University of Maryland College Park

Professional Master of Engineering

Robotics Engineering

ENPM673 - Perception for Autonomous Robots - Project 4

Students:
Akash Agarwal (116904636)
Nupur Nimbekar (116894876)
Anubhava Paras (116905909)

Lecturer:
Dr. Mohammed Samer Charifa

April 20, 2020

Link to drive to access data and output

Lucas Kanade Template Tracker

The Lucas Kanade method is used for optical flow estimation, to track desired features in a video. The following equation is used to calculate the optical vectors and is solved for every pixel using the least squares method.

$$\begin{bmatrix} \Sigma I_x^2 & \Sigma I_x I_y \\ \Sigma I_x I_y & \Sigma I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \Sigma I_x I_t \\ \Sigma I_x I_t \end{bmatrix}$$

The objective of the Lucas Kanade Template Tracker is to align a template image to an input image, where the template is an extracted sub-region of a particular window size of image at point in time and the input image is the image after one time step.

However, there are a few shortcomings to the LK tracker:-

1. The tracker is sensitive to intensity or illumination and requires constant lighting to be efficient.
2. The distance travelled by the pixels between two time steps should be very small or close to constant as a large distance renders the tracker ineffective.

Implementation of the Tracker

Step 1:- The first step in implementing the tracker was to define the ROI or bounding box around the object to be tracked. OpenCV's built-in functionality was used to draw the box as well as record the coordinates for the same.

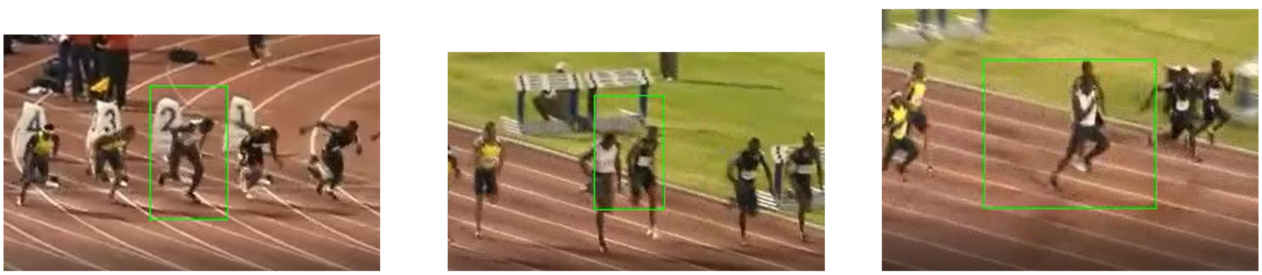
Step 2:- The next step was creating the *affineLKtracker* function, which took as input a grayscale image of the current frame, the template image, the bounding box that marks the template region and the parameters of the previous warping. The function returned the new warped parameters. To get these parameters, a series of steps were involved which are iterated till the change in parameters is negligible.

- Warping the image
- Computing the difference between the template and the warped image
- Warping the gradient after calculating the image gradient
- Calculating the Jacobian
- Calculating the steepest descent
- Computing the Hessian matrix
- Computing the steepest descent gradient
- Obtaining the change in parameters and then updating it

Evaluating the tracker

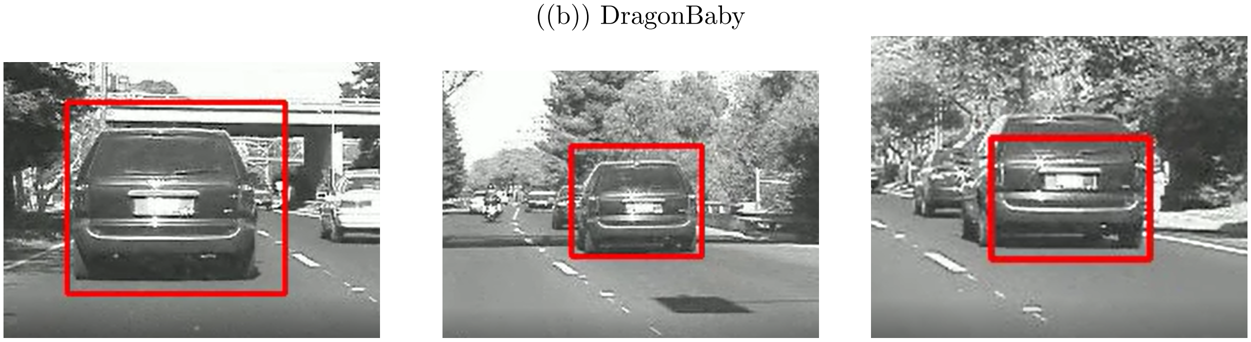
The tracker breaks down when the distance covered by the pixels in consecutive frames is too large, which is the second shortcoming of the LK tracker. To overcome this we use the pyramidal representation(coarse to fine registration), so that tracking of faster movements is possible.
In its crux, what the pyramidal representation does is, it lowers the resolution of the input image by a number of levels, so that faster movements/larger distances appear slower/shorter and can then be tracked as usual. The Lucas Kanade tracker is then applied on it as an iterative approach, till we get back to the original input image.

