ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Computer Vision
AS 2019

Prof. Marc Pollefeys *Prof. Luc Van Gool* *Prof. Vittorio Ferrari*

Final Exam

20 January 2020

First and Last name: ______________________________

ETH number: ______________________________

Signature: ______________________________

General Remarks

- Remove all material from your desk which is not allowed by examination regulations. The following materials are allowed for this exam:
  - exam questionnaire & blank paper (both provided by us)
  - ruler/square & pen (pencil and red color pens are not allowed)
- Check that your exam questionnaire is complete.
- Fill in your first and last name and your ETH number and sign the exam. Place your student ID in front of you.
- You have **2** hours for this exam.
- Put your name and ETH number on top of each sheet.
- Please do not use a pencil or red color pen to write your answers.
- You may provide at most one valid answer per question. Invalid solutions must be canceled out clearly.

	Topic	Max. Points	Points Achieved	Visum
1	Feature extraction	13		
2	RANSAC	12		
3	Epipolar geometry	14		
4	Optical flow	17		
5	Stereo matching	14		
6	Object class recognition	14		
7	Image segmentation	16		
Total		100		

Grade: ....

Question 1: Feature Extraction (13 pts.)

a) Describe SSD (sum-of-squared-differences) mutual nearest neighbors matching, including the definition of the SSD function. **3 pts.**

ANSWER

Writes the SSD function correctly - 1pt. Each feature from the first image is associated to its closest feature from the second image (1pt) and vice versa (1pt). The relationship holds both ways.

ANSWER

b) Fill out the following table comparing the three different feature descriptors and their invariance properties for Harris Coner Detector, SIFT and MSER (Maximally Stable Extremal Regions). Write **Y** if the invariance applies, **N** if it does not. **3 pts.**

Geometric Invariance Type	Harris Corner Detector	MSER	SIFT
translation			
rotation			
scale			
affine			

ANSWER

Geometric Invariance Type	Harris Corner Detector	MSER	SIFT
translation	Y	Y	Y
rotation	Y	Y	Y
scale	N	Y	Y
affine	N	Y	N

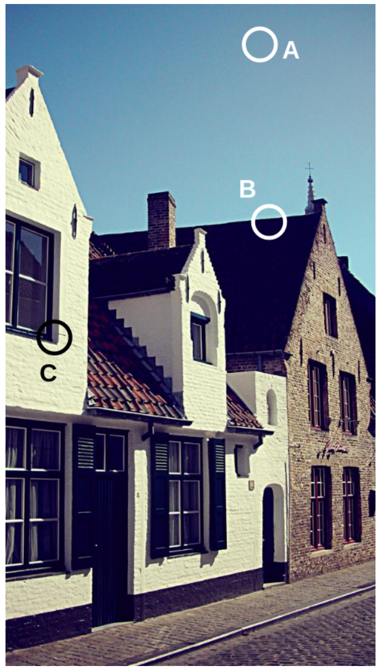
ANSWER

c) Briefly describe the main steps of the SIFT algorithm. **5 pts.**

1pt for each: 1. Scale-space Extrema Detection 2. Keypoint Localization 3. Orientation Assignment 4. Keypoint Descriptor 5. Keypoint Matching

ANSWER

d) Choose which patch A/B/C describes the most useful feature in the image below and explain why that is so. **2 pts.** ANSWER ANSWER ANSWER ANSWER ANSWER ANSWER ANSWER



patch C - a corner - the most discriminative (gradient changes in every direction)

ANSWER

Question 2: RANSAC (12 pts.)

a) Fill in the following pseudo-code for the RANSAC algorithm when fitting a model that requires at least N samples to a dataset D. The number of iterations K is fixed and the inlier threshold is T. Please explain the intermediate steps. **6 pts.**

