02503 Image Analysis Exam Fall 2025

02503 Image Analysis Exam Fall 2025

Written exam, December 12, 2025

Course name: Image Analysis

Course number: 02503 and 02502 re-exam

Number of Questions: 25

Aids allowed: All aids allowed.

Duration: 4 hours.

Weighting: All questions are equally weighted.

Notes: There are five possible answers to each question and a "do not know" option. A correct answer will be equivalent to 5 points. An incorrect answer will be equivalent to -1 points. Questions unanswered (equivalent to "do not know") will not produce points. The final grade is determined by the examiners.

Appendix: Remember to submit your code (Python files, Notebooks, PDF, AI prompts or similar) to the "mellemregninger "/appendix part of the exam.

Data: All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b)

Candy sorting

You are organizing a social event in December and asked a friend to buy a lot of mixed candy. Unfortunately, your friend did not know that you cannot stand blue M&Ms. Therefore you are designing a computer vision system to identify the blue M&M candy, so you can remove them.

To test your system, you have taken a photo with mixed candy. It is an RGB image and it is called "candy.jpg".

All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b) .

You start your analysis by converting the image to gray scale using rgb2gray from scikit-image. Secondly, you use a threshold of 0.6 and set all pixels with a value below to foreground (1) and the rest of the pixel to background (0). How many foreground pixels are there in this binary image?

Choose one answer

- ○ Do not know
- ○ Between 400.000 and 450.000
- ◉ Between 550.000 and 600.000
- ○ Between 600.000 and 650.000
- ○ Between 500.000 and 550.000
- ○ Between 450.000 and 500.000

For comparison, you also compute a threshold using Otsu's method. What is this threshold? (Do not use this threshold in the future computations)

Choose one answer

- ○ Between 0.30 and 0.40
- ◉ Between 0.40 and 0.50
- ○ Between 0.50 and 0.60
- ○ Between 0.10 and 0.20
- ○ Between 0.20 and 0.30
- ○ Do not know

After creating the binary image (with threshold 0.6) you clean the image by first doing a morphological closing with a disk-shaped structuring element with radius 3 followed by a morphological erosion with a disk-shaped structuring element with radius 6.
After cleaning, you do a BLOB analysis of the binary image. How many BLOBs do you find?

Choose one answer

- ○ Between 25 and 30
- ◉ Between 35 and 40
- ○ Between 40 and 45
- ○ Between 30 and 35
- ○ Between 20 and 25
- ○ Do not know

To be able to classify the candy, you compute a series of values per BLOB. They are (per BLOB):

1. The average R value of the pixels inside the BLOB sampled from the original image values
2. The average G value of the pixels inside the BLOB sampled from the original image values
3. The average B value of the pixels inside the BLOB sampled from the original image values
4. The standard deviation of the R values of the pixels inside the BLOB sampled from the original image values
5. The standard deviation of the G values of the pixels inside the BLOB sampled from the original image values
6. The standard deviation of the B values of the pixels inside the BLOB sampled from the original image values
7. The area (in pixels) of the BLOB
8. The perimeter of the BLOB
9. The circularity of the BLOB computed as (2 * sqrt(pi * area)) / perimeter)

You start by trying to simply set a threshold to classify based on the average B value of the BLOB. You find the threshold as the 0.75 quantile of all the sampled B mean values. What is this threshold?

Choose one answer

- ○ Do not know
- ○ Between 65 and 75
- ○ Between 75 and 85
- ○ Between 105 and 115
- ○ Between 95 and 105
- ○ Between 85 and 95

You manually select a B threshold of 65 and classify all BLOBs with an average B value above this value to be "blue M&M". How many false positives do you get?

Choose one answer

- ○ Do not know
- ○ Between 6 and 8 (inclusive)
- ○ Between 3 and 5 (inclusive)
- ○ Between 13 and 15 (inclusive)
- ○ Between 0 and 2 (inclusive)
- ○ Between 9 and 12 (inclusive)

To use the combined set of features, you want to do a principal component analysis (PCA). To do that you start by gathering the nine measured features for each BLOB into a data matrix, where one row is the features from on BLOB. Secondly, you subtract the mean from each feature and divide by the standard deviation of the feature.

You use the functions **cov** and **linalg.eig** from **Numpy** to compute the Eigenvectors and the Eigenvalues of the data matrix.

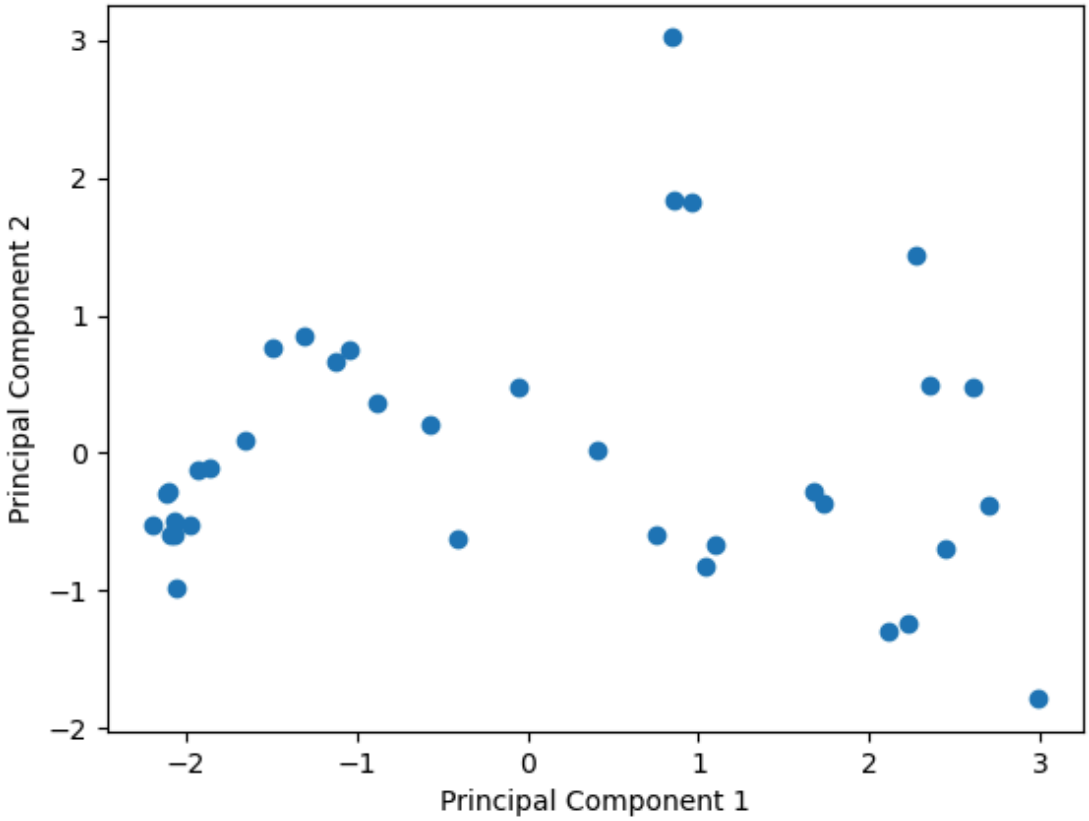
How much of the total variation is explained by the first three principal components?

