Perception For Autonomous Robots

Project 4

Implementation of Lucas-Kanade(LK) template tracker and its Optimization

Nikhil Mehra (116189941) Pranali Desai (116182935)

Sayan Brahma (116309165)

Contents

List of Figures

1 Introduction

Lucas Kanade is a method to generate the optical flow of events in a frame. These events are gathered after considering a pixel and its neighbouring pixels. The least square criterion, an approach in regression analysis is used to approximate the solution. The displacement of an image and its pixel contents are studied for two frames and at these two instants these are nearly the same. Thus, optical flow equations are used to and assumed to be true for all the pixels in a window center.

2 Inverse Compositional Image Alignment

2.1 Least Square Minimization

'u' and *'v'* are the location of template in current frame:

$$E(u,v) = \sum [I(x+u, y+v) - T(x,y)]^2$$

$$E(u,v) \approx \sum [I(x,y) + uI_x(x,y) + vI_y(x,y) - T(x,y)]^2$$

$$E(u,v) = \sum [uI_x(x,y) + vI_y(x,y) + D(x,y)]^2$$

By taking partial derivatives:

$$\partial E/du = \sum [uI_x(x,y) + vI_y(x,y) + D(x,y)]I_x(x,y) = 0$$

$$\partial E/dv = \sum [uI_x(x,y) + vI_y(x,y) + D(x,y)]I_y(x,y) = 0$$

Least-Squares method is used to solve Form matrix :

$$\sum \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -\sum \begin{bmatrix} I_xD \\ I_yD \end{bmatrix}$$

2.2 Least Square with Affine Parameters

$W(x;p)$ denotes the set of parameters $p = (p_1, ..., p_n)^T$ is the vector of parameters. The Warp $W(x;p)$ takes the pixel 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 the image I.

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

where the vector of parameters $p = (p_1, p_2)^T$ then became the optical flow. We considered the moving 3D frame and calculated the affine warps as follows:

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

2.3 Inverse Compositional Image Alignment

The cost to calculate the Hessian matrix in each step is significantly huge and to reduce this cost, there is an updated process named Inverse Compositional Image Alignment. In this various approximate solution are assumed to be constant and the parameters are updated after a couple of iterations. The Hessian is considered to be relatively constant in the complete image.

The role of template and image is switched or the variables are interchanged. There are two approaches to do this, either the additive approach or the compositional approach.

$$\begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_{25}) & I_y(p_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(p_1) \\ I_t(p_2) \\ \vdots \\ I_t(p_{25}) \end{bmatrix}$$

Minimizing the equations using inverse compositional algorithm:

$$\sum_x \left[T\left(W\left(x;\Delta p\right)\right) - I\left(W\left(x;p\right)\right)\right]^2$$

The updated warp after reversing the I and T is as follows:

$$W\left(x;p\right) \leftarrow W\left(x;p\right) \circ W\left(x;\Delta p\right)^{-1}$$

The parameters for the inverse for the affine warp are:

$$\frac{1}{(1+p_1)\cdot(1+p_4) - p_2 \cdot p_3} \begin{pmatrix} -p_1 - p_1 \cdot p_4 + p_2 \cdot p_3 \\ -p_2 \\ -p_3 \\ -p_4 - p_1 \cdot p_4 + p_2 \cdot p_3 \\ -p_5 - p_4 \cdot p_5 + p_3 \cdot p_6 \\ -p_6 - p_1 \cdot p_6 + p_2 \cdot p_5 \end{pmatrix}$$

By performing the first order Taylor expansion

$$\sum_x \left[T\left(W\left(x;0\right)\right) + \bigtriangledown T \frac{\partial W}{\partial p} \Delta p - I\left(W\left(x;p\right)\right) \right]^2$$

The least squares question is estimated after assuming W is the identity warp:

$$\Delta p = H^{-1} \sum_x \left[\bigtriangledown T \frac{\partial W}{\partial p} \right]^T \left[I(W(x;p)) - T(x) \right]$$

The Hessian matrix where I is replaced with T

$$H = \sum_x \left[\bigtriangledown T \frac{\partial W}{\partial p} \right]^T \left[\bigtriangledown T \frac{\partial W}{\partial p} \right]$$

2.4 The Inverse Compositional Algorithm

Pre-Computation Steps:

1. Evaluate the gradient $\bigtriangledown T$ of the template $T(x)$
2. Evaluate the Jacobian $\frac{\partial W}{\partial p}$ at $(x;0)$
3. Compute the Steepest descent images $\bigtriangledown T \frac{\partial W}{\partial p}$
4. Compute the Hessian matrix