Input: $data, N, K, T$
Output: $best\_model$
$best\_model \leftarrow None$;
$best\_num\_inliers \leftarrow 0$;
for $i = 0 \rightarrow$ .... **do**
  $subset \leftarrow$ ....;
  $model \leftarrow$ ....;
  $residuals \leftarrow$ ....;
  $is\_inlier \leftarrow$ ....;
  $num\_inliers \leftarrow sum(is\_inlier)$;
  if .... **then**
    $best\_model \leftarrow model$;
    $best\_num\_inliers \leftarrow num\_inliers$;
  end
end

ANSWER

Input: $data, N, K, T$
Result: $best\_model$
$best\_model \leftarrow None$;
$best\_num\_inliers \leftarrow 0$;
for $i = 0 \rightarrow K$ **do**
  $set \leftarrow sample(data, N)$;
  $model \leftarrow fit(set)$;
  $residuals \leftarrow compute\_residuals(data, model)$;
  $is\_inlier \leftarrow (residuals \leq T)$;
  $num\_inliers = sum(is\_inlier)$;
  if $num\_inliers > best\_num\_inliers$ **then**
    $best\_model \leftarrow model$;
    $best\_num\_inliers \leftarrow num\_inliers$;
  end
end

ANSWER

b) Choose the right formula for p, the probability of having at least one subset of size N full of inliers after M iterations, given the inlier ratio r.

Prove it. To simplify the equations, you should suppose that the sampling is done with replacement. Finally, derive an equation for M_0, the number of iterations required so that

A. $p = 1 - (1 - r^M)^N$ B. $p = (1 - r^N)^M$ C. $p = 1 - (1 - r^N)^M$ D. $p = r^{MN}$

the probability p is at least p_0. In order to get the full score, detail all intermediate steps. **6 pts.**

Hint: compute p_s, the probability of sampling only inliers in a subset and use it to compute p_n, the probability of sampling only subsets with at least one outlier for M iterations.

ANSWER

We have the following equations: $p_s = r^N$, $p_n = (1-p_s)^M$, $p = 1-p_n$, *so* $p = 1-(1-r^N)^M$. *To compute* M_0*, the equation from above can be rewritten as follows* $(1 - r^N)^M = 1 - p$ *which finally gives* $M_0 = \frac{\log(1-p_0)}{\log(1-r^N)}$.

ANSWER

Question 3: Epipolar Geometry (14 pts.)

Assume the relative motion of a calibrated camera exhibits no rotation. Thus, the Essential matrix of this motion can be expressed as

$$\mathbf{E} = \hat{\mathbf{t}}\,,$$

where $\hat{t}$ is the skew symmetric form of the translation. That is, if $\mathbf{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$, then

$$\hat{\mathbf{t}} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}.$$

Under this condition we are interested in determining how the camera could translate (i.e., which values t may take) such that the epipolar lines on any of the two cameras are **parallel**. Recall that any two image lines $\mathbf{l} = [l_x, l_y, l_z]$ and $\mathbf{k} = [k_x, k_y, k_z]$ are parallel if $l_x k_y = k_x l_y$ (that is, if the slopes of the lines in the image are equal), and that $\mathbf{l} = \mathbf{E}\mathbf{p}$ is the epipolar line in the second image corresponding to the image point $\mathbf{p}$ in the first image.

Assume you are given two **distinct** points in the first image $\mathbf{p} = [p_x, p_y, 1]^\intercal$ and $\mathbf{q} = [q_x, q_y, 1]^\intercal$, with respective epipolar lines $\mathbf{l_p}$ and $\mathbf{l_q}$ in the second image. Answer the following:

a) If we had $\mathbf{t} = [0, 0, 1]^\intercal$ (that is, pure forward motion), would the epipolar lines $\mathbf{l_q}$ and $\mathbf{l_p}$ be parallel? Explain your answer formally in terms of $\mathbf{p}$ and $\mathbf{q}$. **3 pts.**

ANSWER

No, in general not. This would yield

$$\mathbf{E} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

and thus $\mathbf{E}p = l_p$ *and* $\mathbf{E}q = l_q$ *would not be parallel since* $l_p = [p_x, p_y, 0]$ *and* $l_q = [q_x, q_y, 0]$, *and it follows that in general* $p_y q_x \neq q_y p_x$. *However, for all points with* $p_y q_x = q_y p_x$, *the epipolar lines would be parallel.*

ANSWER

b) If we had pure sideways motion ($\mathbf{t} = [1, 0, 0]$), would the epipolar lines be parallel? Again, use $\mathbf{p}$ and $\mathbf{q}$ to explain your answer formally. **3 pts.**

ANSWER

In this case, yes. We now have

$$\mathbf{E} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} ,$$

and thus $l_p = [0, -1, p_y]$ *and* $l_q = [0, -1, q_y]$. *The slopes of these two lines are equal:* $(-1) = (-1)$.