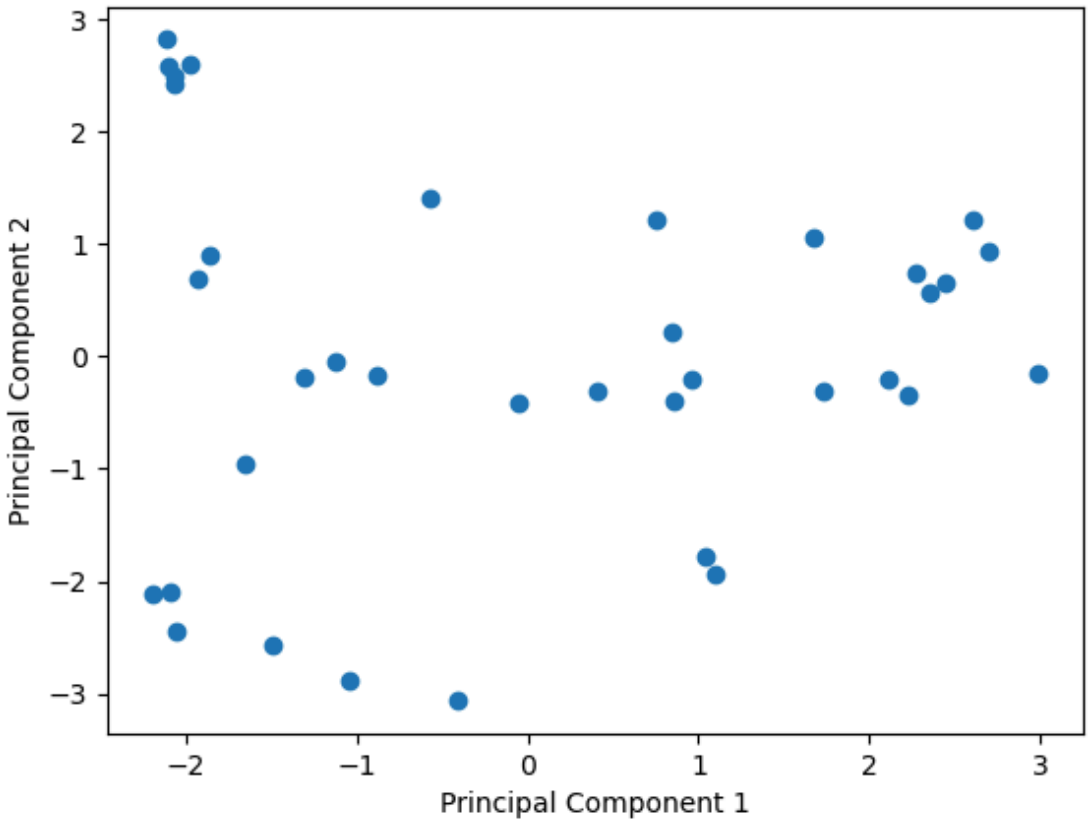
Choose one answer

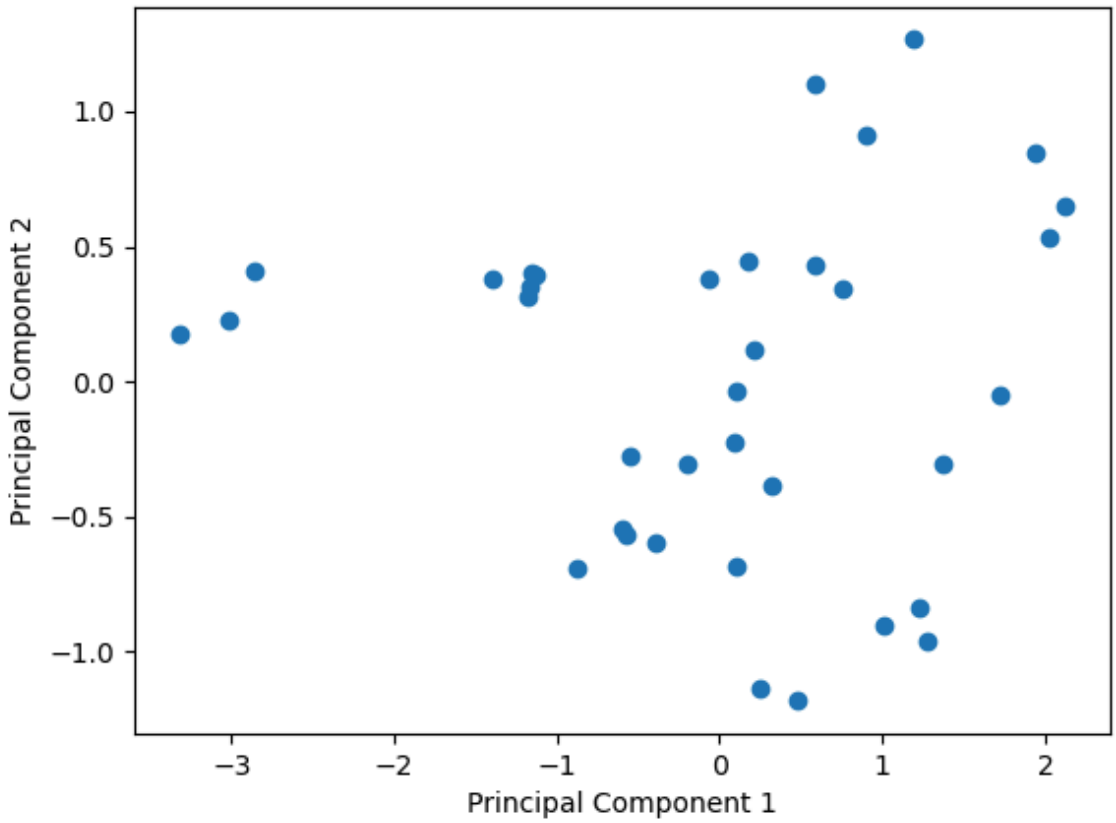
- Between 80 and 85
- Between 75 and 80
- Do not know
- Between 90 and 95
- Between 95 and 100
- Between 85 and 90

