SRM INSTITUTE OF SCIENCE AND TECHNOLOGY, RAMAPURAM

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ANSWER KEY SUBMISSION

Date of Exam & Session	**06/11/2023 & AN**	**Category of Exam**	**CLA3**
Course Name	**Computer Vision**	**Course Code**	**18CSE390T**
Name of the Faculty submitting	**Dr.C.G.BALAJI**	**Date of submission of Answer Key**	**08/11/2023**
Department to which the Faculty belongs to	**CSE**	**Total Marks**	**50**

PART A (10 x 1 = 10)
ANSWER ALL THE QUESTIONS

Q.No.	MCQ Questions	Marks
1	**The problem of determining a point's 3D position from a set of corresponding image locations and known camera positions is ___.** a. **Triangulation** b. Factorization c. Bundle adjustment d. Orthographic projection	1
2	**The_____ refers to the fact that motion estimation is highly ambiguous when the observation window is very small.** a. panoptic segmentation **b. aperture problem** c. geometric reasoning d. calibration error	1
3	**One if the useful property of Fourier transforms is that convolution in the spatial domain corresponds to ____ in the Fourier domain** a. addition b. subtraction c. **multiplication** b. division	1
4	**Bundle adjustment is now the standard method of choice for most ______ problems and is commonly applied to problems with hundreds of weakly calibrated images and tens of thousands of points.** a. motion-from-structure **b. structure-from-motion** c. motion-from-motion d. structure-from-structure	1
5	**Which of the following can also be computed on the basis of line matches alone?** a. **Trifocal Tensor** b. Tensor Flow c. Stack overflow d. both Tensor Flow and Stack overflow	1
6	**____ technique can be used to estimate a series of rotation matrices and focal lengths, which can be chained together to create large panoramas.** a. bundle adjustment b. Parallax removal **c. Gap closing** d. Composting	1
7	**Before we can register and align images, we need mathematical relationships that ______ from one image to another.** a. align an image **b. map pixel coordinates** c. rotate pixel d. compare image	1
8	**In ________, images are translated, optionally rotated and scaled.** a. panography b. cryptography	1

	c. photography **d. cartography**	
9	**An alternative to using homography or 3D motions to align images is to first warp the images into ______ and then use a pure translational model to align them.** a. Spherical coordinates **b. Cylindrical coordinates** c. Planar coordinates d. Vector coordinates	1
10	**If the user takes images in sequence so that each image overlaps its predecessor and also specifies the first and last images to be stitched, bundle adjustment combined with the process of _____ can be used to automatically assemble a panorama.** **a. topology inference** b. mean difference c. sum of squared difference d. change in image	1

PART B (4x4= 16)

ANSWER ANY FOUR OUT OF SIX QUESTIONS

Q.No	Questions	Marks
11	**Why does the Essential Matrix (E) change into the Fundamental Matrix (F)? Discuss in brief.** The Essential Matrix (E) and the Fundamental Matrix (F) are both used in computer vision to relate corresponding points in stereo images. However, they differ in that E assumes that the cameras are calibrated, while F does not. **Essential Matrix** The essential matrix is a 3x3 matrix that relates corresponding points in two calibrated cameras. It is defined as: **E = K'TK** where K and K' are the camera calibration matrices of the two cameras, and T is the translation vector between the two cameras. **Fundamental Matrix** The fundamental matrix is a 3x3 matrix that relates corresponding points in two uncalibrated cameras. It is defined as: **F = HK'** where H is a 3x3 homography matrix that maps points in the first camera to points in the second camera. **Why does E change into F?** The essential matrix changes into the fundamental matrix when the cameras are uncalibrated. This is because the homography matrix H captures the effects of both the rotation and translation between the cameras, as well as the intrinsic parameters of the cameras. In other words, the fundamental matrix is a more general form of the essential matrix that can be used to relate corresponding points in any two cameras, regardless of whether or not they are calibrated.	4
12	**Give the equation and briefly explain the components of Schur complement.** The Schur complement is a way of expressing the inverse of a submatrix of a block matrix in terms of the inverses of the other submatrices. It is a useful tool for solving linear systems of equations and for matrix inversion. **Equation** The Schur complement of a matrix A in a block matrix M is defined as: S = M / A = D - CA-1B where: • M is a block matrix of the form: M = [A B; C D] • A is a square matrix • B and C are matrices of the same dimensions • D is a square matrix **Components** The components of the Schur complement equation are: • A: The matrix to be complemented. • B: A matrix of the same dimensions as A.	4