ANSWER

c) Give the conditions under which $\mathbf{t}$ will always yield parallel epipolar lines. Explain in terms of $\mathbf{p}$ and $\mathbf{q}$. **4 pts.**

ANSWER

Intuitively, if we translate along the image plane (that is, $t_z = 0$*) of the first image, we will always have the epipole at infinity, yielding parallel epipolar lines. We may write out for* p:

$$\mathbf{E}\ p = l_p = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} = \begin{bmatrix} -t_z p_y + t_y \\ t_z p_x - t_x \\ -t_y p_x + t_x p_y \end{bmatrix} ,$$

and similarly for q. *The slope of these epipolar lines is given by*

$$\frac{l_{p,y}}{l_{p,x}} = \frac{t_z p_x - t_x}{-t_z p_y + t_y} \quad \textit{and} \quad \frac{l_{q,y}}{l_{q,x}} = \frac{t_z q_x - t_x}{-t_z q_y + t_y} ,$$

which in general can only be equal if $t_z = 0$.

ANSWER

d) Let us now assume that the calibration and the relative motion of the camera is known, but not their absolute location in the 3D world coordinate frame. Up to which transformation can the scene be reconstructed in 3D? **2 pts.**

ANSWER

Euclidean or proper euclidean or up to rotation+translation

ANSWER

e) In case the calibration matrices and the relative rotation of the cameras are not known, but the infinity homography can be determined, up to which transformation can the scene be reconstructed in 3D? **2 pts.**

Up to an unknown 3D affine transformation

ANSWER

Question 4: Optical Flow (17 pts.)

Notation: In the following, we use $I(x, y, t)$ to denote the brightness of a pixel (x, y) at time t, use u and v to denote the velocity of a pixel or an object in x and y directions, use $\triangledown I$ to denote the gradient of I, i.e., $\triangledown I = [I_x, I_y, I_t]^T = [\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t}]^T$, where the first two elements form the spatial gradient and the last element represent the temporal gradient.

a) Which one of the following choices is the optical flow constraint equation? ____ **2 pts.**

A. $\triangledown I = \mathbf{0}$ B. $\triangledown I^T[u, v, 1]^T = 0$ C. $\triangledown I^T[u, v, -1]^T = 0$
D. $\triangledown I^T[1, 1, 1]^T = 0$ E. $\triangledown I^T[v, u, 1]^T = 0$ F. $\triangledown I^T[v, u, -1]^T = 0$

ANSWER

B

ANSWER

b) Derive the optical flow constraint equation. **3 pts.**
Hint: start from the assumption $I(x, y, t) = I(x + \mathrm{d}x, y + \mathrm{d}y, t + \mathrm{d}t)$, and then use the first-order Taylor polynomial (linear approximation).

ANSWER

$$I(x, y, t) = I(x + \mathrm{d}x, y + \mathrm{d}y, t + \mathrm{d}t) = I(x, y, t) + \frac{\partial I}{\partial x}\mathrm{d}x + \frac{\partial I}{\partial y}\mathrm{d}y + \frac{\partial I}{\partial t}\mathrm{d}t$$
$$\frac{\partial I}{\partial x}\mathrm{d}x + \frac{\partial I}{\partial y}\mathrm{d}y + \frac{\partial I}{\partial t}\mathrm{d}t = 0$$
$$\frac{\partial I}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial I}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}t} + \frac{\partial I}{\partial t} = 0$$
$$I_x u + I_y v + I_t = 0$$

ANSWER

c) Explain what is the aperture problem in optical flow. **2 pts.**

ANSWER

The aperture problem in optical flow means one equation with two unknown variables (u and v) and therefore the equation cannot be solved.