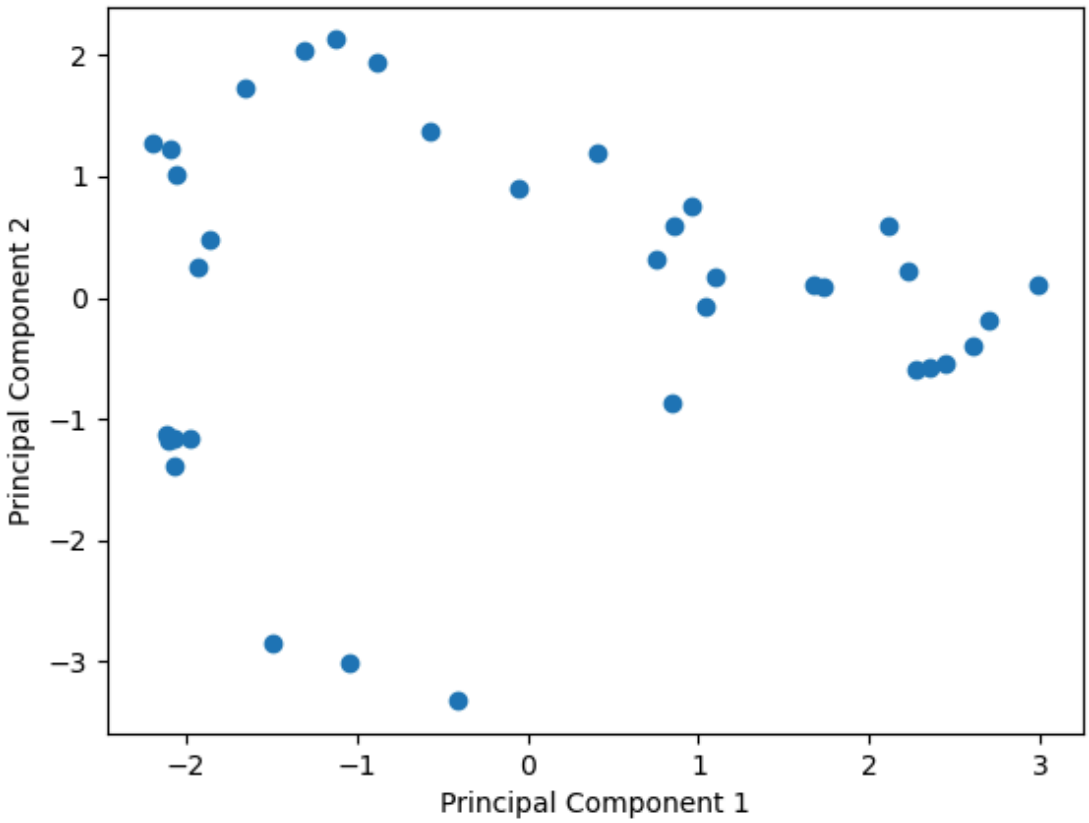
To get a feeling of what can be learned using PCA, you project the original data onto the principal components and plot the position of the samples in PCA space. You start by visualizing the positions on the first two principal components. Which figure corresponds to this plot (it might be flipped up-down and/or left-right)?

Choose one answer

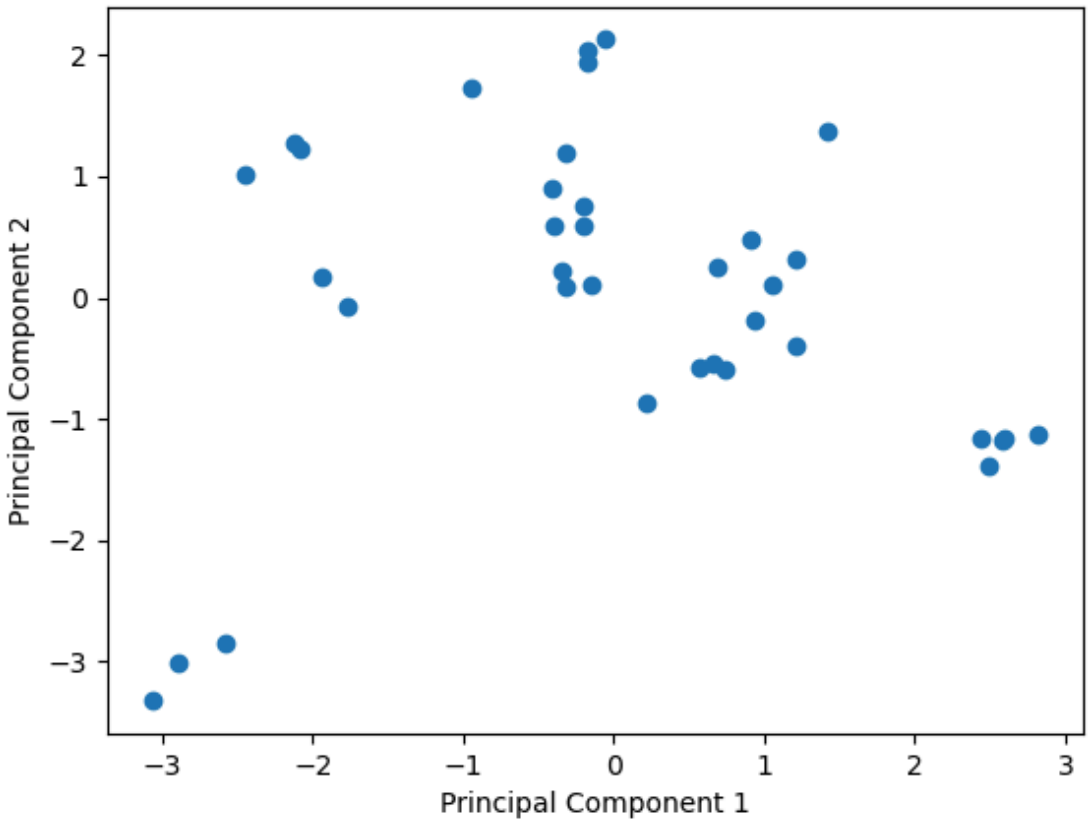



Principal Component 2
Principal Component 1
3
2
1
0
−1
−2
−3
−2
−1
0
1
2
3


Principal Component 2
Principal Component 1
1.0
0.5
0.0
−0.5
−1.0
−3
−2
−1
0
1
2


Principal Component 2
Principal Component 1
2
1
0
−1
−2
−3
−2
−1
0
1
2
3

Do not know

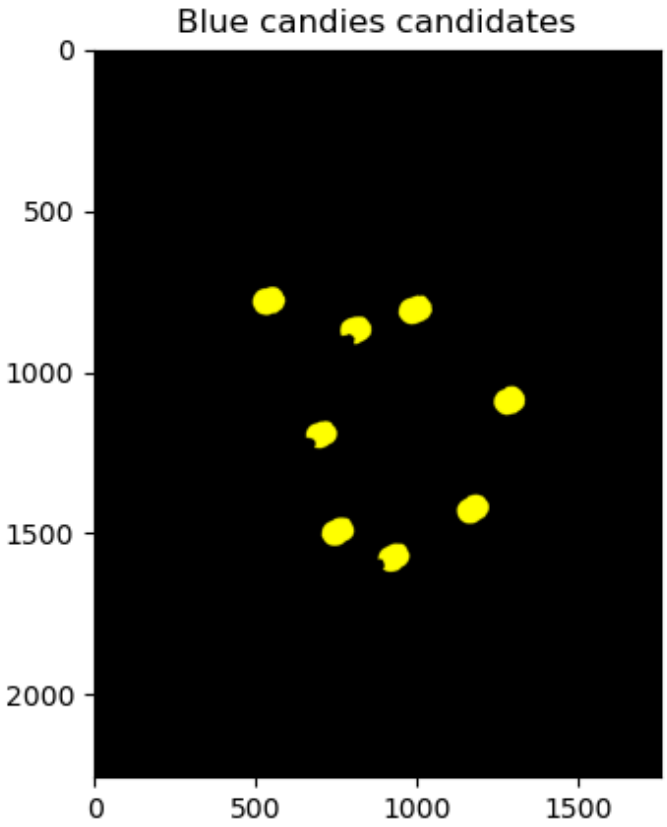

Principal Component 2
Principal Component 1
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

In your first attempt to classify using the PCA space, you examine the PCA coordinates of each sample. You use the first two coordinates and take the absolute value of both of them. You select all BLOBs, where the absolute value of the first component is larger than 1.5 and the absolute value of the second component is larger than 2. Finally, you show the selected BLOBs. It looks like:

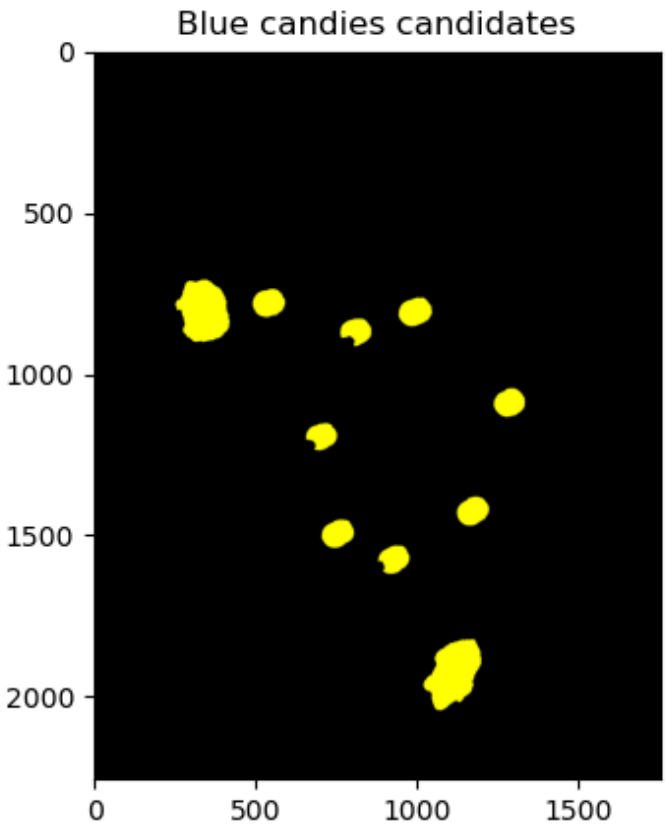
Choose one answer

- ○ Do not know

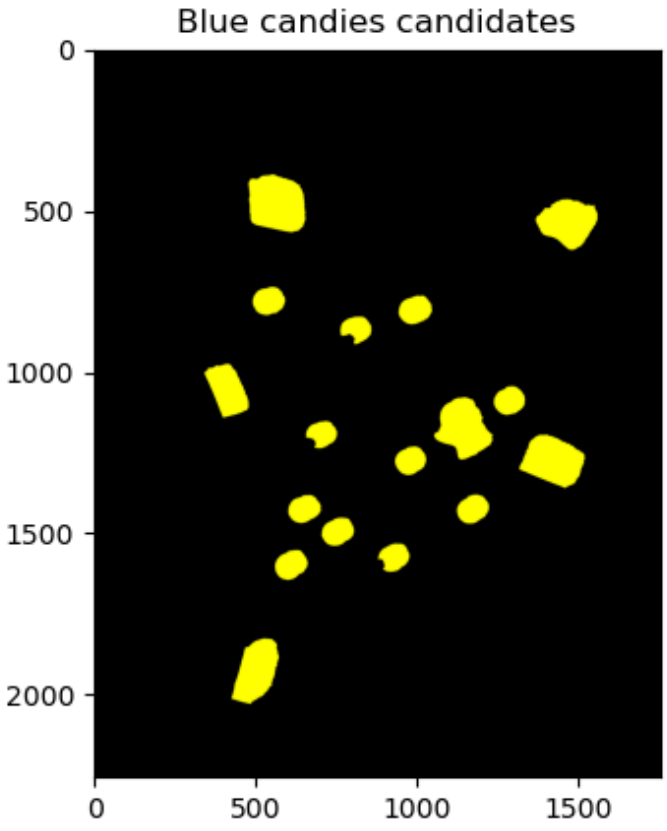
- ◯



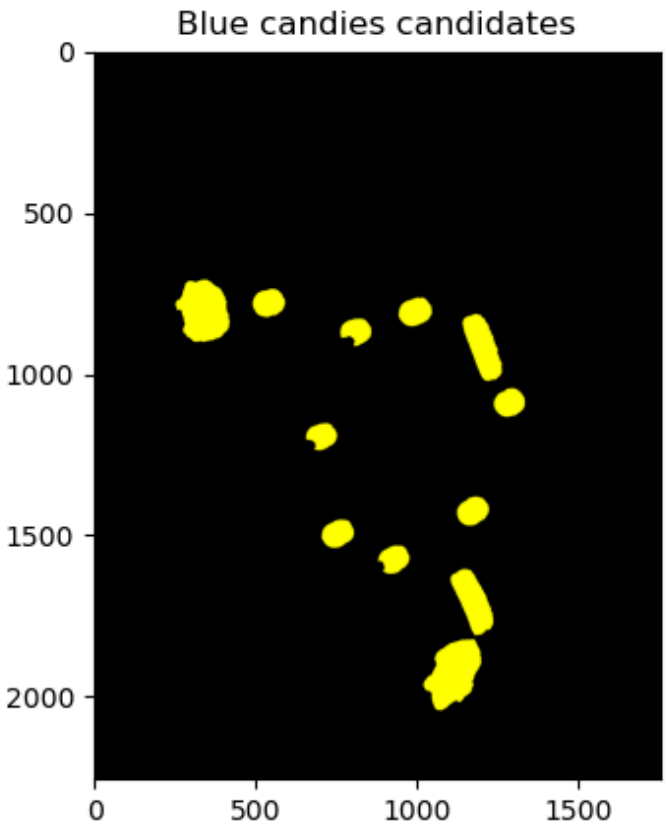
○


Blue candies candidates
0
500
1000
1500
2000
0
500
1000
1500

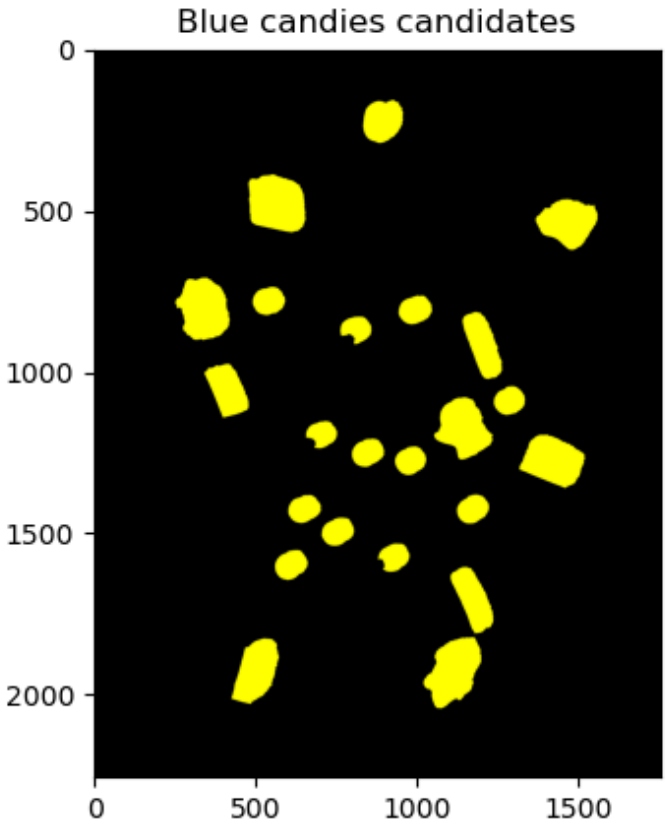
○


Blue candies candidates
0
500
1000
1500
2000
0
500
1000
1500

○

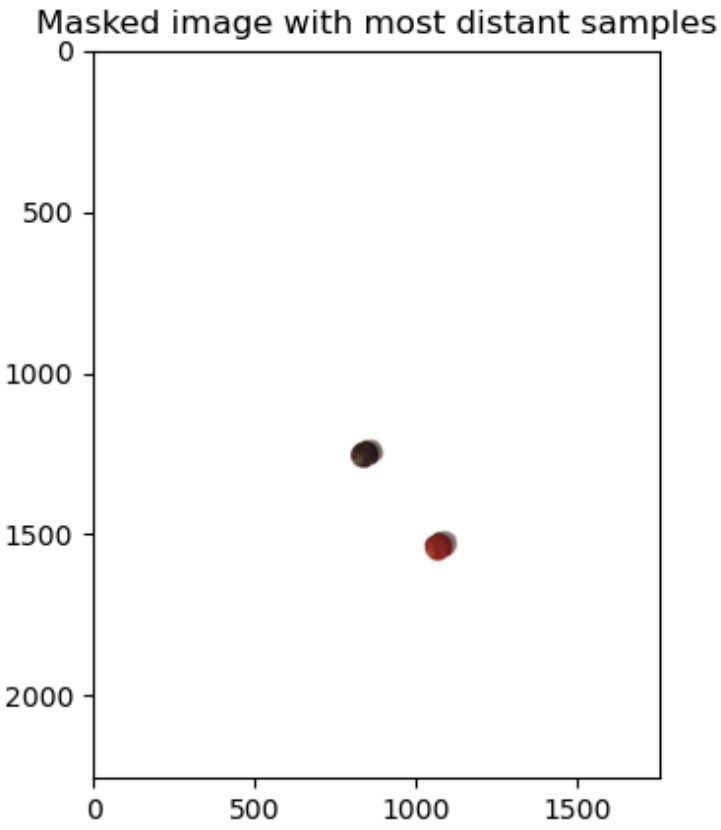

Blue candies candidates
0
500
1000
1500
2000
0
500
1000
1500

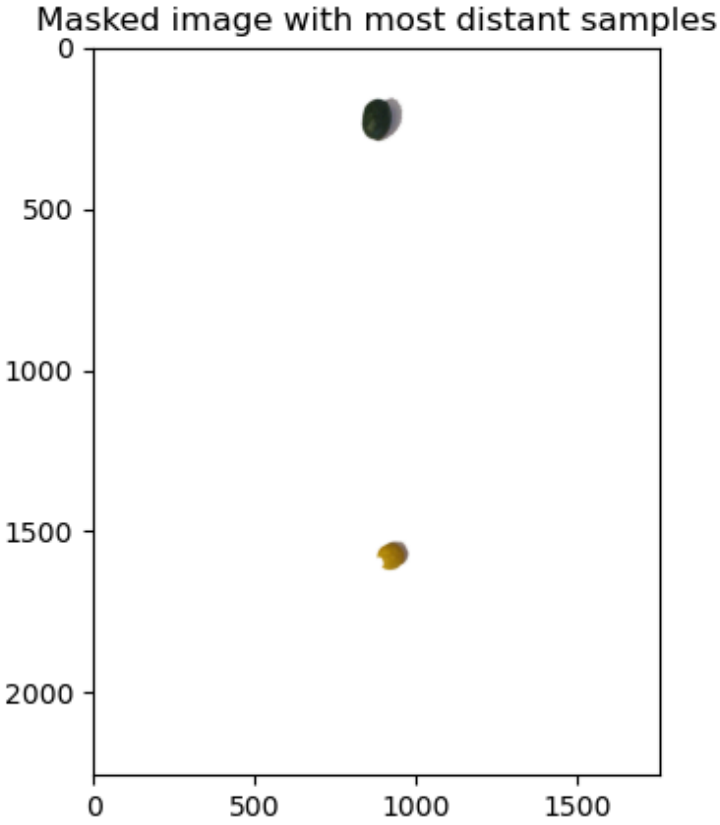