	• C: A matrix of the same dimensions as B. • D: A matrix of the same dimensions as A. • S: The Schur complement of A in M. **Brief Explanation** The Schur complement S is the inverse of the submatrix D of M, after the submatrix A has been eliminated. The matrices B and C are used to compute the coefficients of the elimination process. **Uses of Schur Complement** The Schur complement is used in a variety of applications, including: • Solving linear systems of equations • Matrix inversion • Optimization • Statistics • Control theory	
13	**What is self-calibration? What are the basic assumptions taken by any self-calibration method?** Self-calibration, also known as camera auto-calibration, is the process of determining the intrinsic parameters of a camera without the need for external calibration objects or prior knowledge of the camera's internal structure. This is in contrast to traditional calibration methods, which require a calibration grid or other known object to be placed in the scene in order to determine the camera's parameters. **Basic Assumptions of Self-Calibration Methods** Self-calibration methods typically rely on certain assumptions about the camera and the scene in order to estimate the intrinsic parameters. These assumptions include: 1. Pinhole Camera Model: The camera is assumed to be a simple pinhole camera, which means that all light rays from a point in the scene pass through a single point on the camera's focal plane. 2. Rigid Scene: The scene being imaged is assumed to be rigid, meaning that the relative positions of the points in the scene do not change during the imaging process. 3. Sufficient Motion: The camera is assumed to undergo sufficient motion during the imaging process, meaning that the relative positions of the camera and the scene change significantly. This allows the self-calibration algorithm to extract enough information from the images to estimate the camera's parameters. 4. No Lens Distortion: The lens is assumed to be distortion-free, or the distortion is assumed to be known and accounted for. 5. No Noise: The images are assumed to be noise-free, or the noise is assumed to be negligible. In practice, these assumptions may not always be perfectly satisfied. However, self-calibration methods can still be effective in estimating the camera's parameters even under these conditions.	4
14	**Explain the Projection from 3D to Spherical coordinate.** Spherical coordinates are a system for representing a point in three-dimensional space. They are defined by three values: • Radial distance: The distance from the origin to the point. • Polar angle: The angle between the positive z-axis and the line connecting the origin to the point. • Azimuthal angle: The angle between the positive x-axis and the projection of the line connecting the origin to the point onto the xy-plane. To project a point from 3D to spherical coordinates, we first need to calculate the radial distance. This is simply the distance from the origin to the point: **r = sqrt(x^2 + y^2 + z^2)** Next, we need to calculate the polar angle. This is the angle between the positive z-axis and the line connecting the origin to the point. To calculate this angle, we use the following formula: **theta = arctan(sqrt(x^2 + y^2) / z)** Finally, we need to calculate the azimuthal angle. This is the angle between the positive x-axis and the projection of the line connecting the origin to the point onto the xy-plane. To calculate this angle, we use the following formula: **phi = arctan(y / x)** Once we have calculated these three values, we can represent the point in spherical coordinates as (r, theta, phi).	4
15	**What are the various difficulties faced while recognizing Panoramas?** Recognizing panoramas is a challenging task due to several factors, including: • Image Matching: Finding corresponding points between different images can be difficult, especially	4

in areas with repetitive textures or low contrast.

- Image Registration: Aligning multiple images to create a seamless panorama requires accurate registration of overlapping regions. This can be challenging due to geometric distortions, parallax errors, and motion blur.
- Illumination Changes: Changes in lighting conditions between images can make it difficult to match features and create a consistent visual appearance.
- Object Occlusions: Objects that are occluded in one image may be visible in another, making it difficult to stitch the images together seamlessly.
- Computational Complexity: Stitching together a large panorama can be computationally expensive, especially for high-resolution images.

Here are some additional challenges that can be faced when recognizing panoramas:

- Varying Exposure: Images taken with different exposure settings can have different brightness levels, which can make it difficult to match features.
- Noise: Noise in the images can obscure features and make it difficult to find corresponding points.
- Lens Distortion: Lens distortion can cause straight lines to appear curved, which can make it difficult to align images.
- Limited Overlap: Images that have a small amount of overlap can make it difficult to find enough corresponding points to stitch them together.
- Moving Objects: Moving objects in the scene can cause artifacts in the stitched panorama.