ANSWER

d) Suppose a pixel (x_0, y_0) has the following property, $I(x_0, y_0, t) = 0, \forall t$. Judge the correctness of the following statements. **Cross** the box if the statement is correct. **3 pts.**

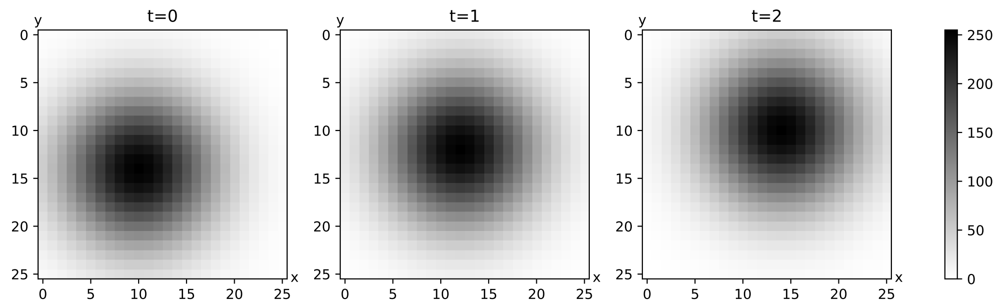
☐ The spatial gradient of this pixel is always a zero vector.
☐ The temporal gradient of this pixel always equals to zero.
☐ The brightness of this pixel does not change.
☐ This pixel cannot belong to a moving object.
☐ The velocity of this pixel is always perpendicular to its spatial gradient.
☐ The velocity of this pixel is always parallel to its spatial gradient.

ANSWER

The 2nd, 3rd and 5th statements are right while the other three are wrong.

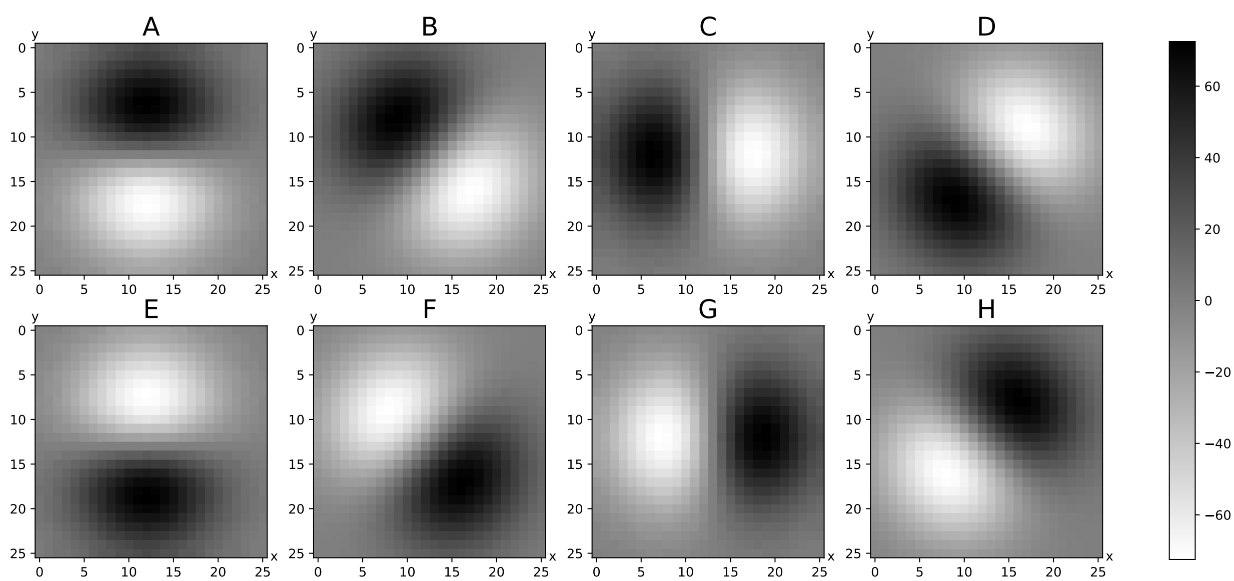
ANSWER

The figure below shows three consecutive frames taken from a video. It demonstrate the movement of the big black dot pointing to the upper right direction. In the following, our goal is to estimate its velocity using optical flow.