○

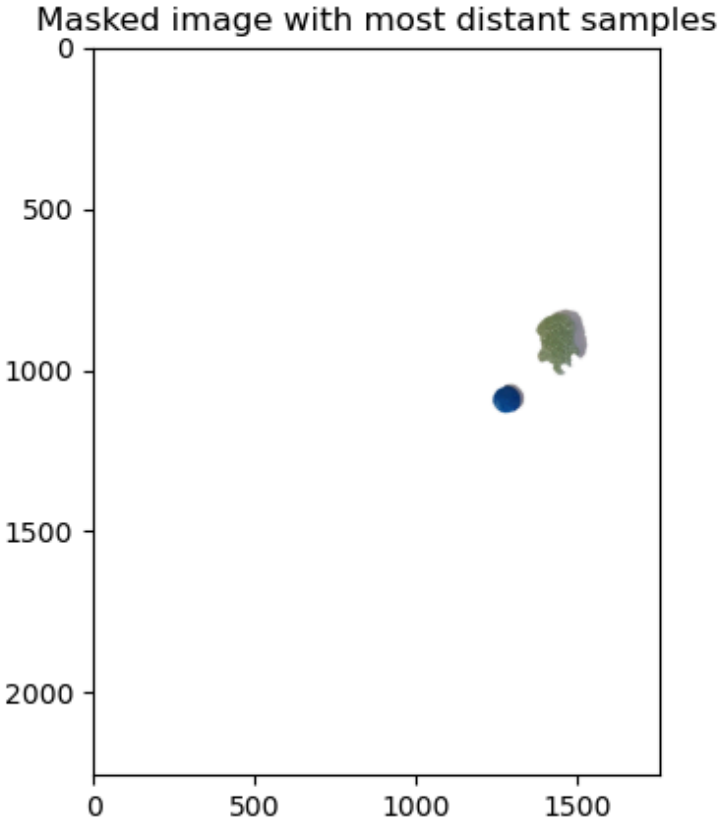

Blue candies candidates
0
500
1000
1500
2000
0
500
1000
1500

To get an idea of the spread of the candy in the PCA space, the two BLOBs that are furthest away from each other when only using the two first principal components are identified. By creating a binary BLOB image with only the two BLOBs, you mask the original RGB image and show the result. You start by filling the masked image with white pixels. How does this masked image look?

