Exam / 31392 / May 2021

Question 1 (0.0% of the exam)

Welcome to the Exam of the course 31392: Perception for Autonomous Systems!

This exam will take 4 hours. There are 34 questions (excluding this!). Each question has a different weight on your final grade; the weight of each question (as a percent) is indicated in the question description.

In this link you will find all the files (materials and datasets) needed to complete ALL the "numerical" exercises in this exam: link (alternate link)
Please download it now!!

The folder structure (after you unzip the file) contains the appropriate files per **Question number.**

The following approach to scoring responses is implemented and is based on "One best answer":

- There is always only one correct answer – a response that is more correct than the rest
- Students are only able to select one answer per question
- The score of each question is given next to the question number.
- Every incorrect answer corresponds to 0 points (incorrect answers do not result in subtraction of points)

Page 1

Have you downloaded the files?

- [x] Yes
- [ ] No, I want to re-read the above

Question 2 (2.0% of the test grade):

What is the output of a smoothing, linear spatial filter?

- [ ] Maximum of pixels
- [ ] Median of pixels
- [ ] Minimum of pixels
- [x] Average of pixels

Question 3 (2.0% of the test grade):

Consider the matrix below. Assume that we want to convolve an image with this matrix. What will be the outcome?

0	0	0
0	0	0
0	1	0

- [x] The image will be shifted downwards by 1 pixel
- [ ] The image will remain unaffected
- [ ] The image will be blurred
- [ ] The image will be sharpened
- [ ] The image will be shifted to the right by 1 pixel
- [ ] The image will be shifted upwards by 1 pixel

Question 4 (2.0% of the test grade):

Which of the following filters response is based on ranking of pixel values?

- [x] Nonlinear smoothing filters
- [ ] Linear Smoothing Filters
- [ ] Sharpening Filters

Question 5 (2.0% of the test grade):

Consider that we are using the following kernel to perform convolution and correlation on an image. Do you expect the results of the two operations to be different or identical and why?

0.2	0	0.2
0	0.2	0
0.2	0	0.2

- ○ The result will be different because none of the kernel elements is larger than 0.5
- ✔ The results will be identical because the kernel is symmetric
- ○ The results will be identical because all the elements of the kernel add up to 1
- ○ The results will be different because convolution and correlation are two different operations

---

Question 6 (2.0% of the test grade):

Can we perform corelation using convolution?

- ● Yes, by modifying the kernel
- ○ Yes, the outcome of corelation and convolution are always the same
- ○ No, correlation and convolution are fundamentally different
- ✔ No, we can only perform convolution using correlation, but not the inverse

Question 7: Harris Detector
Harris Detector

Question 7.1 (1.0% of the test grade): With regards to **Harris Detector**, choose the correct statement

- ○ Harris detector uses the difference of gaussians method
- ✔ The Harris detector is used in computer vision to detect Corners
- ○ The Harris detector is used in computer vision to detect Edges

---

Question 7.1 (1.0% of the test grade): With regards to **Harris Detector**, assuming M the "Harris Matrix" (autocorrelation matrix or structure tensor), choose the correct statement

- ○ Assuming λ1 and λ2 the eigenvalues of the matrix M, points where λ1>>λ2 are considered slopes
- ○ Assuming λ1 and λ2 the eigenvalues of the matrix M, points where λ2>>λ1 are considered corners
- ✔ Assuming λ1 and λ2 the eigenvalues of the matrix M, points where λ1~λ2 and large are considered corners

Question 8: SIFT
Scale-invariant feature transform

Question 8.1 (1.0% of the test grade): Concerning the **SIFT** algorithm, choose the correct statement:

- ✔ The detection is based on the Difference of Gaussians
- ○ The detection is based on the Gradient of the Image
- ○ The description is based on the Difference of Gaussians

Question 8.2 (1.0% of the test grade): Concerning the **SIFT** algorithm, choose the correct statement:

- ✔ The description is based on the Gradient of the Image
- ○ The fingerprint of the SIFT algorithm is a vector of 64 values

  Not in original paper, it was brute forced there (eucledian distance)
- ○ In the original paper, the matching of the features used the k-NN algorithm

Question 9: Hough Transform
Hough Transform

Question 9.1 (1.0% of the test grade):

With regards to **Hough Transform**, choose the correct statement:

- ○ The Hough transform can't be used to detect circles
- ✔ The Hough transform can be used to detect lines
- ○ The Hough transform is a classification algorithm

Question 9.2 (1.0% of the test grade):

With regards to **Hough Transform**, choose the correct statement:

- ◯ The Hough transform is only used in Computer Vision
- ◯ Hough transform can't handle outliers
- ✔ RANSAC should be used as an alternative to the Hough transform when the dimension of feature is high

Question 10 (2.0% of the test grade) - Numerical - Hough Line Detection

Using the following image (in the files you downloaded), do the following

- Convert it to grayscale (do not change the size!).
- Apply Canny edge detection (cv2.Canny) with the parameters "threshold1=100" and "threshold2=200".
- Apply HoughLines (cv2.HoughLines) with the parameters "rho=1", "theta=0.0017", "threshold=200".