e) Match the correct figure (A to H) for the following gradients. **3 pts.**

$I_x(\cdot,\cdot,1)$	$I_y(\cdot,\cdot,1)$	$I_t(\cdot,\cdot,1)$



ANSWER

C, A, H.

ANSWER

f) In order to estimate the moving velocity of the big black dot at time $t = 1$, a 2×2 window containing 4 pixels is selected. The following table gives their gradients.

Pixel	$I_x(\cdot,\cdot,1)$	$I_y(\cdot,\cdot,1)$	$I_t(\cdot,\cdot,1)$
1	15.5	-6.5	-44
2	13.5	-7.5	-42
3	18	-9	-52.5
4	15.5	-10	-50

i) Write down the two linear equations with two variables using pixel 1 and pixel 2, and then find the solution. **2 pts.**

ANSWER

$$\begin{cases} 15.5u - 6.5v = 44 \\ 13.5u - 7.5v = 42 \end{cases}, \begin{cases} u = 2 \\ v = -2 \end{cases}$$

ANSWER

The solution found in (i) may not work for pixel 3 and pixel 4. In order to fully utilize the information of all of these 4 pixels, we can use Lukas-Kanade flow, i.e., to reformulate them to a least squares problem. In general, an unrestricted least squares problem $\min_\beta ||X\beta - b||^2$ has a close-form solution $\beta = (X^TX)^{-1}X^Tb$, where $X \in \mathbb{R}^{n \times d}$, $\beta \in \mathbb{R}^d$, $b \in \mathbb{R}^n$.

ii) What is β in our case? Write down the matrix X and vector b using the numbers in the table above. **2 pts.**

ANSWER

$$\beta = \begin{bmatrix} u \\ v \end{bmatrix}, X = \begin{bmatrix} 15.5 & -6.5 \\ 13.5 & -7.5 \\ 18 & -9 \\ 15.5 & -10 \end{bmatrix}, b = \begin{bmatrix} 44 \\ 42 \\ 52.5 \\ 50 \end{bmatrix}$$

ANSWER

Question 5: Stereo matching (14 pts.)

a) To recover the disparity map from a set of images we need to calculate the offset $d(x, y)$ of matching pixels

$$x' = x + d(x, y),\ y' = y,$$

between the left and right image. Assuming we use the winner-takes-all technique for this purpose, and given a set of disparity values and a window size, describe a search pseudo algorithm that iterates over all pixels to estimate the optimal disparity per pixel. Note that, you are only allowed to use for loops and numerical operations. **5 pts.**

ANSWER

For each pixel (x, y), for each disparity value d
$SSD = 0$
For each pixel (i, j) in the window: $SSD = SSD + (I1(x+i, y+j) - I2(x+d+i, y+j))^2$
Optimal disparity is the one with the smallest SSD.

ANSWER

b) How does increasing the window size affect the final result in previous cases? **2 pts.**

ANSWER

The bigger the filter the smoother the result. However, for very large window sizes we might lose the local details.

ANSWER

c) Disparity computation is often solved by minimizing an energy function. Provide an appropriate energy function for the graph-cut technique. **5 pts.**

ANSWER

In graph-cut, computing the disparity is formulated as a graph labeling problem. Each pixel corresponds to a graph node and each disparity to a label. The goal is to find a labeling $f : P \rightarrow L$ *which minimizes the energy function*

$$E(f) = E_{data}(f) + \lambda E_{smooth}(f) = \sum_{p \in P} D_p(f_p) + \lambda \sum_{p,q \in \mathcal{N}} S(f_p, f_q).$$

P *represents the set of pixels and* L *represents a discrete set of labels corresponding to different disparities.* $D_p(f_p)$ *is the cost of assigning label* f_p *to pixel* p *and it is given by the SSD value calculated for the disparity corresponding to label* f_p. $\mathcal{N}$ *is the set of neighboring pixels and* $S(f_p, f_q)$ *is the cost of assigning labels* f_p *and* f_q *to neighboring pixels* p *and* q.

