Computer Vision Coursework

Yaqi Zha

November 14, 2024

1. To detect salient features in the provided video frames, I want to use the **Scale-Invariant Feature Transform (SIFT)**. SIFT is effective for detecting distinctive keypoints in images that are invariant to scale, rotation, and changes in illumination, making it ideal for video sequences with varying perspectives.

   SIFT identifies keypoints by detecting local extrema in the Difference of Gaussian (DoG) scale space, allowing it to capture high-contrast features like edges and corners, particularly useful for architectural details such as window frames, building edges, and lawn boundaries present in the frames. Each keypoint includes a descriptor vector that encodes local gradient information, providing stable, distinguishable features across frames. This consistency across transformations ensures reliable tracking and matching, which is essential for tasks like 3D reconstruction throughout the video sequence.

2. To match the detected salient features between the video frames using **Nearest Neighbor Distance Ratio (NNDR)**, we can follow these steps:

   1. For each detected keypoint in both frames, calculate the SIFT descriptor, which provides a unique vector based on local gradient information at each keypoint.

   2. For each descriptor in Frame 1, using k-nearest neighbors ($k = 2$) to find the two closest matches (nearest and second-nearest neighbors) in Frame 2 based on Euclidean distance.

   3. Apply the distance ratio test by comparing the distances of the two closest matches. If the nearest match distance is significantly smaller (e.g., less than 0.7 times) than the second-nearest, accept the match; otherwise, discard it. This reduces incorrect matches by ensuring only distinct, well-matching points are retained.

   4. To refine matches, apply RANSAC to eliminate outliers by fitting a geometric model (e.g., homography) to the matches, retaining only consistent correspondences.

3. (a) The detected salient features in both frames are shown below, highlighted by circular markers on distinctive points such as edges and corners.

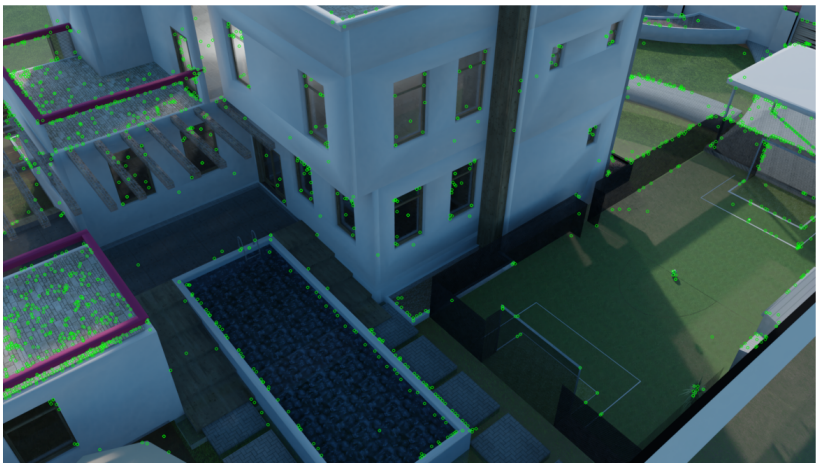

Figure 1: Detected features in Frame 1

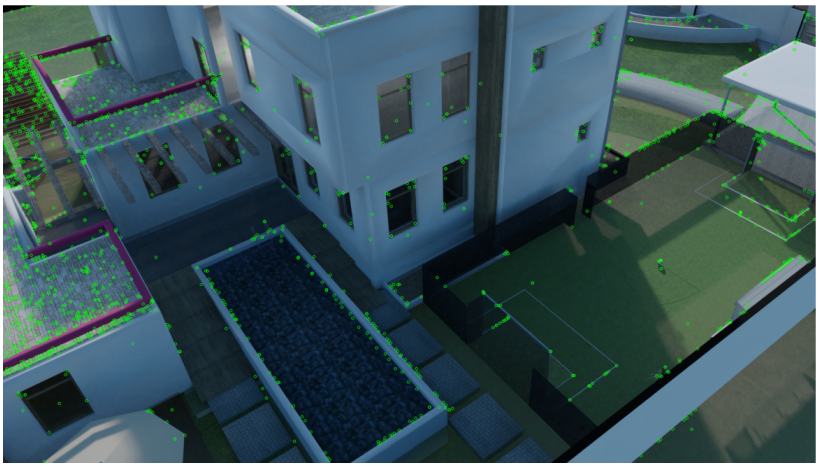

Figure 2: Detected features in Frame 2

(b) After applying NNDR matching, corresponding features between the two frames were identified and visualized as shown below. Correct matches are depicted as lines connecting keypoints in each frame.