Iterative Steps:

5. Warp I with $W(x;p)$ to compute $I(W(x;p))$
6. Compute the error in the image $I(W(x;p)) - T(x)$
7. Evaluate $\sum_x \left[\bigtriangledown T \frac{\partial W}{\partial p} \right]^T \left[I\left(W\left(x;p\right)\right) - T\left(x\right) \right]$
8. Compute Δp
9. Update the warp $W\left(x;p\right) \leftarrow W\left(x;p\right) \circ W\left(x;\Delta p\right)^{-1}$

until $\|\Delta p\| \leq \epsilon$

3 Coping with the loss of Features

Lucas Kanade method involves the input of a bounding box whose features(in the form of pixels) are extracted and then compared with the consecutive upcoming frames. This process is carried on for the continuous frames of the complete video, but as the frames progress, the features are lost in the transition from one frame to other. This issue is corrected with the help of reiterating the features every few frames so the lost features are recovered and the bounding box is perfect along the object to be tracked.

4 Histogram Equalization

1. Histogram Equalization has been applied for tracking the car under the bridge.
2. This equalizes the darkness thus able to detect the car.
3. Gamma Correction was also tried to equalize the darkness but the output was not that promising.
4. We see that in 1-2 frames the rectangle goes a bit out of car, but the delta-p again converges thus giving us less error.

5 Pyramid for rapid motion

1. This is used when there is rapid movement in the video frames/ rapid motion (input video of vase in our case).
2. We have used pyramid by decreasing the frame resolution by 4 times, i.e in total three layers are used, wherein first layer is our normal layer, second layer and third layer have resolution reduced by 2 and 4 as compared to the first layer.
3. The parameters are calculated from bottom to top layer. Each time the bottom layer is warped to the layer above it with updated parameters.
4. Hence in the end the parameters are upgraded to the first layer by considering the rapid shifts in the lower resolution layers.

6 Pipeline

1. Initially a template was defined by drawing a rectangle around the object that needed to be tracked
2. The subsequent frames were checked using Lucas Kanade algorithm, which returns the change in parameters between two frames (Δp).
3. Parameters are updated in the current frame ($p + \Delta p$)
    - For Human and the described pipeline worked accurately

- For Car, to counteract the sudden change in intensity, Histogram equalization is used and also, features are extracted at every 100^{th} frame again that are to be tracked in following frames.
- For Vase, as there are large and rapid motions between two frames, Pyramid Algorithm is introduced into the pipeline.

4. After the convergence of the error of the parameters below some threshold(usually 0.001) the parameters are returned.

5. Updated parameters is used to draw the rectangle and warped back to the current frame.

7 Discussions

1. Lucas-Kanade Inverse Compositional Algorithm variation is used as it gave more robust results from normal Lucas-Kanade Affine Tracker Algorithm.

2. The algorithm breaks down when there is illumination change, this was avoided by using Histogram Equalization as talked above.

3. Threshold for ΔP of about 0.001, 0.001 and 0.0001 is used in case of human, car and vase input respectively.

4. To make the tracking more robust, we have used the combination of different template, i.e we change the template to the previous one at every 100th frame, for the car video.

5. In order to compensate for large motions we have added Pyramid Algorithm to our pipeline. Three layer Pyramid Algorithm worked well for us.