Despite these challenges, there have been significant advances in panorama recognition in recent years. New algorithms have been developed that are more robust to variations in lighting, exposure, and image quality. Additionally, the availability of more powerful computing resources has made it possible to stitch together larger and more complex panoramas.

As a result, panorama recognition is now a widely used technology in a variety of applications, including photography, virtual reality, and autonomous driving.

16 — **What are the problems in producing the final stitched mosaic image? List the techniques used to address the image stitching problem.** — 4

Image stitching, also known as image mosaicking or panorama stitching, is the process of combining multiple overlapping images into a single panoramic image. The goal of image stitching is to create a seamless and visually pleasing panorama that accurately represents the scene.

Challenges of Image Stitching

There are several challenges associated with image stitching. These challenges include:

- Image alignment: The overlapping images need to be accurately aligned to avoid artifacts such as seams and mismatches.
- Exposure compensation: The overlapping images may have different exposure levels, which can cause the panorama to have uneven brightness.
- Geometric distortions: The overlapping images may have geometric distortions, such as lens distortion or barrel distortion, which can cause the panorama to look distorted.
- Occlusions: Objects in the scene may be occluded in some of the images, which can make it difficult to stitch the images together seamlessly.
- Moving objects: Moving objects in the scene can cause artifacts in the stitched panorama.

Image Stitching Techniques

There are a number of different image stitching techniques available. Some of the most common techniques include:

- Feature-based stitching: This technique identifies and matches features, such as corners or edges, in the overlapping images. The matched features are then used to align the images.
- Direct stitching: This technique directly aligns the overlapping images without explicitly identifying and matching features.
- Graph-based stitching: This technique represents the overlapping images as a graph, where the nodes represent the images and the edges represent the relationships between the images. The graph is then used to find the optimal alignment of the images.

Techniques to Address Image Stitching Problems

- Image registration: This technique aligns overlapping images so that they share a common coordinate system. This is essential for image stitching, as it allows the images to be combined seamlessly.
- Exposure compensation: This technique adjusts the brightness of images so that they have a consistent exposure level. This is important for image stitching, as it prevents the panorama from having uneven brightness.
- Geometric distortion correction: This technique corrects geometric distortions in images, such as

	lens distortion or barrel distortion. This is important for image stitching, as it prevents the panorama from looking distorted. • Inpainting: This technique fills in occluded regions in images. This is important for image stitching, as it prevents the panorama from having holes. • Object motion compensation: This technique compensates for the motion of objects in the scene. This is important for image stitching, as it prevents artifacts from appearing in the panorama when objects move.	

PART B (2x12= 24)

ANSWER ALL THE QUESTIONS

Q.No	Questions	Marks
17.a.	**a) Briefly explain the following methods to establish a transitional alignment between two images or image patches:** **i) Spatially varying weights** **ii) Bias and gain (exposure differences)** **iii) Correlation** The following methods to establish a transitional alignment between two images or image patches: **i) Spatially varying weights** Spatially varying weights are used to adjust the importance of different pixels in the image alignment process. This is useful for situations where some parts of the image are more reliable than others, such as when there is motion blur or occlusion. For example, if there is motion blur in one part of the image, the pixels in that area can be given a lower weight, so that they do not have as much influence on the alignment. Similarly, if there is occlusion in one part of the image, the pixels in that area can be given a weight of zero, so that they are not considered at all in the alignment process. **ii) Bias and gain (exposure differences)** Bias and gain are used to adjust for differences in exposure between the two images. This is useful for situations where the two images were taken with different camera settings, or where there is uneven lighting in the scene. Bias is used to adjust the overall brightness of the image, while gain is used to adjust the contrast of the image. By adjusting the bias and gain, the two images can be made to have the same overall brightness and contrast, which can improve the accuracy of the alignment process. **iii) Correlation** Correlation is a measure of the similarity between two images. It is used to find the best alignment between the two images by maximizing the correlation between them. There are different types of correlation, such as cross-correlation and normalized cross-correlation. Cross-correlation is a simple measure of similarity, while normalized cross-correlation is more robust to changes in brightness and contrast. These are just a few of the many methods that can be used to establish a transitional alignment between two images or image patches. The best method for a particular situation will depend on the specific characteristics of the images.	12
	OR	
17.b.	**b) Examine the various Bundle Adjustment techniques involved in accurately recovering structure and motion.** Bundle adjustment is a non-linear optimization technique used in computer vision to refine the estimates of camera poses and 3D structure from a set of corresponding image points. It is a crucial step in many structure-from-motion (SfM) pipelines, as it can significantly improve the accuracy of the reconstructed 3D model. **Motivation for Bundle Adjustment** In SfM, we aim to reconstruct the 3D structure of a scene from a set of images taken from different viewpoints. This is achieved by matching features across the images and triangulating them to obtain 3D points. However, the initial estimates of camera poses and 3D structure obtained through triangulation are often inaccurate due to various factors, such as noise in the image measurements, lens distortion, and inaccuracies in the camera calibration parameters. Bundle adjustment aims to refine these initial estimates by minimizing a reprojection error. The reprojection error is the difference between the observed image locations of the 3D points and the predicted image	12

