ENPM673 Project 4 Lucas-Kanade Template Tracker

Kumar Sambhav Mahipal, Raghav Agarwal and Vasista Ayyagari

20^{th} April 2020

Link to Results:
https://drive.google.com/open?id=1pyejaWCyIdXDF5WEYHmkM0RkfbLblu7e

1 Introduction

Optical flow of events inside a frame is tracked by Lucas Kanade Algorithm. These events in a frame are gathered after considering a pixel and its adjacent pixels. The least square criterion is used to approximate the solution. For optical flow, same pixel intensity between two frames is used to find the flow of motion or the vector flow. Following this, the optical flow equations are used in the pixels and the window center. In this project we implemented the Lucas Kanade tracker on given video sequence i.e. bolt, car and dragon baby. The tracker was initialized by defining a template, i.e. by bounding a rectangle around the object/ROI needed to be tracked. For subsequent frames, the tracker updates the parameters of affine transformation and warps it to the current frame.

2 Lucas-Kanade Algorithm

The goal of Lucas Kanade is to align a template image $T(\mathbf{x})$ to an input image $I(\mathbf{x})$, where $\mathbf{x} = (x, y)^{\mathrm{T}}$ is a column vector containing the pixel coordinates.If the Lucas-Kanade algorithm is being used to compute optical flow or to track an image patch from time $t = 1$ to time $t = 2$, the template $T(x)$ is an extracted sub-region (a 5×5 window, maybe) of the image at $t = 1$ and $I(x)$ is the image at $t = 2$.

2.1 Least Square:Affine Parameters

The Warp $W(x; p)$ takes the values of the pixels x in the coordinate frame of the template T and maps it to the sub-pixel location $W(x; p)$ in the coordinate frame of image $I.W(x; p)$ denotes the set of translations where $p = (p_1, \ldots\ldots, p_n)^T$ is the vector of parameters.

$$W(x;p) = \begin{pmatrix} x + p_1 \\ y + p_2 \end{pmatrix}$$

The optical flow is now the vector of parameters $p = (p_1, p_2)^T$. To calculate Affine warps:

$$W(x;p) = \begin{pmatrix} (1+p_1)\cdot x + p_3 \cdot y + p_5 \\ p_2 \cdot x + (1+p_4)\cdot y + p_6 \end{pmatrix} = \begin{pmatrix} 1+p_1 & p_3 & p_5 \\ p_2 & 1+p_4 & p_6 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

2.2 Goal of Lucas-Kanade Algorithm

The goal of the Lucas-Kanade algorithm is to minimize the sum of squared error between two images,the template and the image warped back onto the coordinate frame of the template:

$$\sum_{\mathbf{x}} [I(\mathbf{W}(\mathbf{x};\mathbf{p})) - T(\mathbf{x})]^2$$

2.3 Steps involved in Lucas-Kanade Algorithm

Iterate:
(1) Warp I with $\mathbf{W}(\mathbf{x};\mathbf{p})$ to compute $I(\mathbf{W}(\mathbf{x};\mathbf{p}))$
(2) Compute the error image $T(\mathbf{x}) - I(\mathbf{W}(\mathbf{x};\mathbf{p}))$
(3) Warp the gradient ∇I with $\mathbf{W}(\mathbf{x};\mathbf{p})$
(4) Evaluate the Jacobian $\frac{\partial W}{\partial p}$ at $(x;p)$
(5) Compute the steepest descent images $\nabla I \frac{\partial W}{\partial p}$
(6) Compute the Hessian Matrix
(7) Compute $\sum_{\mathbf{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}}\right]^{\mathrm{T}} [T(\mathbf{x}) - I(\mathbf{W}(\mathbf{x};\mathbf{p}))]$
(8) Compute $\Delta\mathbf{p}$
(9) Update the parameters $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$ until $\|\Delta\mathbf{p}\| \leq \epsilon$

3 Methods and Observations used to improve performance

- Histogram Equalization is found to improve the tracking as a measure against brightness variation.
- Interpolating the image to another size improved the performance because it controls the overall flow of the image to a desirable state that can be detected by the tracker

Figure 1: Result from bolt video

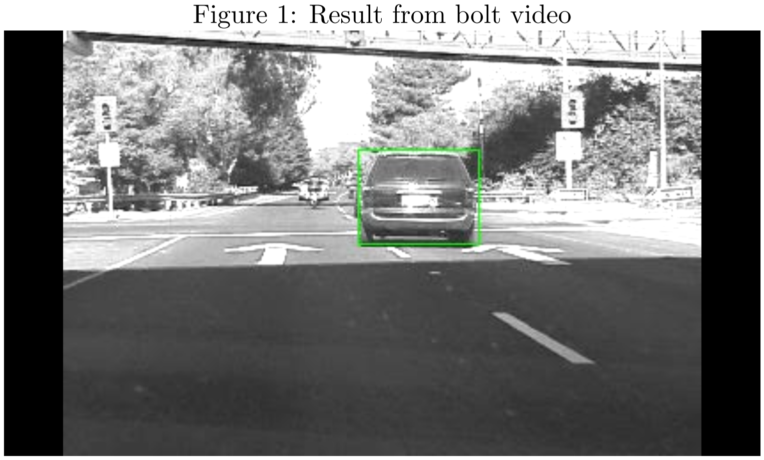

Figure 2: Result from car video

Figure 3: Result from dragon baby video

- The tracker fails almost completely and goes random in highly dynamic scenes (The dragon baby video). Strong gaussian blurring helps especially in this situation, but fails in slow moving scenes (car video).
- Each scene needs a different threshold
- Occasionally refreshing the template to reduce strong warping showed improvement as the tracker fails when the overall difference between template and current image becomes high
- Different scenes had different amount of flow. The car was slow moving in the image, bolt in the medium speed and baby-dragon scene was fast as well as had scenes completely changing. Hence, the threshold and blur parameters are different for all for optimal results

4 Conclusion

Hence, using Lucas Kanade Tracker alone will not be enough. The main disadvantage of Lucas Kanade Tracker is that it works well only for subpixel level flow. Both Bolt and Dragon-Baby had a large flow between pixels. So the tracker often fell into a local minima. The above techniques improved the performance and avoided the local minima