How many lines are detected?

- ◯ 16
- ✔ 32
- ◯ 64
- ◯ 128

Question 11 (2.0% of the test grade) - Numerical - Harris Corner Detection

Using the following image (in the files you downloaded), do the following:

- Convert the image to grayscale (do not change the size!).
- Apply Harris corner (cv2.cornerHarris) to the image with the parameters "blockSize=2", "ksize=3", "k=0.04".

From the result of Harris Corners, how many values are above 0.01?

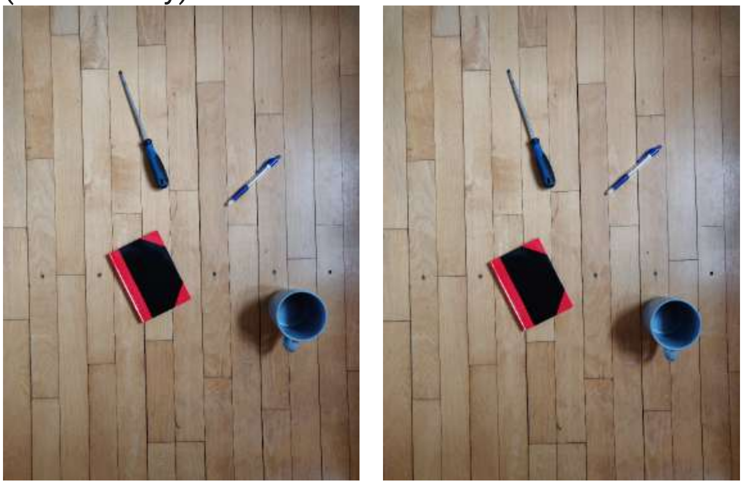
- ○ 128
- ○ 158
- ✔ 188
- ○ 208

Question 12 (2.0% of the test grade) - Numerical - Optical Flow

Using the following image (in the files you downloaded), do the following:

- Load the two images and convert them to grayscale (do not change the size!).
- Use cv2.goodFeaturesToTrack to find features on the first image (things1.png) using the parameters "maxCorners=100", "qualityLevel=0.3", and "minDistance=7".
- Apply sparse optical flow using the function ccv2.calcOpticalFlowPyrLK()

What is the maximum amount of pixels moved for any object in the x direction (horizontally)?



- [x] ~15.5
- [ ] ~12.5
- [ ] ~20.5
- [ ] ~22.5

Question 13: Stereo Vision
Rectified stereo

Question 13.1 (1.0% of the test grade):

- [x] Epipolar geometry can be used to describe both unrectified and rectified stereo cases
- [ ] Physically mounting two sensors on a common plane results in a stereo system that requires no further rectification Double check with the answer from previous exam
- [ ] In rectified stereo, there are some epipolar lines that are not parallel.

Question 13.2 (1.0% of the test grade):

- ○ The disparity value of a point grows with its depth
- ✔ We cannot obtain depth from stereo without knowing the system's focal length and baseline
- ○ The rectified stereo case is more computationally demanding than un-rectified stereo

---

Question 14 (2.0% of the test grade):

Which of the following statements about stereo correspondence algorithms are correct and which are wrong?

- ○ Bigger windows for calculating dis-similarity metrics such as SAD or SSD result always in better disparity results, at the expense of more calculations
- ○ Bigger windows for calculating dis-similarity metrics such as SAD or SSD result always in worse disparity results, but calculation is faster
- ✔ Local algorithms typically produce inferior disparity maps, compared to global algorithms
- ○ Bigger windows are typically preferred for images with finer and more complicated texture

---

Question 15 (2.0% of the test grade):

Could convolution be used to implement (dis-)similarity calculations in stereo matching?