locations based on the current estimates of camera poses and 3D structure.

Key Components of Bundle Adjustment

Bundle adjustment involves optimizing a set of parameters that represent the camera poses and 3D structure. These parameters include:

- Camera translation vectors
- Camera rotation matrices
- 3D coordinates of the feature points

The optimization is performed by iteratively updating the parameters until the reprojection error is minimized.

Common Bundle Adjustment Techniques

There are several different bundle adjustment techniques available. Some of the most common techniques include:

- Levenberg-Marquardt algorithm: This is a widely used optimization algorithm that is well-suited for problems with non-linear constraints, such as those encountered in bundle adjustment.
- Sparse bundle adjustment: This technique is used when the number of 3D points is small compared to the number of images. It is computationally less expensive than dense bundle adjustment.
- Dense bundle adjustment: This technique is used when the number of 3D points is large. It is more computationally expensive than sparse bundle adjustment, but it can produce more accurate results.

Benefits of Bundle Adjustment

Bundle adjustment offers several benefits for SfM, including:

- Improved accuracy: Bundle adjustment can significantly improve the accuracy of the reconstructed 3D model.
- Reduced noise: Bundle adjustment can reduce the effects of noise in the image measurements.
- Correction of lens distortion: Bundle adjustment can correct for lens distortion, which can improve the accuracy of the reconstructed 3D model.
- Incorporation of camera calibration parameters: Bundle adjustment can incorporate camera calibration parameters, which can further improve the accuracy of the reconstructed 3D model.

Applications of Bundle Adjustment

Bundle adjustment is used in a wide variety of applications, including:

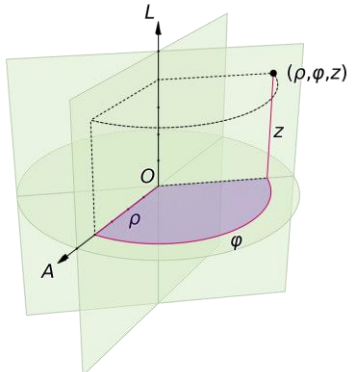
- Photogrammetry: Bundle adjustment is used in photogrammetry to reconstruct 3D models from a set of overlapping images.
- Robotics: Bundle adjustment is used in robotics to estimate the pose of a robot from a set of images.
- Virtual reality: Bundle adjustment is used in virtual reality to create 3D models of real-world environments.
- Autonomous driving: Bundle adjustment is used in autonomous driving to reconstruct 3D models of the surrounding environment.

Bundle adjustment is a powerful tool for improving the accuracy of 3D reconstructions obtained from SfM. It is a widely used technique in a variety of applications, and it is likely to continue to be an important part of SfM pipelines in the future.

18.a.

a) Draw neat figures for "Projection from 3D to cylindrical and spherical coordinates" and explain the same. 12

Cylindrical Coordinates



Cylindrical coordinates are a three-dimensional coordinate system that uses three values to represent a point in space:

- Radial distance: The distance from the origin to the point.
- Polar angle: The angle between the positive x-axis and the projection of the line connecting the origin to the point onto the xy-plane.
- Height: The z-coordinate of the point.

