Question 1. What feature detection and tracking method is used in the provided pipeline? Can we use any other method? if yes, what? Please describe in only a couple of sentences.

For feature detection, we are using the FAST (Features from Accelerated Segment Test) algorithm implemented in OpenCV. The detected features are tracked using the Kanade-Lucas-Tomasi feature tracker.
Yes, we can use other methods too. For example, for feature detection, we can use Shi-Tomasi Corner Detector to select corners and use either of dense or sparse Lucas-Kanade method to track those corners.

Question 2. What steps do we take to calculate the essential matrix? Can you find the intrinsic parameters of the camera using essential matrix? If yes, please write steps on how to achieve it? Please refer to this document to learn more about essential matrix: https://en.wikipedia.org/wiki/Essential_matrix

We detect features and track them on both images. Using those points, we use the RANSAC algorithm to find the essential matrix. An Essential matrix gives the relation between two poses. Further, we pass it onto the recover pose method to get rotation and translation.
To find the camera parameters, we can use the following equation where E is the essential matrix, F is the fundamental matrix and K and K' are inner camera parameters.

$$\mathbf{E} = (\mathbf{K}')^{\top}\ \mathbf{F}\ \mathbf{K}.$$

Question 3. What was the mean ATE (Average Trajectory Error) between the estimated trajectory and provided ground truth?

Average Trajectory Error is the root mean squared error that we calculate for all the estimated poses. For a single estimated pose, the error is the euclidean distance between the estimated pose and its corresponding ground truth pose. We sum all these squared errors and divide them by the number of estimated poses and take the square root of it to get the RMSE which gives us the average error in the trajectory.

Question 4. How can you optimize the ATE in this case? What possible issues can you find in the given pipeline?

To optimize the ATE, we need to include more features and lower the frame rate. That will help in better estimation of poses.
Another issue with this pipeline is that we are not using the scale information while estimating the poses which leads to high ATE.