ANSWER

d) How does the graph-cut technique compare with the winner-takes-all technique? **2 pts.**

ANSWER

Graph labeling problem vs energy minimization, takes into account constraints about neighboring pixels vs doesn't, etc.

ANSWER

Question 6: Object Class Recognition (14 pts.)

a) Define the tasks of object classification and object detection. Explain the difference. What is the expected output in each task? **4 pts.**

ANSWER

Classification: Assign input vector to one of several classes, e.g. "is there a car in this image?" Answer is class label.

Detection: in addition to a class label, you need also to answer "where is the car?" We need localization, thus answer is precise object location (bounding box, segmentation, etc). Note this is not restricted to proposal-based approaches. You can do it with a sliding-window mechanism, with Hough Forests, with local feature clustering, or even just by wild random guessing.

ANSWER

b) Describe pros and cons of bag-of-words image representation. **5 pts.**

ANSWER

Pros:

- *Flexible to geometry / deformations / viewpoint*
- *Compact summary of image content*
- *Provides vector representation for sets (bags to be precise)*
- *Empirically good recognition results in practice*

Cons:

- *Basic model ignores geometry, can be verified afterwards, or embed within descriptors*
- *Background and foreground mixed when bag covers whole image*
- *Interest points or sampling: no guarantee to capture object-level parts*
- *Optimal vocabulary formation remains unclear*

ANSWER

c) Explain the main idea behind boosting. Describe each step of the AdaBoost algorithm: initialization, update at each iterations, and how the final strong classifier is built. **5 pts.**

ANSWER

Boosting: Build a strong classifier by combining many weak classifiers, which need only be better than chance. Boosting is based on sequential learning process, i.e. at each iteration, add a weak classifier.

Initialization: Start with uniform weights on training examples.

Update: Evaluate weighted error for each feature, pick best. Then re-weight the examples: more weight for incorrectly classified, and less weight for correctly classified.

Final classifier is a combination of the weak ones, weighted according to the error they had.

ANSWER

Question 7: Image Segmentation (16 pts.)

a) What is the fundamental difference between image classification and image segmentation tasks? Why can't we just use image classification algorithms in the latter case? **3 pts.**

ANSWER

Image segmentation = classify each pixel of the image: we need to know not only what objects are on the image, but also their exact locations (defined by the segmentation mask).

ANSWER

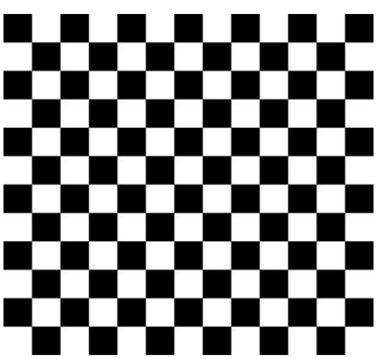
b) Provide two real-world examples of the tasks where image segmentation algorithms are used, and describe why they cannot be replaced by simple image classification models. **4 pts.**

ANSWER

E.g., self-driving (semantic image segmentation for scene understanding, need to know where the pedestrians, vehicles and other objects are located) or portrait segmentation (separate the foreground from the background, need to locate the borders).