Another technique to make the tracker more efficient is re-evaluating the desired features at regular intervals as over time or as the video progresses, it is possible that some features have been lost and thus making it difficult to track the object.



((a)) Bolt

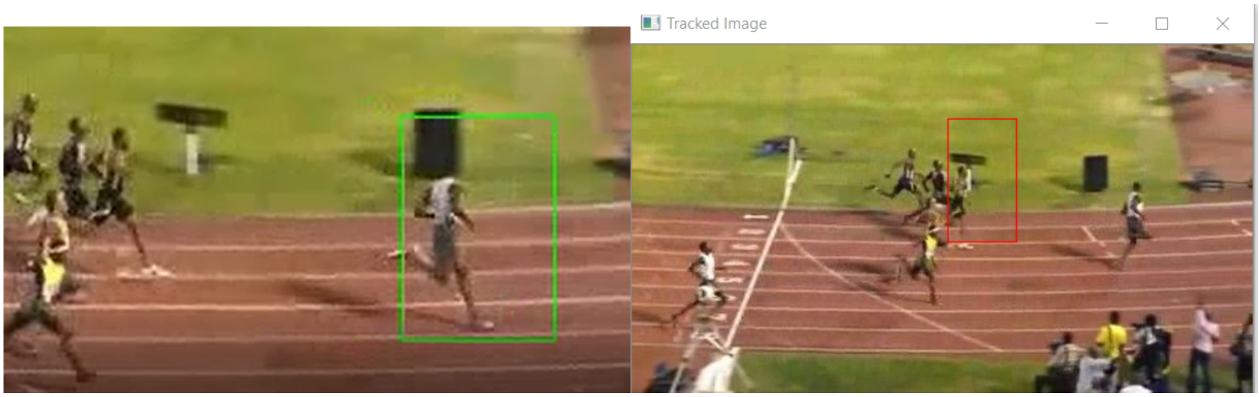
((b)) DragonBaby



((c)) Car

Figure 1: Object Tracking for the various datasets at different time stamps in the video

Tracking of features without use of pyramidal representation gives large tracking error, as seen in the images below.



((a)) Bolt

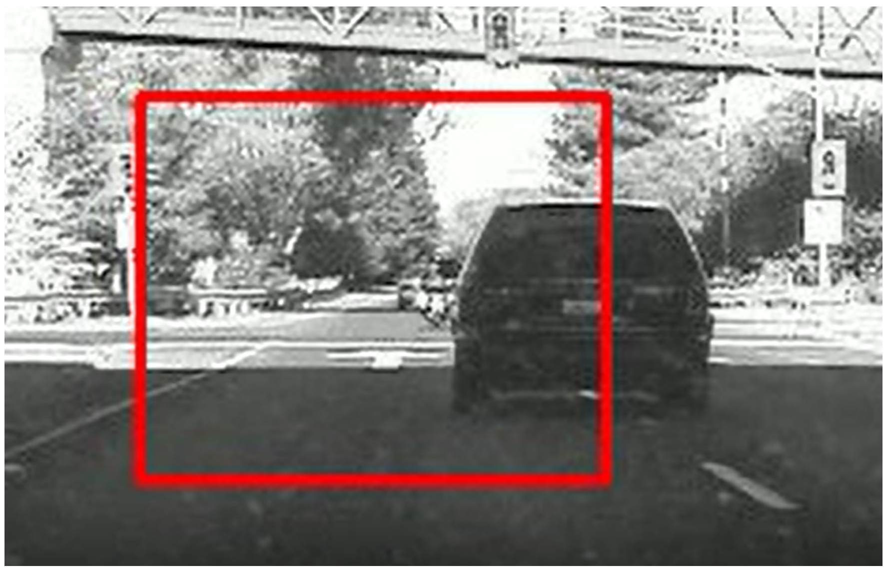
((b)) Car

Figure 2: Comparing object tracking with(left) and without(right) pyramidal representation (i.e. number of layers of pyramid=0)

Robustness to illumination

1. To scale the brightness of the pixels for every frame, we apply the CLAHE(Contrast Limited Adaptive Histogram Equalization) method to each frame as a form of pre-processing before implementing the L-K method.

    - CLAHE - The image is divided into small blocks called ”tiles”. Then, each of these blocks are histogram equalized. So in a small area, histogram would confine to a small region. To avoid this, contrast limiting is applied to avoid amplification of any noise present. If any histogram bin is above the specified contrast limit, those pixels are clipped and distributed uniformly to other bins before applying histogram equalization.



((a)) Before applying CLAHE

((b)) After applying CLAHE

Figure 3: Effect of illumination on object tracking