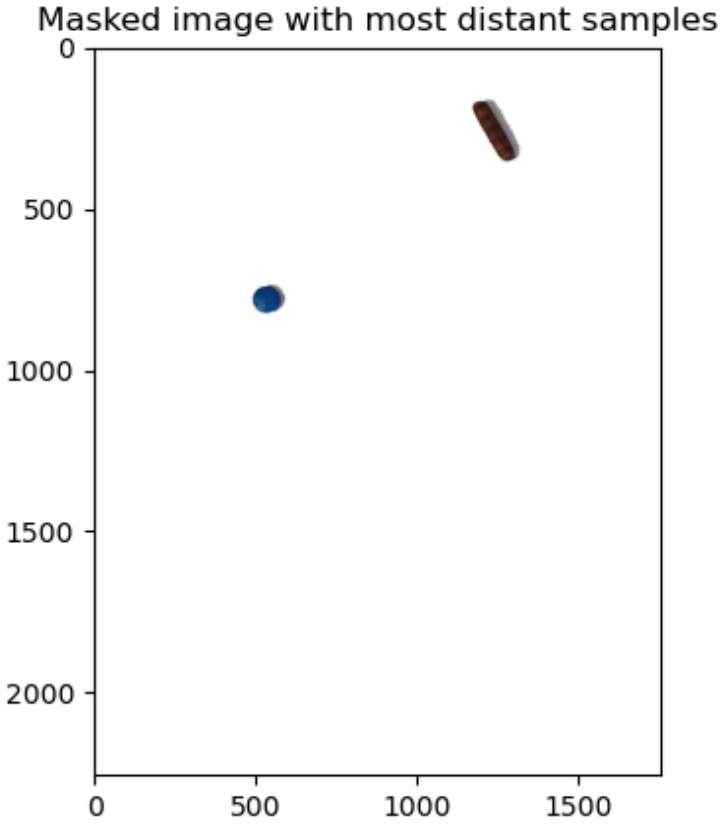
Choose one answer



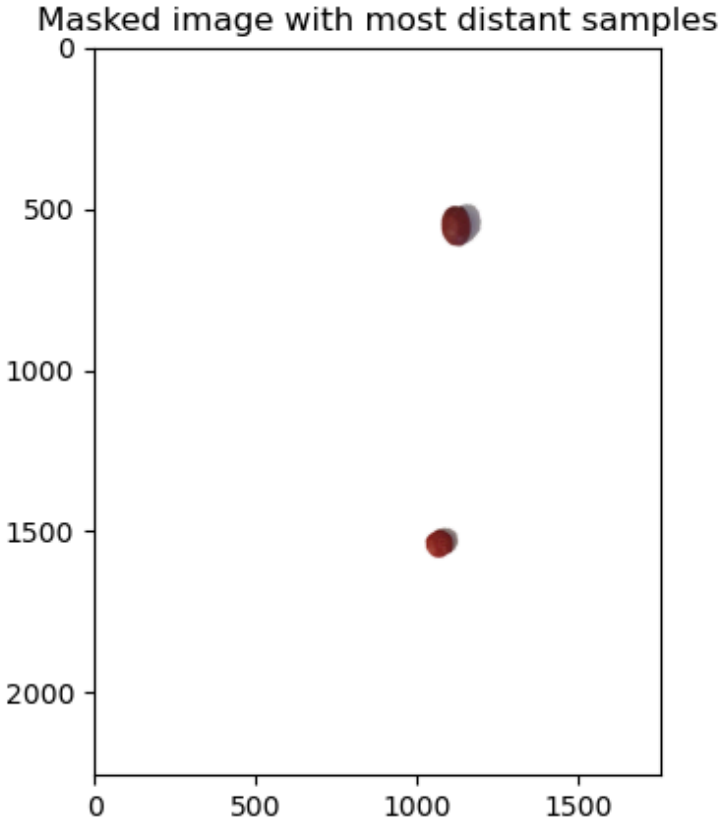

Masked image with most distant samples
0
500
1000
1500
2000
0
500
1000
1500


Masked image with most distant samples
0
500
1000
1500
2000
0
500
1000
1500

○


Masked image with most distant samples
0
500
1000
1500
2000
0
500
1000
1500

○


Masked image with most distant samples
0
500
1000
1500
2000
0
500
1000
1500

○ Do not know

Change detection on computed tomography

You would like to try the methods used in change detection in videos to analyze slices from a computed tomography (CT) scan. Specifically, you would like to investigate if the methods can be used to measure the anatomical changes of bone through the body.

We got a set of training DICOM images:

all_images = ["1-020.dcm", "1-021.dcm", "1-022.dcm", "1-023.dcm", "1-024.dcm", "1-025.dcm", "1-026.dcm", "1-027.dcm", "1-028.dcm", "1-029.dcm", "1-030.dcm"]

and a test image 1-040.dcm

All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b)

You start by estimating a slowly changing background image by starting with setting the background to 1-020.dcm. For each new image, the background image is updated with $\alpha = 0.85$.

After updating the background image with all the training images, you inspect the background image. What is the maximum value in background image?

Choose one answer

- ○ Between 800 and 1000
- ○ Do not know
- ○ Between 1000 and 1200
- ◉ Between 1400 and 1600
- ○ Between 1200 and 1400
- ○ Between 1600 and 1800

To get a first estimate of anatomical changes, you compare the test image 1-040.dcm with the estimated background. You do that by computing the absolute difference image between the two. Secondly, you set a threshold of 400, where pixels above is classified as foreground (1) and the rest as background (0). How many pixels are classified as foreground?

Choose one answer

- ○ Between 11000 and 12000
- ◉ Between 8000 and 9000
- ○ Between 9000 and 10000
- ○ Between 10000 and 11000
- ○ Do not know
- ○ Between 12000 and 13000

To try another approach, you start by thresholding the estimated background image with a threshold of 400, where pixels above threshold is classified as foreground (1) and the rest as background (0). You do the same with the test image 1-040.dcm. Finally, you compare the two binary images using the DICE score. What is the DICE score?

Choose one answer

- ○ Between 0.50 and 0.60
- ◉ Between 0.20 and 0.30
- ○ Do not know
- ○ Between 0.30 and 0.40
- ○ Between 0.60 and 0.70
- ○ Between 0.40 and 0.50

In the final analysis, you compute the overlapping foreground pixels in the two binary images computed in the last question. You do a BLOB analysis on the overlap image and find the largest BLOB and measure its area in pixels. What is this area?

Choose one answer

- ○ Between 500 and 600
- ○ Between 200 and 300
- ◉ Between 300 and 400
- ○ Between 100 and 200
- ○ Between 400 and 500
- ○ Do not know

Candy quality control

Making candy with interesting and non-toxic colors is not easy and therefore a candy-company has contacted you to develop an automated color control system. They provide you with a test photo from their conveyer belt. It is called mm.jpg.

All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b)

You start your analysis by converting the image to gray scale using rgb2gray from scikit-image. Secondly, you use a threshold of 0.65 and set all pixels with a value below to foreground (1) and the rest of the pixel to background (0).
To clean the binary image you first perform a morphological closing with a disk-shaped structuring element with radius 3, followed by a morphological erosion with disk-shaped structuring element with radius 1.
Finally, you compute the BLOBs in the image. How many BLOBS do you find?

Choose one answer

- ○ Between 110 and 120
- ◉ Between 140 and 150
- ○ Between 130 and 140
- ○ Between 100 and 110
- ○ Between 120 and 130
- ○ Do not know

We know that the candy is approximately round and therefore we filter the BLOBs based on area and circularity, where circularity is computed as (2 * sqrt(pi * area)) / perimeter). You keep all BLOBs with a circularity larger than 0.7 and an area larger than 100 pixels. How many candy candidates do you find?

Choose one answer

- Between 40 and 45
- Between 45 and 50
- Between 35 and 40
- Do not know
- between 30 and 35
- Between 25 and 30

The company would like to know how well their mechanism to spread out the candy on the conveyer belt is working. To estimate this, you compute the center of mass of each BLOB that is classified as a candy. For each center of mass you compute, the Euclidean distance to the center of the image (in pixels). The spread of the candy can be measured by the average and standard deviation of these distances. What is your estimate?

