Exam in
02504 Computer Vision
Spring 2024

General information

- The exam consists of 22 questions. All questions have equal weight: the correct answer gives 1 point, incorrect or missing answer gives 0 points. You only need to submit the answers to the questions. You should not upload any notes or calculations.
- There is *one* correct answer for each question. Some of the numeric results have been rounded, and may deviate slightly from your result. This should not prevent you from being able to pick the correct answer.
- Each page contains one question. If there are illustrations and images, those refer to the question on that page.
- The notation in the questions is the same as used in the course slides.
- You can load files for a specific exercise using
  - `np.load("filename.npy", allow_pickle=True).item()`

Resources

Resources are available for a subset of questions in form of `.npy` or `.jpg` files. Filenames typeset in `typewriter font` indicate that you can find files in the `materials` folder.

Question 1

A camera has a focal length of 1400 pixels and a principal point at $(750, 520.0)$ pixels. The camera has the rotation:

```
cv2.Rodrigues(np.array([0.2, 0.2, -0.1]))[0]
```

and the translation:

```
np.array([-0.08, 0.01, 0.03])
```

A 3D point in the world coordinate system has coordinates:

```
np.array([-0.38, 0.1, 1.32])
```

What is the projection of this point to the camera's image plane?

a) $[348.52, 627.62]^{\mathrm{T}}$

b) $[458.00, 826.40]^{\mathrm{T}}$

c) $[777.71, 535.32]^{\mathrm{T}}$

d) $[346.97, 626.06]^{\mathrm{T}}$

e) $[555.79, 385.35]^{\mathrm{T}}$

f) $[348.71, 626.32]^{\mathrm{T}}$

g) $[557.53, 383.76]^{\mathrm{T}}$

Question 2

You are given a pairs of corresponding 2D points $\mathbf{p}_{1i}$ and $\mathbf{p}_{2i}$, that are related by a homography

$$\mathbf{q}_{1i} = \mathbf{H}\mathbf{q}_{2i} \tag{2.1}$$

where $\mathbf{q}_{1i}$ and $\mathbf{q}_{2i}$ are homogeneous versions of $\mathbf{p}_{1i}$ and $\mathbf{p}_{2i}$, respectively. For $i = 1, 2, 3, 4$, the corresponding points are given in the following numpy arrays:

```
p1 = np.array([[ 1.45349587e+02, -1.12915131e-01,  1.91640565e+00, -6.08129962e-01],
               [ 1.05603820e+02,  5.62792554e-02,  1.79040110e+00, -2.32182177e-01]])
p2 = np.array([[ 1.3753556,  -1.77072961,  2.94511795,  0.04032374],
               [ 0.30936653,  0.37172814,  1.44007577, -0.03173825]])
```

Which of the following homographies is the one from Equation (2.1)?

You do not need to apply point coordinate normalization.

a) $\mathbf{H} = \begin{bmatrix} 1.0 & -0.511 & -1.13 \\ -1.01 & 0.811 & -0.181 \\ 0.152 & 1.22 & 0.153 \end{bmatrix}$

b) $\mathbf{H} = \begin{bmatrix} 1.0 & -0.482 & -0.254 \\ -0.482 & 0.675 & 0.318 \\ -0.254 & 0.318 & 0.604 \end{bmatrix}$

c) $\mathbf{H} = \begin{bmatrix} 1.0 & -1.01 & 0.152 \\ -0.511 & 0.811 & 1.22 \\ -1.13 & -0.181 & 0.153 \end{bmatrix}$

d) $\mathbf{H} = \begin{bmatrix} 1.0 & 1.18 & 2.73 \\ 1.28 & 1.44 & 1.04 \\ 3.92 & -2.06 & -4.73 \end{bmatrix}$

e) $\mathbf{H} = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$

f) $\mathbf{H} = \begin{bmatrix} 1.0 & 0.0105 & 0.581 \\ -2.17 & 3.31 & -0.53 \\ 1.78 & -1.44 & 0.0171 \end{bmatrix}$

g) $\mathbf{H} = \begin{bmatrix} 1.0 & 0.37 & -3.93 \\ -3.76 & 0.944 & -3.76 \\ 2.92 & 3.83 & 0.858 \end{bmatrix}$

Question 3

What is the minimum number of point correspondences you need to estimate a fundamental matrix?