ANSWER

c) Suppose that we have the following black-and-white 1200×1300 pixel image of a checkerboard, and we want to apply the Mean-Shift algorithm to segment it. Provide the results of the segmentation for each value of h (radius of the spherical window) and ϵ (convergence threshold). Do not smooth the image and use the RGB color space instead of the L*a*b*. **5 pts.**



ANSWER

Since the Mean-Shift segmentation algorithm is working in the color domain, all pixels will be mapped to two points: (0,0,0) and (255, 255, 255). If the radius h is less than 255, the *algorithm will result in two clusters (for black and white points, respectively), otherwise - in one single cluster. ϵ is not affecting the results since the value of the mean of all points inside the spherical window is constant in all cases.*

ANSWER

d) Recall the EM image segmentation algorithm. As you have already learned, a good initialization is very important to get proper segmentation results using this method. Suppose that you have initialized all Gaussian components with identical values. How this will affect the final segmentation results? Use formulas to illustrate your explanation. **4 pts.**

ANSWER

All resulting clusters will be also identical - will have the same center and covariance matrices. During the E-step, the probability that the data point x_k was generated by mixture i is the same for all Gaussian components if their parameters $\theta_i = (\mu_i, V_i)$ and α_i are identical:

$$P(i \mid x_k, \mu_i, V_i) = \frac{\alpha_i P(x_k \mid \mu_i, V_i)}{\sum_{k=1}^{K} \alpha_k P(x_k \mid \mu_k, V_k)}.$$

Therefore, during the M-step, the new values of the variables μ_i, V_i and α_i will be the same:

$$\alpha_i^{new} = \frac{1}{N} \sum_{k=1}^{N} P(i \mid x_k, \mu_i, V_i),$$

$$\mu_i^{new} = \frac{\sum_{k=1}^{N} x_k P(i \mid x_k, \mu_i, V_i)}{\sum_{k=1}^{N} P(i \mid x_k, \mu_i, V_i)},$$

$$V_i^{new} = \frac{\sum_{k=1}^{N} (x_k - \mu_i^{new})(x_k - \mu_i^{new})^T P(i \mid x_k, \mu_i, V_i)}{\sum_{k=1}^{N} P(i \mid x_k, \mu_i, V_i)}.$$

Thus, repeating these two steps will always lead to the same updates to all Gaussian components, and they will be identical at each iteration.

ANSWER

Question 8: Stereo matching (14 pts.)

a) To recover the disparity map from a set of images we need to calculate the offset $d(x, y)$ of matching pixels

$$x' = x + d(x, y), \; y' = y,$$

between the left and right image. Assuming we use the winner-takes-all technique for this purpose, and given a set of disparity values and a window size, describe a search pseudo algorithm that iterates over all pixels to estimate the optimal disparity per pixel. Note that, you are only allowed to use for loops and numerical operations. **5 pts.**

ANSWER

For each pixel (x, y), for each disparity value d
$SSD = 0$
For each pixel (i, j) in the window: $SSD = SSD + (I1(x+i, y+j) - I2(x+d+i, y+j))^2$
Optimal disparity is the one with the smallest SSD.

ANSWER

b) How does increasing the window size affect the final result in previous cases? **2 pts.**

ANSWER

The bigger the filter the smoother the result. However, for very large window sizes we might lose the local details.

ANSWER

c) Disparity computation is often solved by minimizing an energy function. Provide an appropriate energy function for the graph-cut technique. **5 pts.**

ANSWER

In graph-cut, computing the disparity is formulated as a graph labeling problem. Each pixel corresponds to a graph node and each disparity to a label. The goal is to find a labeling $f : P \rightarrow L$ *which minimizes the energy function*

$$E(f) = E_{data}(f) + \lambda E_{smooth}(f) = \sum_{p \in P} D_p(f_p) + \lambda \sum_{p,q \in \mathcal{N}} S(f_p, f_q).$$

P *represents the set of pixels and* L *represents a discrete set of labels corresponding to different disparities.* $D_p(f_p)$ *is the cost of assigning label* f_p *to pixel* p *and it is given by the SSD value calculated for the disparity corresponding to label* f_p. $\mathcal{N}$ *is the set of neighboring pixels and* $S(f_p, f_q)$ *is the cost of assigning labels* f_p *and* f_q *to neighboring pixels* p *and* q.

ANSWER

d) How does the graph-cut technique compare with the winner-takes-all technique? **2 pts.**

ANSWER

Graph labeling problem vs energy minimization, takes into account constraints about neighboring pixels vs doesn't, etc.

ANSWER