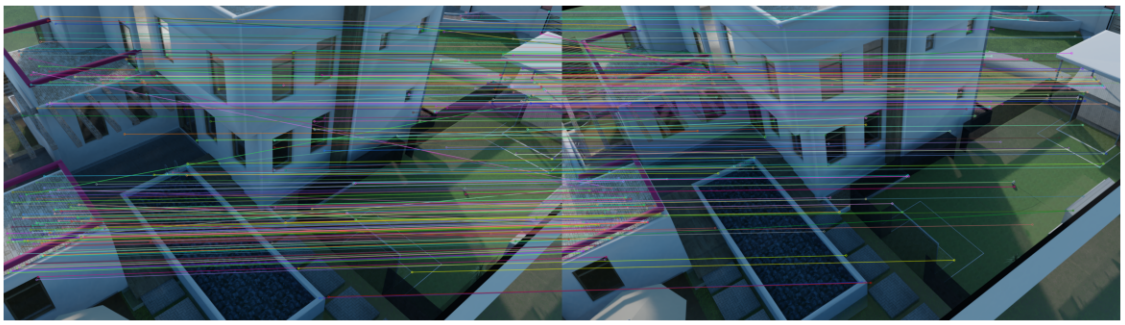

Figure 3: Matching Keypoints Between Frame 1 and Frame 2

(c) To estimate the fundamental matrix F from matched features, we applied the eight-point algorithm with RANSAC, which reduces outlier impact and provides an approximation based on feature correspondences. Some residual noise, however, may still influence the resulting matrix.

$$F_{\text{features}} = \begin{bmatrix} -5.488 \times 10^{-9} & -5.070 \times 10^{-8} & 1.233 \times 10^{-4} \\ 1.552 \times 10^{-6} & -5.753 \times 10^{-8} & -2.715 \times 10^{-2} \\ -7.938 \times 10^{-4} & 2.406 \times 10^{-2} & 1.000 \end{bmatrix}$$

For comparison, F was also computed using known intrinsic and extrinsic camera parameters. With intrinsic matrices K and extrinsic parameters (rotation R and translation T), the fundamental matrix F is calculated as $F = K'^{-T} E K^{-1}$, where $E = [T]_x R$ is the essential matrix.

$$F_{\text{camera}} = \begin{bmatrix} -4.012 \times 10^{-10} & -8.780 \times 10^{-8} & 4.977 \times 10^{-5} \\ 2.512 \times 10^{-8} & 1.366 \times 10^{-9} & 1.314 \times 10^{-3} \\ -1.883 \times 10^{-5} & -1.158 \times 10^{-3} & -5.249 \times 10^{-2} \end{bmatrix}$$

The **parameter-based method** is generally more accurate because it uses known camera calibration data, minimizing errors from noisy matches. In contrast, the feature-based method is sensitive to matching errors and noise. To improve accuracy, we can use RANSAC to reject outliers, and refine matches with distinctive descriptors or a more robust estimator with stricter thresholds for outlier rejection, enhancing the reliability of the feature-based approach.

(d) To find correctly matched points that meet the epipolar constraint, we filter matches by checking if each pair satisfies the condition

$x'^T F x = 0$, where F is the fundamental matrix, and x and x' are corresponding points in the two frames. If the points meeting this constraint, they have small epipolar line distances, meaning their matched point lies close to the epipolar line.

In practice, we check if each point in Frame2 lies close to the corresponding epipolar line, by calculating its perpendicular distance to the epipolar line. Matches with distances below a small threshold (e.g., 1 pixel) are considered inliers, as they closely follow the camera's geometric relationship.

After identifying these inlier matches, we can visualize them by drawing lines connecting each matching pair across the two images, showing only those that closely follow the epipolar geometry. This filtering highlights points that align with the camera's spatial relationship and reduces mismatches.