You can assume nothing is known about the cameras intrinsics or extrincis.

a) 1

b) 2

c) 3

d) 4

e) 5

f) 6

g) 7

h) 8

i) 9

j) 10

k) 11

l) 12

Question 4

A camera has the following intrinsic camera matrix

$$K = \begin{bmatrix} 300 & 0 & 840 \\ 0 & 300 & 620 \\ 0 & 0 & 1 \end{bmatrix}$$

The distortion coefficients are

$$k_3 = -0.2, \quad k_5 = 0.01, \quad k_7 = -0.03.$$

or given as python:

```
K = np.array([[300, 0, 840], [0, 300, 620], [0, 0, 1]], float)
k3 = -0.2
k5 = 0.01
k7 = -0.03
```

You would like to undistort an image taken by this camera.

Which pixel in the distorted image corresponds to the one at pixel

$$\begin{bmatrix} 400 & 500 \end{bmatrix}^{\mathrm{T}}$$

in the undistorted image?

a) $\begin{bmatrix} -105.9 & -132.3 \end{bmatrix}^{\mathrm{T}}$

b) $\begin{bmatrix} 172.4 & 215.5 \end{bmatrix}^{\mathrm{T}}$

c) $\begin{bmatrix} 400.0 & 500.0 \end{bmatrix}^{\mathrm{T}}$

d) $\begin{bmatrix} -311.7 & -389.6 \end{bmatrix}^{\mathrm{T}}$

e) $\begin{bmatrix} -61.3 & -76.6 \end{bmatrix}^{\mathrm{T}}$

f) $\begin{bmatrix} 88.3 & 110.4 \end{bmatrix}^{\mathrm{T}}$

g) $\begin{bmatrix} 742.8 & 593.5 \end{bmatrix}^{\mathrm{T}}$

h) $\begin{bmatrix} 1182.8 & 713.5 \end{bmatrix}^{\mathrm{T}}$

i) $\begin{bmatrix} 277.3 & 346.6 \end{bmatrix}^{\mathrm{T}}$

Question 5

Which topic(s) in the course use non-maximum suppression?

a) BLOB detection, camera calibration, and Harris corners

b) SLAM, and structured light

c) Triangulation, Harris corners, and BLOB detection

d) Triangulation, and BLOB detection

e) SLAM, and BLOB detection

f) Triangulation, and SLAM

g) Camera calibration

h) Camera calibration, and structured light

i) BLOB detection

j) BLOB detection, triangulation, and structured light

k) Triangulation

l) Triangulation, structured light, and Harris corners

m) Harris corners, and camera calibration

n) Structured light, and BLOB detection

o) Camera calibration, and BLOB detection

p) BLOB detection, and Harris corners

q) Harris corners

r) SLAM

s) Structured light

t) Structured light, and Harris corners

Question 6

Harris corner detector. For a small region of a larger image, we have computed the elements of the structure tensor. They are available in `harris.npy` and also presented here:

$g * (I_x^2)$

	0	1	2	3	4
0	8.5	9.1	9.8	10.6	11.5
1	7.4	8.0	8.6	9.4	10.4
2	6.2	6.7	7.3	8.1	9.1
3	5.1	5.4	6.0	6.7	7.7
4	4.1	4.3	4.7	5.4	6.3

$g * (I_y^2)$

	0	1	2	3	4
0	2.1	2.1	2.1	2.1	2.1
1	2.2	2.3	2.4	2.3	2.3
2	2.4	2.5	2.6	2.6	2.5
3	2.5	2.7	2.8	2.8	2.7
4	2.6	2.8	3.0	3.0	2.9

$g * (I_x I_y)$

	0	1	2	3	4
0	0.4	0.7	1.1	1.4	1.7
1	0.2	0.6	1.0	1.3	1.5
2	0.1	0.5	0.9	1.2	1.4
3	0.0	0.4	0.8	1.1	1.3
4	-0.1	0.3	0.7	1.0	1.1

Let $k = 0.06$ and set the threshold to $\tau = 5$.

Does the Harris corner detector find any corners in the image? Corners are specified as (row index, column index).

a) There is a corner at (2, 2).

b) There is a corner at (1, 3).

c) There is a corner at (3, 1).

d) There is a corner at (3, 3).

e) There is a corner at (1, 2).

f) There is a corner at (1, 1).

g) There is a corner at (2, 3).

h) There is a corner at (2, 1).

i) There is a corner at (3, 2).

j) There is no corner in the image.