- ○ Yes, but with proper considerations for formulating the kernel
- ✔ No, it is correlation that expresses (dis-)similarity
- ○ Yes, because in (dis-)similarity calculations the considered kernel is symmetric
- ○ No, because in convolution the kernel needs to be smaller than the image

---

Question 16: Monocular Case
Mono Calibration and projection

Question 16.1 (1.0% of the test grade): With regards to the monocular camera Projection and Calibration, select the correct statement

- The projection matrix includes instrinsic parameters
- The projection matrix includes extrinsic parameters
- The Homography is used to project a point from 3D to 2D

2

Question 16.2 (1.0% of the test grade): With regards to the monocular camera Projection and Calibration, select the correct statement

- When using a flat calibration pattern we can employ the homography to perform calibration
- Lens distortion is modeled as a linear function
- The barrel distortion cannot be corrected

Question 17: Stereo Case
Stereo Calibration and projection

Question 17.1 (1.0% of the test grade): With regards to the stereo camera Projection and Calibration, select the correct statement

- The Fundamental Matrix describes only extrinsic parameters
- The Essential matrix contains only extrinsic parameters
- The camera matrix cannot be calculated from the Fundamental Matrix

Question 17.2 (1.0% of the test grade): With regards to the stereo camera Projection and Calibration, select the correct statement

What?

- The Fundamental matrix projects image points from one image of the stereo pair to the other
- The epipoles of a stereo pair cannot be on the baseline
- The epipoles of a stereo system are always found between the two cameras

Question 18 (6.0% of the test grade): - Numerical - Stereo Calibration

One property of stereo rectified images is that epipolar lines are parallel, and that the points on an epipolar line in one image plane can be found on the corresponding epipolar line in the other image plane. Therefore, it is necessary to know the equation for the epipolar in order to optimize the performance of any later template matching.

Determine the approximate coefficients of the epipolar lines in the imageset left.png and right.png by taking the average of all the epipolar lines. You should use **2000** of the best matching sift keypoints to compute the epipolar lines.

Note: opencv computes the epipolar lines in the form ax+by+c=0.

Please select the average epipolar lines that are closer to the ones you calculated

- [ ] [-1.69e-02 7.97e-01 -0.26e+02]
  [17.02e-02 -7.97e-01 0.25e+02]
- [x] [-6.69e-02 9.97e-01 -1.26e+02]
  [7.02e-02 -9.97e-01 1.25e+02]
- [ ] [-11.69e-02 17.97e-01 -1.06e+02]
  [11.02e-02 -17.97e-01 1.05e+02]
- [ ] [-2.69e-02 3.97e-01 -0.06e+02]
  [1.02e-02 -3.97e-01 0.05e+02]

Question 19:
The number of iterations k that RANSAC needs in order to achieve a good (outlier-free) model with probability p
is defined by the formula: $k = \frac{log(1-p)}{log(1-w^2)}$
where k is the number of iterations, p is the probability of RANSAC having chosen a set of points free of outliers, w is the proportion of inliers with respect to all the points in the dataset.

Question 19.1 (1.5% of the test grade):

Assume that we want a probability of success at least 98% and we know that 75% of the points are inliers. How many iterations does RANSAC need to achieve this?

- ○ The number of needed iterations is: $k = 3$
- ✔ The number of needed iterations is: $k = 5$
- ○ The number of needed iterations is: $k = 14$
- ○ The number of needed iterations is: $k = 64$

Question 19.2 (1.5% of the test grade):

How would the number of needed iterations change if the size of our dataset (number of points) doubled, but all other aspects of our scenario remained the same?

- ○ With a dataset twice as big, the number of required iterations would be half
- ✔ With a dataset twice as big, the number of required iterations would be the same
- ○ With a dataset twice as big, the number of required iterations would be double

---

Question 20
Iterative Closest Point (ICP) algorithm

Question 20.1 (1.5% of the test grade):

- ○ ICP is guaranteed to converge (within a reasonable accuracy) no matter what the initial relative pose of the two point clouds is.
- ○ ICP is one of the underlying algorithms for implementing the Kabsch algorithm
- ✔ ICP is not particularly robust to outliers

Question 20.2 (1.5% of the test grade):

- ◯ ICP could employ either Spin Images or FPFH for finding the interediate transformations until convergence
- ◯ When using ICP, one may only use the Kabsch algorithm and not the Procrustes analysis
- ✔ ICP is a deterministic algorithm

Question 21 (1.0% of the test grade):

In Point Cloud Registration, local alignment usually takes place first, and global alignment second.
Is the statement correct or wrong?

- ◯ Correct
- ✔ Wrong

Question 22 (1.0% of the test grade):

The Kabsch algorithm is typically used both for Local and Global alignment.

- ✔ Yes, the Kabsch algorithm is used in both cases
- ◯ No, the Kabsch algorithm is only used for Local alignement
- ◯ No, the Kabsch algorithm is only used for Global alignment

---

Question 23
Apply K-means and/or PCA on the provided dataset and select the correct answers to the following questions.