7.1 Human

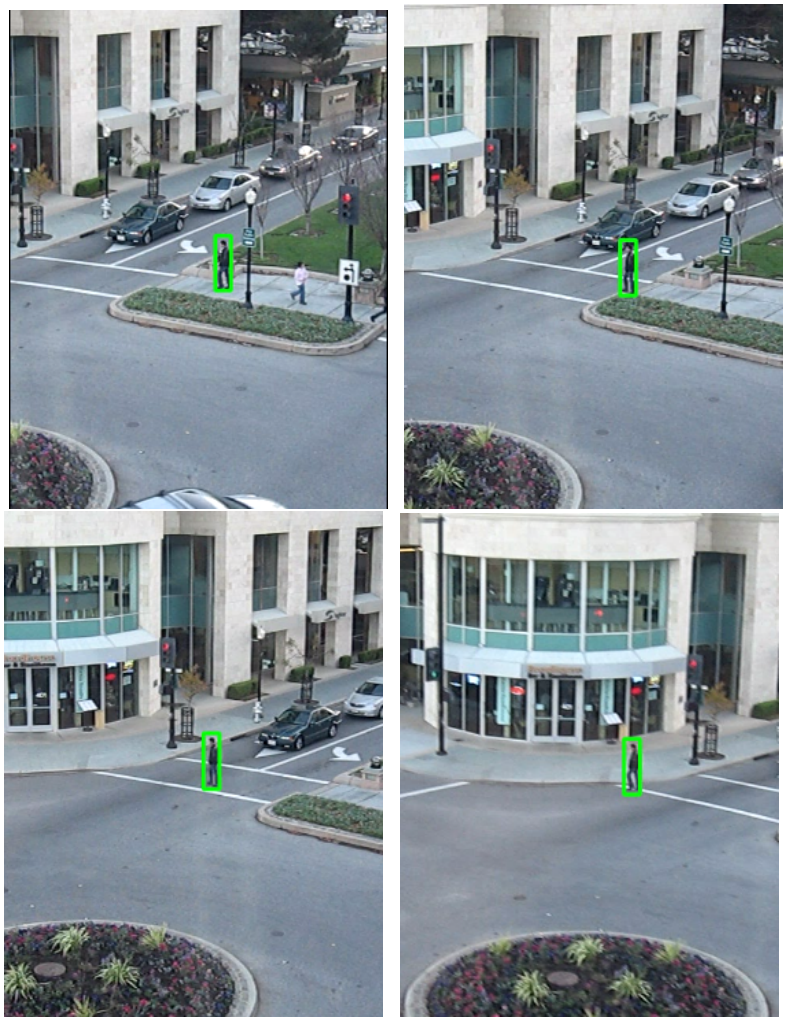

Figure 1: Human tracking results

7.2 Car

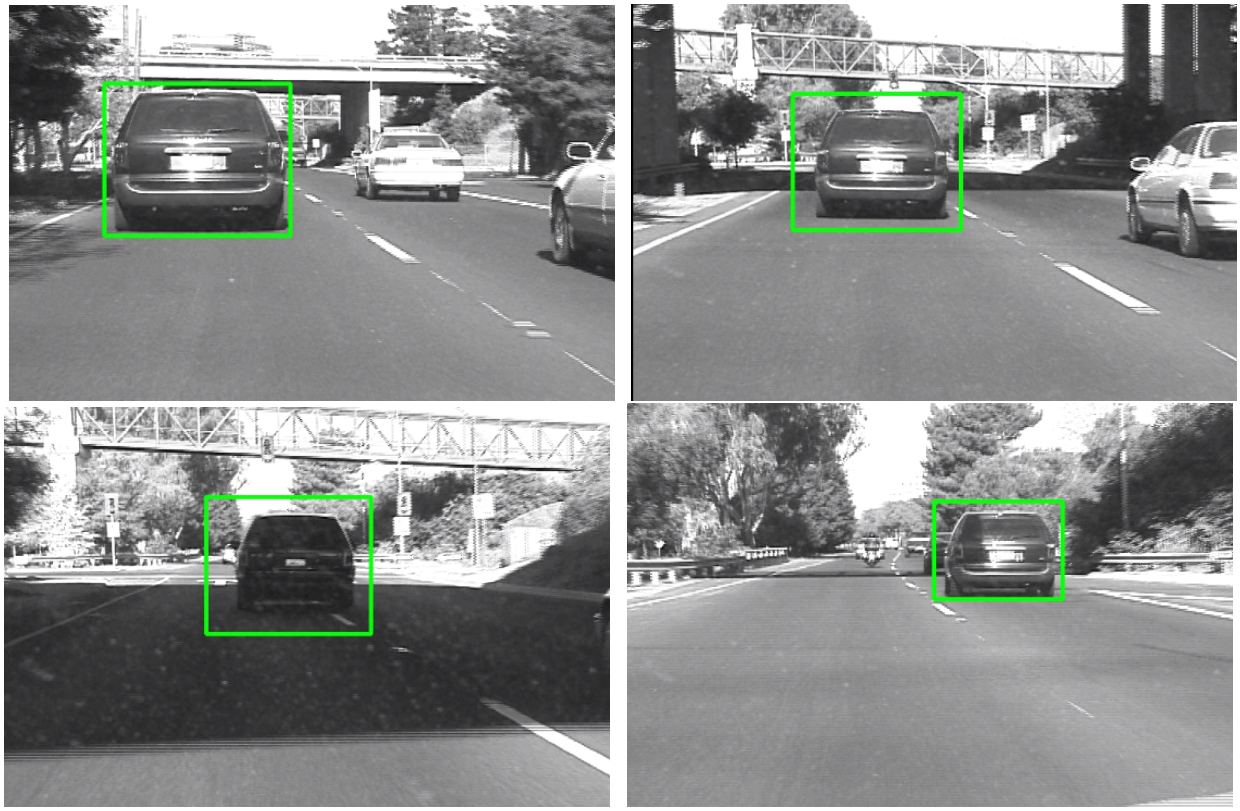

Figure 2: Car tracking results

7.3 Vase

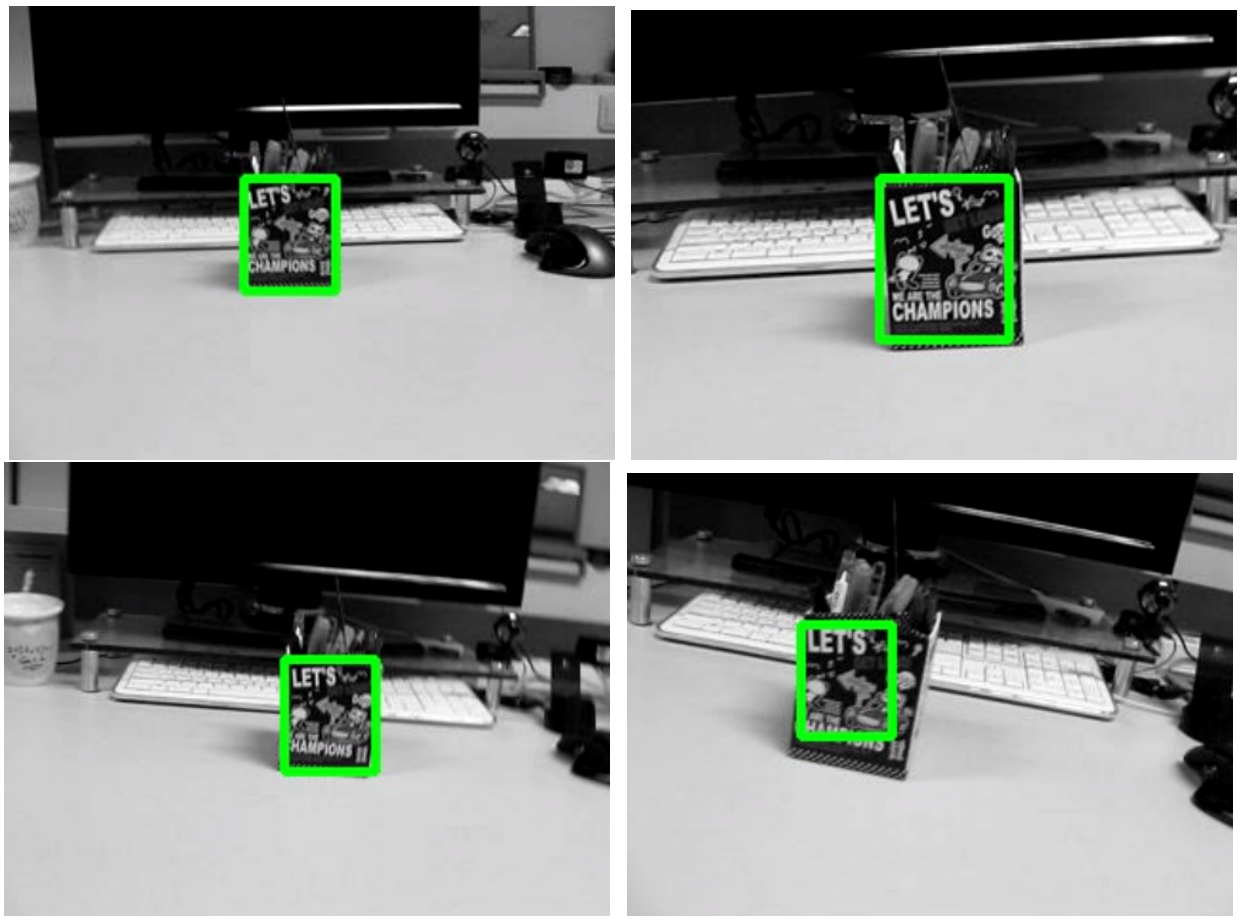

Figure 3: Vase tracking results

8 Link

The link for the video is -

https://drive.google.com/open?id=1ldZf$_3$1$jp9ZujMkSzQRw2RAwxWoljjc3$

9 References

1. https://www.youtube.com/watch?v=tzO245uWQxA
2. https://www.ri.cmu.edu/pub_files/pub3/baker_simon_2002_3/baker_simon_2002_3.pdf
3. http://rpg.ifi.uzh.ch/docs/teaching/2015/11_tracking.pdf
4. https://www.youtube.com/watch?v=1r8E9uAcn4E
5. https://www.youtube.com/watch?v=5VyLAH8BhF8
6. The coursework lecture of "lecture-tracking.pdf"