Question 7

We are given a sequence of ten images of a static scene taken with the same camera, that has been moved gradually between the images.

The first camera has identity rotation and position $\begin{bmatrix}0 & 0 & 0\end{bmatrix}^{\mathrm{T}}$ in the world's coordinate system.

Can we recover the pose of the camera capturing the tenth image in the world's coordinate system using the information contained in the image sequence?

a) We cannot find the pose.

b) We can find the rotation up to an arbitrary rotation and the translation up to an arbitrary scale.

c) We can find the rotation and the translation up to an arbitrary scale.

d) We can find the translation without any ambiguity, but not the rotation.

e) We can find the pose without any ambiguity.

f) We can find the rotation but not the translation.

g) We can find the translation up to an arbitrary scale but not the rotation.

h) We can find the rotation up to an arbitrary rotation, but not the translation.

Question 8

In the context of computer vision, the SIFT descriptor is used to represent local features of images. When finding SIFT keypoints and their descriptors, which of the following steps is *not* a part of the process?

a) Utilization of a transform to detect lines and curves that contribute to the keypoint descriptor.

b) Normalization of the descriptor to enhance invariance to changes in illumination.

c) Calculation of the gradient magnitude and orientation for image regions around the keypoint.

d) Creation of a histogram of gradient orientations for pixels within a region around the keypoint.

e) Division of the region around the keypoint into subregions to capture spatial information.

f) Application of a Gaussian filter to the image at various scales to identify potential keypoints.

g) Assignment of one or more dominant orientations to the keypoint based on local image gradients.

h) Aggregation of the histograms from all subregions to form the final feature descriptor.

i) Rotation of the descriptor according to the dominant gradient orientation to achieve rotation invariance.

Question 9

We have found SIFT keypoints and descriptors in two images, using

```
kp1, des1 = sift.detectAndCompute(im1, None)
kp2, des2 = sift.detectAndCompute(im2, None)
```

These variables (`kp1`, `des1`, `kp2`, `des2`) are stored in the file `sift_data.npy`. You can load them with the following code:

```
sift_data = np.load("sift_data.npy", allow_pickle=True).item()
kp1 = sift_data["kp1"]
des1 = sift_data["des1"]
kp2 = sift_data["kp2"]
des2 = sift_data["des2"]
```

Match the features between the two images using RootSIFT and the ratio test with a ratio of 0.8.

How many matches are there?

a) 0

b) 301

c) 306

d) 319

e) 331

f) 352

g) 365

h) 367

i) 372

j) 373

k) 402

l) 406

m) 419

n) 2000

Question 10

You are given the images `board0.jpg`, `board1.jpg`, `board2.jpg`, `board3.jpg`, and `board4.jpg`, that have been captured with the same camera.

You can assume there is no lens distortion.

Which of the following focal lengths best matches the camera used to take these images?

a) $f = 200$

b) $f = 400$

c) $f = 600$

d) $f = 800$

e) $f = 1000$

f) $f = 1200$

g) $f = 1400$

h) $f = 1600$

i) $f = 1800$

j) $f = 2000$

Question 11

When applying Zhang's algorithm for camera calibration, which of the following statements is true regarding the estimation of the intrinsic camera parameters?

a) The skew is always assumed to be zero.

b) The intrinsic camera parameters are estimated using the right singular vector associated with the largest singular value of the matrix obtained from the homography matrices.

c) Zhang's algorithm can calibrate the camera with a single image of a three-dimensional object.

d) Zhang's algorithm is based on the assumption that all world points lie on a single plane parallel to the image plane.

e) The intrinsic parameters are determined by the ratio of the number of world points to the number of image points.

f) The principal point is assumed to be at the center of the image sensor.

g) The intrinsic camera parameters can be estimated from a single image of the calibration pattern, assuming the skew is zero and the principal point is known.

h) Zhang's algorithm estimates the intrinsic parameters by solving a system of linear equations derived from the image points and their corresponding world points.

Question 12

You are provided with two images, `im1.jpg` and `im2.jpg`, that have been captured with a camera undergoing a pure rotation around its origin.

(a) `im1.jpg`

(b) `im2.jpg`

Estimate the 3×3 matrix (let's call it $\mathbf{A}$) that can relate points between the two images. Given a 2D point in `im2.jpg` in homogeneous coordinates $\mathbf{q}_2$, then using this matrix, $\mathbf{A}\mathbf{q}_2$ should increase our knowledge of where this point is located in `im1.jpg`.