Question 23.1 (4.0% of the test grade):

Apply K-means and use the elbow method to find how many clusters (K) are optimal for this dataset.

- Follow the guidelines and use the material found in the provided archive.

- [ ] The optimal K is: 3
- [ ] The optimal K is: 4
- [x] The optimal K is: 5
- [ ] The optimal K is: 6
- [ ] The optimal K is: 7

Question 23.2 (3.0% of the test grade):

Perform PCA on the provided dataset and determine the minimum number of components required to express 95% of the variance.

- Follow the guidelines and use the material found in the provided archive.

- [ ] For 95%, the number of needed components is: 2
- [ ] For 95%, the number of needed components is: 3
- [ ] For 95%, the number of needed components is: 4
- [ ] For 95%, the number of needed components is: 5
- [x] For 95%, the number of needed components is: 6

Question 23.3 (3.0% of the test grade):

Perform K-means on the PCA transformed data and calculate the Distortion/Inertia (sum of squared distances of samples to the closest centroid)

- Use the value of K found in "Question 23.1" and apply K-means on the PCA transformed data that you obtained in "Question 23.2".
- Follow the guidelines and use the material found in the provided archive.

- [x] Distortion has a value between: 1500 and 2000
- [ ] Distortion has a value between: 3000 and 4000
- [ ] Distortion has a value between: 4500 and 5000

Question 24 (6.0% of the test grade):

Apply Linear Regression to the provided data to obtain a model of the form *y=ax+b*
.
Follow the provided guidelines for the implementation.
What are the values of the parameters "a" and "b" in that model (APPROXIMATELY)?

- ○ a = 0.23, b = 25.60
- ✔ a = 0.55, b = 19.86
- ○ a = 0.82, b = 19.86
- ○ a = 0.55, b = 12.37
- ○ a = 0.23, b = 12.37
- ○ a = 0.82, b = 25.60

Question 25 (1.0% of the test grade):

The clustering algorithm DBSCAN needs to always have some points assigned as "noise points".
Is this statement correct or wrong?

- ○ Correct
- ✔ Wrong

Question 26: State Estimation
Histogram Filter + Kalman Filter

Question 26.1 (1% of the test grade): With regards to Histogram Filter, select the correct statement

- ✔ Histogram Filter Concerns Discrete States
- ○ Histogram Filter Concerns Unimodal Uncertainty Distributions
- ○ In the Histogram filter, measurement involves convolution and movement involve the Bayes rule

---

Question 26.2 (1% of the test grade): With regards to Kalman filter, select the correct statement

- [ ] In Kalman filter, the variance of the estimation is higher after measurement
- [x] In Kalman filter, the variance of the estimation is higher after movement
- [ ] Kalman Filter Concerns Discrete States
- [ ] Kalman Filter Concerns Multimodal Uncertainty Distributions

Question 27 (5.0% of the test grade) - Numerical - Kalman Filter, State Prediction, Covariance Prediction

You are tracking an object with a Kalman filter. At the current time the state is:

x	3
x'	0.5
y	2
y'	0.33

and the current covariance is:

5	1	0	0
1	2	0	0
0	0	5	1
0	0	1	2

With x and y being the position in the x and y direction and x', y' being the velocities in the x and y direction, we assume the object follows a constant velocity model and that there are no external forces (u=0) and no process noise.

If we have a timestep of dt=1, What is:
The next predicted state and the next predicted covariance?

- ○

x	4
x'	0.5
y	2
y'	0.33

5	4	0	0
4	2	0	0
0	0	5	4
0	0	4	5

- ○

x	3
x'	1
y	2
y'	0.33

1	2	0	0
2	1	0	0
0	0	1	4
0	0	4	1

- ✔

x	3.5
x'	0.5
y	2.33
y'	0.33

9	3	0	0
3	2	0	0
0	0	9	3
0	0	3	2

- ○

x	3

x'	0.5
y	2.33
y'	0.33

7	1	0	0
1	2	0	0
0	0	7	1
0	0	1	2

Question 28 (3% of the test grade) - Numerical - Kalman Tracking

You are now tracking a different object using a Kalman filter and a constant velocity model. You have just predicted the state and covariance:

x	5
x'	0.5
y	7
y'	0.8

0.2	0	0	0
0.2	0.1	0	0
0	0	0.2	0
0	0	0.2	0.1

Furthermore, we have just measured the position of the object to be:

4.8
7.1

Finally, we have the observation noise R:

0.2	0.2
0.2	0.2

What is the state after having updated it with the current measurement (do not run the predict step, only the update step)?

