresponds to smaller variation in the z-distance measurement. It's harder to see in the

equations this time, because the baseline, the focal length, and the difference between x_l and x_r will change. But do the math, it's a great exercise.

(d) If we increase the baseline, however, another problem becomes more and more difficult. What problem is that?

The stereo matching, i.e., determining the correspondence between points in the left and the right image, becomes more difficult. This is because we are now looking at objects from more distinct angles, and so the two images of the same object look less similar.

Question 6: Teaching the Artificial Brain

Describe the basic principles underlying supervised learning in artificial neural networks. Use at least two paragraphs to outline the most important ideas.

To teach the network to compute a function, we need to provide information in the form of exemplars, i.e., desired input-output pairs. If, for example, the network is supposed to output "1" for the input (1, 0, 1), then the corresponding exemplar is ((1, 0, 1), 1).

Then we randomly pick one of the exemplars and feed its input part into the input layer neurons of our network. The network then performs all its computations, i.e., each neuron multiplies each of its inputs with the corresponding synaptic weight and applies the output function, usually a sigmoid function, to determine its output. Then we compare the network's output with the desired output specified in the exemplar. If there is a discrepancy, then we use a rule (e.g., the backpropagation learning rule) to change the weights in such a way that next time the same input is presented, the output will be closer to the desired one. Such rules typically use gradient-descent, i.e., they move against the error gradient in weight space.

After each exemplar was shown to the network and its weights changed accordingly, one epoch is finished. If the overall network error, i.e., the deviation between desired and actual output values, is still larger than a certain threshold, we perform another epoch, and so on, until the error falls below the threshold. Then the network should have learned the function and be able to predict appropriate output values even for inputs that it has never seen before.

(This text is more detailed than what you would need to write in order to get the full score.)

Question 7 (Bonus Question): Tricking the Difference Image Technique

In difference images, pixel values greater than zero (or greater than a certain threshold) are thought to reflect moving objects in the scene. However, this does not always have to be the case. There are other factors besides object motion that can cause non-zero values in a difference image. List as many such factors as you can think of.

- Change in lighting
- Noise in the images varying over time
- Motion of the camera