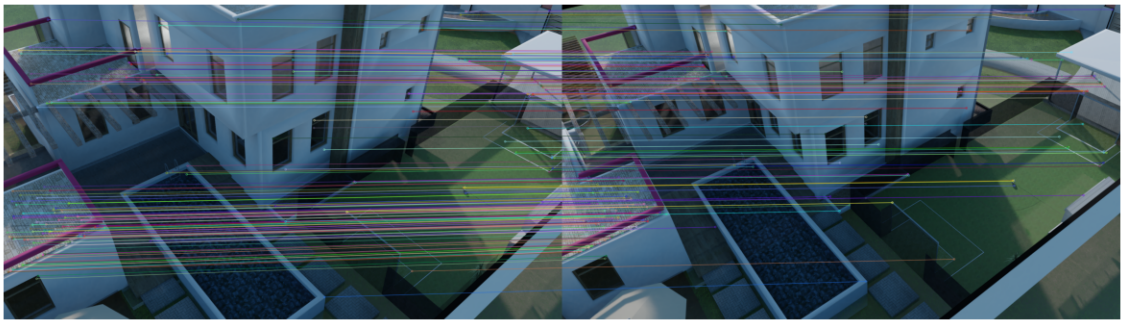

Figure 4: Correct matches meeting the epipolar constraint

(e) To estimate the pool area and touchline length, we rectified the frames with camera parameters and computed a disparity map using Stereo SGBM. Depth values were derived from disparity using $Z = \frac{f \cdot b}{\text{disparity}}$, where f is the focal length and b is the baseline. Key 2D points on the pool and touchline were converted to 3D coordinates, allowing us to calculate the pool's area and the touchline's length.

The estimated pool area is approximately 25.32 square meters, and the touchline length is approximately 15.40 meters.

4. We rectified both frames using camera parameters to align corresponding points horizontally, then draw horizontal lines to check rectification.

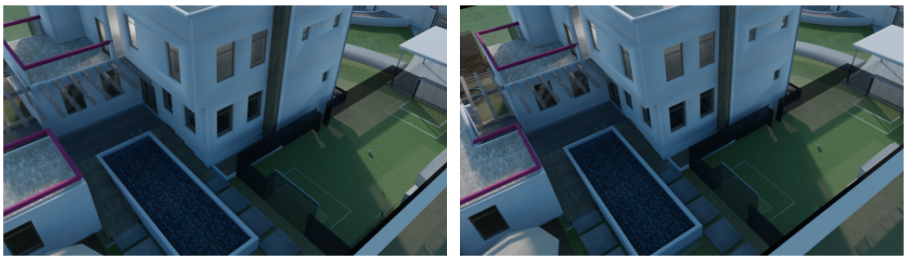

Figure 5: Original Frame1 and Frame2

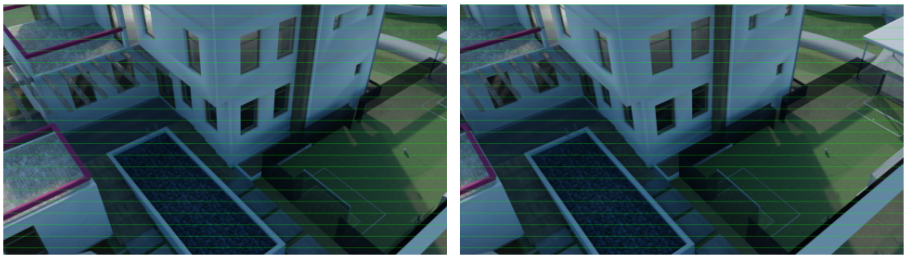

Figure 6: Rectified frames with horizontal lines

Next, we computed a disparity map using Stereo SGBM, where each pixel's disparity value represents depth information for 3D reconstruction.

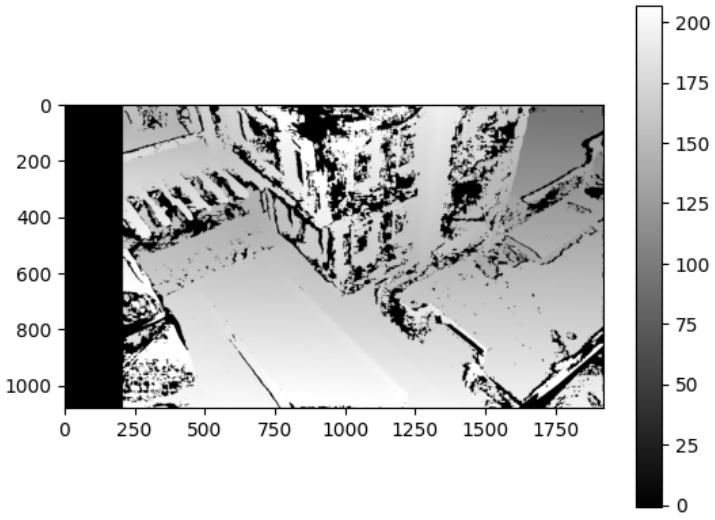

Figure 7: Disparity map