- [x]

x	4.833333
x'	0.333333
y	7.133333
y'	0.933333

- [ ]

x	5.133333
x'	0.633333
y	6.933333
y'	1.133333

- [ ]

x	4.933333
x'	0.533333
y	7.033333
y'	0.833333

- [ ]

x	4.733333
x'	0.533333
y	6.833333
y'	0.833333

Question 29
You will train and test a Support Vector Machine (SVM).
Please use the material provide in the archive.

Question 29.1 (3.0% of the test grade):

Examine the provided code. What proportion of the original dataset is used for **training**?

- ○ Proportion used for training: 30%
- ✔ Proportion used for training: 70%
- ○ Proportion used for training: 75%

Question 29.2 (3.0% of the test grade):

Apply a SVM as instructed in the provided code. What is the reported accuracy of your SVM (APPROXIMATELY)?

- ○ Accuracy: 0.83
- ○ Accuracy: 0.92
- ✔ Accuracy: 0.96

Question 30 (1.0% of the test grade):

The classification algorithm k-NN requires approximately similar time for training and testing (inference).
Is this statement correct or wrong?

- ✔ Correct

knn learn just by storing the points. Complexity is linear O (1)

- Wrong

Question 31: Visual Odometry
Visual Odometry

Question 31.1 (1.0% of the test grade) - Concerning Visual Odometry, select the correct statement

Triangulation is not accurate, in 3D - 3D we have two triangulation,, for 3D-2D only one and for 2D-2D zero

- ✔ The 3D to 3D methods are less accurate than the 3D to 2D ones
- ○ The axis angle representation suffers from the "gimbal lock" problem
- ○ Visual Odometry can only be performed on a frame to frame manner

Question 31.2 (1.0% of the test grade) - Concerning Visual Odometry, select the correct statement

- [x] 2D to 2D methods are "accurate up to scale"
- [ ] 3D to 3D approaches use the image coordinates of the current frame and the 3D position of the previous frame to estimate the motion
- [ ] 2D to 2D methods are equivallent to calculating the fundamental matrix

Question 32 (14.0% of the test grade) - Numerical - Visual Odometry

As an alternative to GPS, a car is driving while recording images with a stereo camera setup with the goal of tracking the relative position of the car.

In the provided material (you downloaded) the two directories left and right contain timestamped images from the drive

The steps to tracking the car is as follows:

- Do only once:
  1.1. Capture two frames I_{k-2}, I_{k-1}
  1.2. Extract and match features between them
  1.3. Triangulate features from I_{k-2}, I_{k-1}
- Do at each iteration:
  2.1. Capture new frame I_k
  2.2. Extract features and match with previous frame I_{k-1}
  2.3. Compute camera pose (PnP) from 3-D-to-2-D matches
  2.4. Triangulate all new feature matches between I_k and I_{k-1}
  2.5. Iterate from 2.1

Compute the final Pose (Position: [X,Y,Z] Orientation: [axis angles x, axis angles y, axis angles z]) of the car.

Note:
baseline = 0.54
#Camera Matrix
K = np.array([[7.188560e+02, 0.000000e+00, 6.071928e+02], [0, 7.188560e+02, 1.852157e+02], [0, 0, 1]])
#projection matrix for Left and Right Image. The images are stereo rectified, i.e. perfectly parallel and scanline, so:
M_left = K.dot(np.hstack((np.eye(3), np.zeros((3, 1)))))
M_rght = K.dot(np.hstack((np.eye(3), np.array([[-baseline, 0, 0]]).T)))

- ○ Position = [-11.908 [0.984] [1.727]
  Rotation = [0.000] [1.296] [0.002]

2

- ○ Position = [-16.908] [0.684] [11.727]
  Rotation = [0.0102] [1.596] [0.072]

- ✔ Position = [-6.908] [0.384] [4.727]
  Rotation = [0.040] [1.396] [0.271]

- ● Position = [-7.908] [0.184] [7.727]
  Rotation = [0.040] [1.996] [0.471]

Question 33 (2.0% of the test grade): With regards to non-linear least squares optimization for SLAM select the correct statement

- ○ Gauss-Newton does not suffer from overshooting
- ○ Levenberg Marquadt suffers from slow covergence
- ✔ Levenberg Marquadt is a combination of Gauss-Newton and Gradient Descent

Question 34 (2.0% of the test grade): With regards to SLAM select the correct statement

- ○ SLAM is concerned solely with the identification of the position of a camera
- ✔ SLAM is defined as a graph where nodes are the poses/3D landmark positions and edges are the observations/odometry
- ○ The goal of SLAM is to satisfy only the pose nodes in the pose graph