Which matrix below best describes this relation?

a) $\begin{bmatrix} 1.000 & 0.007 & -383.637 \\ 0.068 & 0.972 & -40.847 \\ 0.000 & 0.000 & 0.886 \end{bmatrix}$

b) $\begin{bmatrix} 1.000 & 0.006 & -555.325 \\ 0.069 & 0.973 & -31.463 \\ 0.000 & 0.000 & 1.033 \end{bmatrix}$

c) $\begin{bmatrix} 1.000 & 0.006 & -250.762 \\ 0.066 & 0.656 & -32.941 \\ 0.000 & 0.000 & 0.500 \end{bmatrix}$

d) $\begin{bmatrix} 1.000 & 0.011 & -376.169 \\ 0.100 & 1.587 & -67.834 \\ 0.000 & 0.000 & 0.828 \end{bmatrix}$

e) $\begin{bmatrix} 1.000 & 0.008 & -345.224 \\ 0.062 & 1.042 & -40.518 \\ 0.000 & 0.000 & 0.676 \end{bmatrix}$

f) $\begin{bmatrix} 1.000 & 0.008 & -376.173 \\ 0.071 & 0.808 & -41.341 \\ 0.000 & 0.000 & 1.054 \end{bmatrix}$

g) $\begin{bmatrix} 1.000 & 0.004 & -279.648 \\ 0.043 & 0.751 & -36.961 \\ 0.000 & 0.000 & 0.626 \end{bmatrix}$

Question 13

What is the difference between a fundamental matrix and an essential matrix?

a) The fundamental matrix can be estimated from point correspondences alone, while estimating the essential matrix also requires camera intrinsics

b) An essential matrix defines a relationship between two images, while a fundamental matrix defines a transformation within a single image.

c) The fundamental matrix is related to the extrinsic parameters of a camera, while the essential matrix deals with the intrinsic parameters.

d) The fundamental matrix applies only to pinhole cameras, whereas the essential matrix is for cameras with complex lens systems.

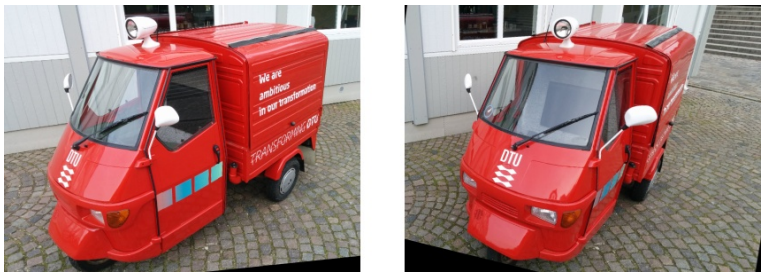
e) Fundamental matrices are for perspective projection models, and essential matrices are for orthographic projection models.

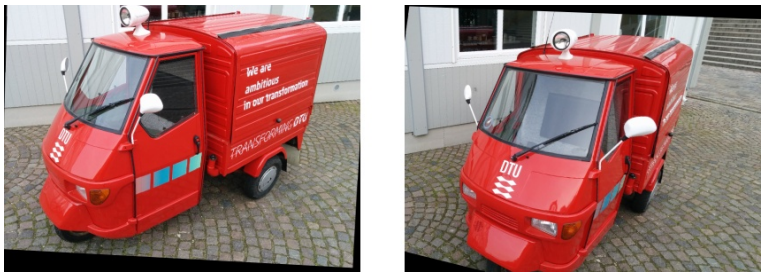
f) An essential matrix is a subset of the fundamental matrix, representing cases with zero translation.

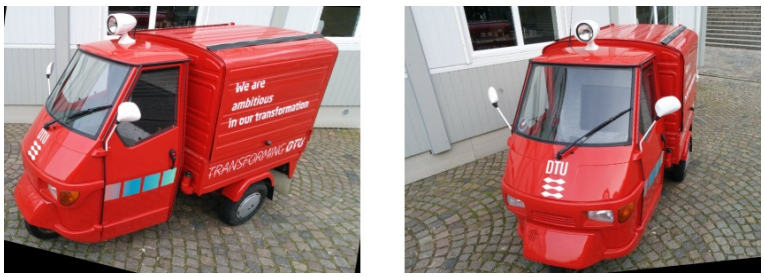
g) Essential matrices are calculated using linear methods, while fundamental matrices require non-linear optimization.