To project a point from 3D to cylindrical coordinates, we first need to calculate the radial distance. This is simply the distance from the origin to the point:

r = sqrt(x^2 + y^2)

Next, we need to calculate the polar angle. This is the angle between the positive x-axis and the projection of the line connecting the origin to the point onto the xy-plane. To calculate this angle, we use the following formula:

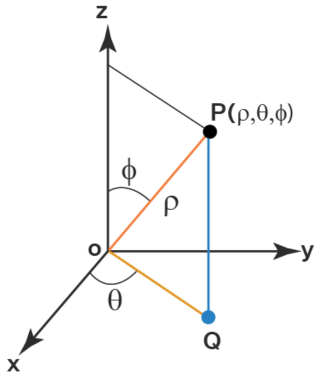
theta = arctan(y / x)

Finally, we need to calculate the height. This is simply the z-coordinate of the point:

z = z

Once we have calculated these three values, we can represent the point in cylindrical coordinates as (r, theta, z).

Spherical Coordinates



Spherical coordinates are a three-dimensional coordinate system that uses three values to represent a point in space:

- Radial distance: The distance from the origin to the point.
- Polar angle: The angle between the positive z-axis and the line connecting the origin to the point.
- Azimuthal angle: The angle between the positive x-axis and the projection of the line connecting the origin to the point onto the xy-plane.

To project a point from 3D to spherical coordinates, we first need to calculate the radial distance. This is simply the distance from the origin to the point:

r = sqrt(x^2 + y^2 + z^2)

Next, we need to calculate the polar angle. This is the angle between the positive z-axis and the line connecting the origin to the point. To calculate this angle, we use the following formula:

theta = arctan(sqrt(x^2 + y^2) / z)

Finally, we need to calculate the azimuthal angle. This is the angle between the positive x-axis and the projection of the line connecting the origin to the point onto the xy-plane. To calculate this angle, we use the following formula:

phi = arctan(y / x)

Once we have calculated these three values, we can represent the point in spherical coordinates as (r, theta, phi).

OR

18.b.

b) How do panoramas are recognized? Explain various methods and algorithms used for recognizing panoramas.

12

Panorama recognition is the task of identifying images that represent a panoramic view of a scene. This is a challenging task due to the large variations in appearance and content that can occur in panoramas.

Here are some of the methods and algorithms that are used for recognizing panoramas:

Feature-based methods:

These methods extract features from the images, such as SIFT features or SURF features, and then match the features between the images. The matched features are then used to determine whether the images overlap and whether they represent a panoramic view of a scene.

Direct methods:

These methods do not explicitly extract and match features from the images. Instead, they directly compare the pixel intensities of the images to determine whether they overlap and whether they represent a panoramic view of a scene.

Graph-based methods:

These methods represent the images as a graph, where the nodes represent the images and the edges represent the relationships between the images. The graph is then used to identify connected components, which can represent panoramas.

Deep learning-based methods:

These methods use deep learning techniques to learn features from the images and to classify the images as either panoramas or not.

Examples of panorama recognition algorithms:

- AutoStitch: This algorithm uses a combination of feature-based and direct methods to recognize panoramas.
- PanoTools: This algorithm uses a combination of feature-based and graph-based methods to recognize panoramas.
- Panorama Stitcher: This algorithm uses a combination of direct and deep learning-based methods to recognize panoramas.
- DeepPano: This algorithm uses a deep learning-based method to recognize panoramas.

Performance of panorama recognition algorithms:

The performance of panorama recognition algorithms depends on a number of factors, including the quality of the images, the amount of overlap between the images, and the complexity of the scene. In general, deep learning-based methods have been shown to outperform traditional methods on this task.

Applications of panorama recognition:

Panorama recognition has a number of applications, including:

- Photo editing software: Panorama recognition can be used to automatically stitch together panoramic images.
- Virtual reality: Panorama recognition can be used to create virtual reality experiences.
- Robotics: Panorama recognition can be used to help robots navigate their environment.

Future directions of panorama recognition:

Future research in panorama recognition is likely to focus on improving the performance of deep learning-based methods and on developing new methods for recognizing panoramas in challenging conditions, such as when the images are low-resolution or when there is a lot of noise in the images.

HOD/CSE