Choose one answer

- ○ Average between 400 and 420 and standard deviation between 150 and 160 (pixels)
- ◉ Average between 410 and 430 and standard deviation between 150 and 160 (pixels)
- ○ Average between 390 and 400 and standard deviation between 130 and 140 (pixels)
- ○ Average between 400 and 420 and standard deviation between 160 and 170 (pixels)
- ○ Average between 410 and 430 and standard deviation between 140 and 150 (pixels)
- ○ Do not know

To get an estimate of the average color of the candy, you create smaller image per candy candidate. This is done by computing the center of mass of each BLOB that is classified as a candy. This position is used as a center of a rectangular crop that has a side length of 100 pixels. Using these crop coordinates a crop is extracted from the original color image for each BLOB that is classified as candy. The result of this is a list of rectangular crops each containing a color image of a candy in the middle.
You compute an average image based on these crops. What is the color of the resulting candy?

Choose one answer

- ○ Black
- ○ White
- ◉ Brown
- ○ Light gray
- ○ Purple
- ○ Do not know

Using the color image crops you compare the pixel-wise sum of squared difference between all crops to find the two candies that are most different in color. You find these two candies as the ones that have the largest sum of squared difference. What color are they?

Choose one answer

- ○ Blue and green
- ○ Blue and red
- ○ Red and dark brown
- ○ Green and yellow
- ○ Do not know
- ◉ Yellow and dark brown

Tiger identification

A zoologist believes that it is possible to identify tigers based on their stripe patterns. To test that, you help developing a system to extract this pattern from photos of tigers. To start, you try to segment the photo Tiger.jpg into two classes: **black tiger stripes** or **not black tiger stripes.**

A training image (“**ROI_Tiger.png”)** contains regions of interest used to train the classifier and has the same matrix dimensions as the image. It includes three intensity values: ‘**0’** for background pixels that should be ignored, ‘**90’** for pixels belonging to class 1 (black tiger stripes), and ‘**165’** for pixels belonging to class 2 (not black tiger stripes).

All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b)

You create a parametric classifier based on the training data and use this classifier on Tiger.jpg where you show the resulting class 1 pixels as white. How does your resulting image looks like?

Choose one answer

○

○

○ Do not know

You want to see if a Fishers LDA can provide even better strip classification. The two classes are assumed to have equal prior probabilities (50%). What is the probability that the pixel at position (r, c) = (347, 247) in the image belongs to class 1 (black tiger stripes)?

Choose one answer

- ○ Between 0.1 and 0.25
- ○ Between 0.98 and 1
- ○ Between 0.65 and 0.71
- ○ Do not know
- ○ Between 0.31 and 0.63
- ◉ Between 0.7 and 0.87

Registration of 3D microscopy images

Two 3D microscopic image volumes, Image A (“VolA.nii”) and Image B (“VolB.nii”), are to be stitched into a single volume to extend the field of view of the scanned object. You are given a [3x3] forward transformation matrix T in “matrix.txt”.

Resample Image B (the moving image) by linear interpolation using the backward transform and a translation of Tt = [-2,-10,5], using the default transformation center. The resampled image is called Image BB.

Tips: “matrix.txt” can be loaded using the numpy function np.loadtxt()

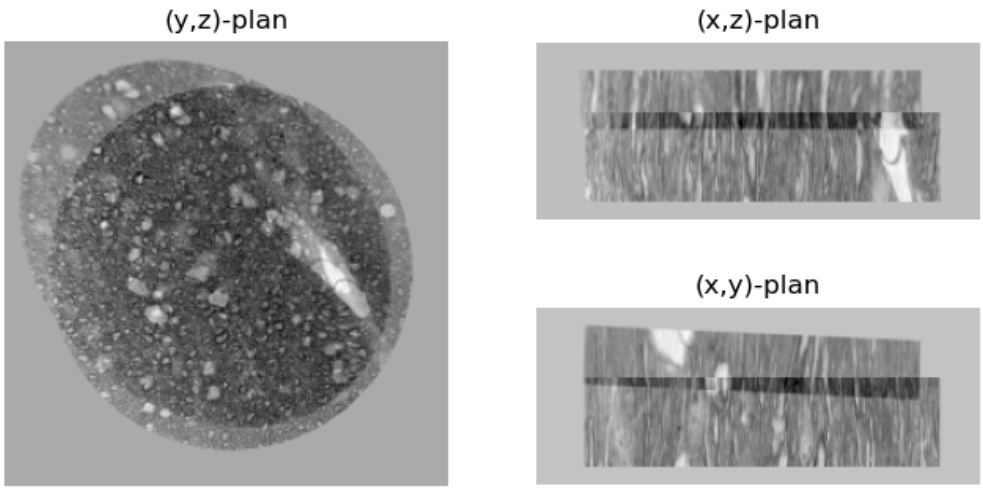
After this step, Image BB and Image A (“VolA.nii”) are merged into a new image volume by blending their overlapping region with 80% weighting to Image BB intensities and 20% weighting to Image A intensities.

All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b)

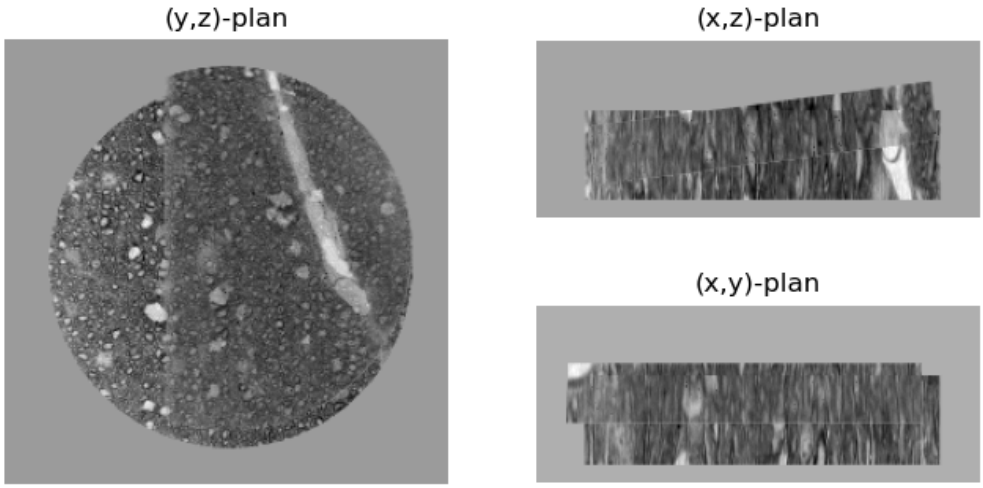
How does the new stitched image volume look in the three orthogonal views in coordinate [59,100,100]?