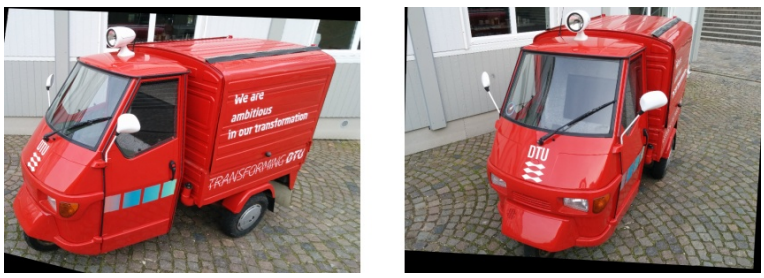
Question 14

Which of the following pairs of images are rectified?

a) 

b) 

c) 

d) 

Question 15

You are given three cameras (1, 2 and 3) that share the same camera matrix `K` and have the following extrinsics. You can copy and paste the following into Python:

```
K = np.array([[300, 0, 840], [0, 300, 620.0], [0, 0, 1]], float)
R1 = cv2.Rodrigues(np.array([-2.3, -0.7, 1.0]))[0]
t1 = np.array([0.0, -1.0, 4.0], float)
R2 = cv2.Rodrigues(np.array([-0.6, 0.5, -0.9]))[0]
t2 = np.array([0.0, 0.0, 9.0], float)
R3 = cv2.Rodrigues(np.array([-0.1, 0.9, -1.2]))[0]
t3 = np.array([-1.0, -6.0, 28.0], float)
```

You observe the same point in all three cameras, but with some noise. The observed points are:

```
p1 = np.array([853.0, 656.0])
p2 = np.array([814.0, 655.0])
p3 = np.array([798.0, 535.0])
```

How far is `p1` from the epipolar line in camera 1 that is induced by `p2`?

a) 37.85

b) 8.937e+06

c) 2.562

d) 1.071e+03

e) 1.075e+03

f) 878.0

g) 79.07

h) 0.1894

i) 9.797e+25

j) 1.133

k) 1.0232e-12

l) 160.7

Question 16

Use all three observations of the point in Question 15 to triangulate the point with using *non-linear* minimization of the reprojection error.

What is the triangulated point?

a) $[0.7672, -3.0802, -1.0808]^{\mathrm{T}}$

b) $[0.2300, -1.4477, -0.0941]^{\mathrm{T}}$

c) $[37.8256, 83.9806, -85.7761]^{\mathrm{T}}$

d) $[91.4675, 205.2109, -205.1722]^{\mathrm{T}}$

e) $[0.2548, -1.5392, -0.1490]^{\mathrm{T}}$

f) $[0.2728, -1.4592, 0.0564]^{\mathrm{T}}$

g) $[-1.8488, -6.9555, 2.0862]^{\mathrm{T}}$

h) $[0.2782, -1.4346, -0.0856]^{\mathrm{T}}$

i) $[-1.8638, -4.2225, 0.3008]^{\mathrm{T}}$

j) $[-1.8852, -4.1820, 0.2940]^{\mathrm{T}}$

k) $[-1.9623, -4.2506, 0.2648]^{\mathrm{T}}$

l) $[0.1617, -1.1047, 0.1597]^{\mathrm{T}}$

m) $[0.2604, -2.6054, -1.0088]^{\mathrm{T}}$

Question 17

You are given a camera with the following extrinsics, $\mathbf{R}$ and $\mathbf{t}$.

You can copy and paste the following into Python:

```
R = cv2.Rodrigues(np.array([-1.9, 0.1, -0.2]))[0]
t = np.array([-1.7, 1.3, 1.5], float)
```

What is the position of the camera?

a) $\begin{bmatrix}-1.6 & -1.9 & -0.7\end{bmatrix}^{\mathrm{T}}$

b) $\begin{bmatrix}-1.3 & 1.2 & -1.9\end{bmatrix}^{\mathrm{T}}$

c) $\begin{bmatrix}-1.7 & 1.3 & 1.5\end{bmatrix}^{\mathrm{T}}$

d) $\begin{bmatrix}1.7 & -1.3 & -1.5\end{bmatrix}^{\mathrm{T}}$

e) $\begin{bmatrix}1.8 & 1.9 & -0.4\end{bmatrix}^{\mathrm{T}}$

f) $\begin{bmatrix}1.6 & 1.9 & 0.7\end{bmatrix}^{\mathrm{T}}$

g) $\begin{bmatrix}1.3 & -1.2 & 1.9\end{bmatrix}^{\mathrm{T}}$

h) $\begin{bmatrix}-1.8 & -1.9 & 0.4\end{bmatrix}^{\mathrm{T}}$

Question 18

Use the cameras defined in Question 15. The following 3D point has been found in the reference frame of camera 3.

```
np.array([[-0.38], [0.1], [1.32]])
```

What is this 3D point in the reference frame of camera 2?

a) $\begin{bmatrix} -0.10 & -6.14 & 29.03 \end{bmatrix}^{\mathrm{T}}$

b) $\begin{bmatrix} 9.57 & 16.58 & -19.56 \end{bmatrix}^{\mathrm{T}}$

c) $\begin{bmatrix} 2.07 & -16.38 & 12.90 \end{bmatrix}^{\mathrm{T}}$

d) $\begin{bmatrix} -0.38 & 0.10 & 1.32 \end{bmatrix}^{\mathrm{T}}$

e) $\begin{bmatrix} 4.41 & 13.22 & -13.40 \end{bmatrix}^{\mathrm{T}}$

f) $\begin{bmatrix} 25.80 & -16.10 & 20.56 \end{bmatrix}^{\mathrm{T}}$

g) $\begin{bmatrix} -4.48 & 32.48 & -10.36 \end{bmatrix}^{\mathrm{T}}$

h) $\begin{bmatrix} -0.38 & 0.10 & 1.32 \end{bmatrix}^{\mathrm{T}}$

i) $\begin{bmatrix} 4.41 & 13.22 & -13.40 \end{bmatrix}^{\mathrm{T}}$

j) $\begin{bmatrix} 2.19 & -5.26 & -17.78 \end{bmatrix}^{\mathrm{T}}$

Question 19

We are using RANSAC to estimate a homography matrix. At iteration number 153 we find a model where 465 out of 1177 point matches are inliers, which is the highest number of inliers we have observed so far.

What is the smallest number of iterations we need to run in total in order to be 90% sure that a model is found that is fitted to only inliers?

a) 17

b) 33

c) 38

d) 94

e) 111

f) 128

g) 187

h) 625

i) 3879

Question 20

You are estimating a projection matrix with RANSAC.

The keypoints have been found with an algorithm that are known to be normally distributed and have a standard deviation of $\sigma_x = \sigma_y = 1.4$ pixels. To measure how well a given projection matrix fits, you use the squared euclidean reprojection distance. You would like to correctly identify 95% of true inliers.

What should you set your squared threshold (τ^2) to?

a) 2.30

b) 2.74

c) 3.01

d) 3.43

e) 3.60

f) 3.79

g) 4.25

h) 5.31

i) 5.38

j) 6.45

k) 7.53

l) 8.39

m) 9.04

n) 9.28

o) 11.74

p) 12.89

q) 12.99

r) 18.05

Question 21

You are performing structured light surface reconstruction with phase shifting. The number of periods in the primary pattern is 40, and the number of periods in the secondary pattern is 41. In a specific pixel, you observe the following values for the intensity of the primary and secondary patterns:

```
primary = np.array([12, 9, 10, 13, 18, 25, 33, 40, 46, 49, 48, 45, 39, 31, 23, 17])
secondary = np.array([15, 29, 43, 49, 43, 29, 15, 10])
```

What is the unwrapped θ in this pixel?

a) 4.3655

b) 4.8471

c) 1.3216

d) 4.2872

e) 8.0

f) 0.3969

g) 1.3299

h) 0.8862

i) 6.0514

Question 22

A camera is moved to five distinct poses, starting at $\begin{bmatrix}0 & 0 & 0\end{bmatrix}^{\mathrm{T}}$ with identity rotation. We are given the four essential matrices relating consecutive poses. Can we find the fifth pose of the camera *purely* from these essential matrices?

a) We can find the translation up to an arbitrary scale but not the rotation.

b) We can find the pose without any ambiguity.

c) We can find the rotation up to an arbitrary rotation and the translation up to an arbitrary scale.

d) We can find the rotation up to an arbitrary rotation, but not the translation.

e) We can find the translation without any ambiguity, but not the rotation.

f) We can find the rotation but not the translation.

g) We can find the rotation and the translation up to an arbitrary scale.

h) We cannot find the pose.