Choose one answer

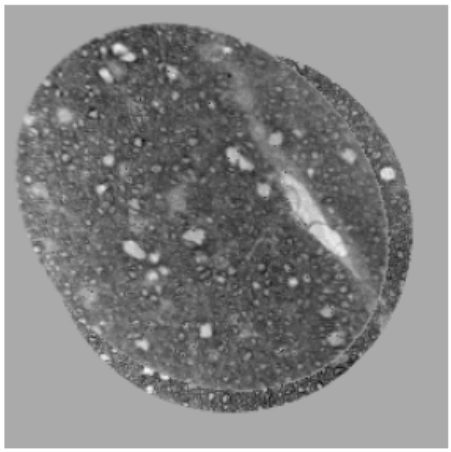
○



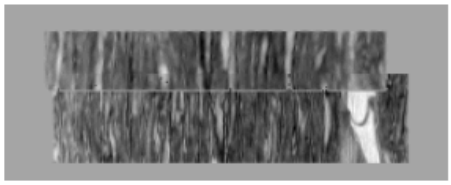
○



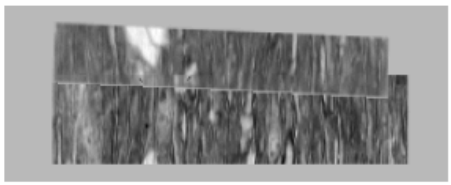
(y,z)-plan



(x,z)-plan

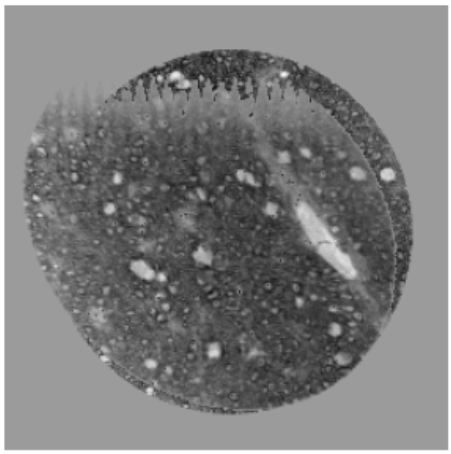


(x,y)-plan

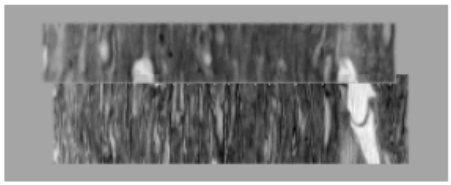


Do not know

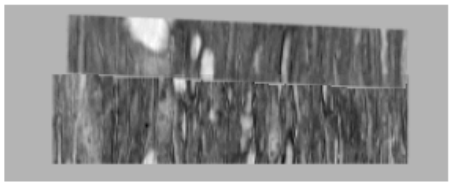
(y,z)-plan



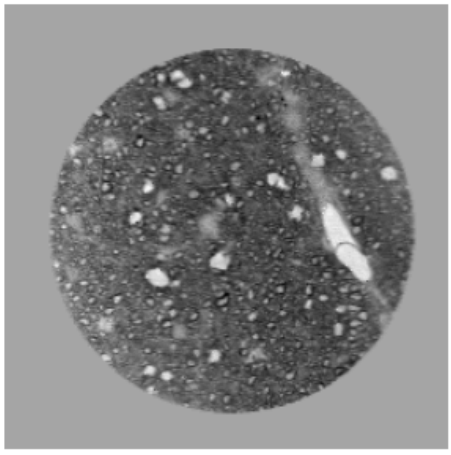
(x,z)-plan



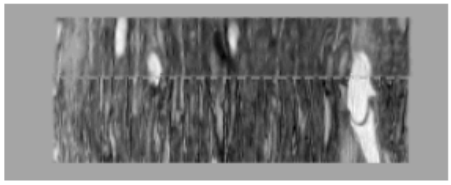
(x,y)-plan



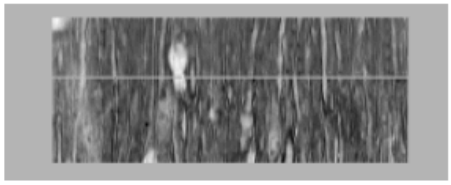
(y,z)-plan



(x,z)-plan



(x,y)-plan



You are provided with two 3D image volumes (“VolA.nii” and “VolB.nii”) of the same matrix size, which share an overlapping region for subsequent stitching. The overlapping region is defined as the voxels that are non-zero in both images. What is the normalised cross correlation (NCC) expressed in degrees for the overlapping region?

In this case, we use the equation below, to compute the NCC:

$$\mathrm{NCC}(x, y) = \frac{\sum_{j=-R}^{R} \sum_{i=-R}^{R} (H \cdot F)}{\sqrt{\sum_{j=-R}^{R} \sum_{i=-R}^{R} (F \cdot F)} \cdot \sqrt{\sum_{j=-R}^{R} \sum_{i=-R}^{R} (H \cdot H)}}, \qquad (5.6)$$

Choose one answer

- Between 61 and 80 degrees
- Between 81 and 90 degrees
- Between 0 and 20 degrees
- Do not know
- Between 41 and 60 degrees
- Between 21 and 40 degrees

Blood vessel trajectories in the eye, extracted from a fundus image, have diagnostic value. We therefore test an algorithm that can trace a dark vessel from the top of the image to the bottom of the image.

We apply a shortest-path algorithm using dynamic programming to the Fundus image **("Eye_Fundus_1.png")** and start by computing an accumulator image.

To avoid boundary issues, the accumulator image is computed for all rows and columns except for the first and last column which are set to "0".

What is the total cost at column 46 in the Fundus image?

All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b)

Choose one answer

- ○ Between 1500 and 1900
- ○ Do not know
- ○ Between 2600 and 3100
- ○ Between 25 and 98
- ◉ Between 2100 and 2599
- ○ Between 1000 and 1200

Finding objects using gradient descent

The binary image (“MergedObjects.png”) contains overlapping circular objects.

We would like to locate the center of one of the circular objects and we would like to explore how well a gradient-descent based approach can do this.

This can be done by computing a distance map, that for each pixel stores the distance to the nearest object. The gradient of the distance map therefore points toward the nearest object.

First a signed distance map and the gradient of the distance map are computed using for example:

```
from scipy import ndimage

img = io.imread(f'{in_dir}MergedObjects.png').astype('uint8')
# Make the inverse distance map
dist_map = - ndimage.distance_transform_edt(img)
#  Make gradient maps
# Row wise gradient (r-direction)
g_r = ndimage.prewitt(dist_map, axis = 0)
# Column wise gradient (c-direction)
g_c = ndimage.prewitt(dist_map, axis = 1)
```

The gradient descent algorithm includes this descent step:

p(n+1) = p(n) - step_size * gradient(p(n)) ,
where **p(n)** represents the (r, c) coordinates at iteration **n**, and **step_size** is a constant.

Since the gradient map is an image, the position **p(n)** and the predicted next position **p(n+1)** are rounded down to the nearest integer coordinates.

All data for the exam can be here (https://designer.mcq.eksamen.dtu.dk/api/images/073d0bec-2af0-4dcb-ad86-5f12d28e644b)

What is the position **p(1)** when the start position **p(0) = (104, 22)** and the step_size = 0.12.

Choose one answer

- (104, 29)
- (29, 104)
- (104, 22)
- (104, 89)
- (103, 23)
- Do not know

You now try to set the step_size to 0.35.

To track the descent and find out when it stabilizes, the Euclidean distance, **D**, between **p(n)** and **p(n+1)** is computed for each step.

What is **D** after 40 iterations when starting the gradient descent in **p(0) = (243, 167)** ?

Choose one answer

- ○ Do not know
- ○ Between 1 and 1.1
- ○ Between 1.1 and 1.25
- ◉ Between 1.35 and 1.45
- ○ Between 2 and 2.20
- ○ Between 1.55 and 1.80