16-720A Computer Vision: Homework 3

Lucas-Kanade Tracking

Shayeree Sarkar

1.1) a) Given : $I_{t+1}(x'+\Delta p) \approx I_{t+1}(x') + \frac{\partial I_{t+1}(x')}{\partial x'^T} \cdot \frac{\partial W(x;p)}{\partial p^T} \Delta p$

Here, $\mathbf{p} = [p_x, p_y]^T$

$x' = W(\mathbf{x};\mathbf{p}) = \mathbf{x} + \mathbf{p}$

Hence $W(x;p)$ has 2 parameters p_1 & p_2

$$W(x;p) = \begin{pmatrix} 1 & 0 & p_1 \\ 0 & 1 & p_2 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} x + p_1 + 0 \\ 0 + p_2 + y \end{pmatrix}$$

$$= \begin{pmatrix} x + p_1 \\ y + p_2 \end{pmatrix}$$

$$\therefore \frac{\partial W}{\partial p^T} = \begin{bmatrix} \frac{\partial W_x}{\partial p_1} & \frac{\partial W_x}{\partial p_2} \\ \frac{\partial W_y}{\partial p_1} & \frac{\partial W_y}{\partial p_2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

(b) • To find 'A' & 'b':

In eqn (5) from write up, we know:

$$\underset{\Delta p}{\arg\min} \| A \Delta p - b \|_2^2$$

This above equation can be re-written as:

$$\underset{\Delta p}{\arg\min} \sum \| I_{t+1}(x; p+\Delta p) - I_t(x) \|_2^2$$

$$= \underset{\Delta p}{\arg\min} \sum \| I_{t+1}(x') + \frac{\delta I_{t+1}(x')}{\delta x'^T} \cdot \frac{\delta W(x;p)}{\delta p^T} \Delta p - I_t(x) \|_2^2$$

(We expand using Taylor series)

$$= \underset{\Delta p}{\arg\min} \sum \| \frac{\delta I_{t+1}(x')}{\delta x'^T} \cdot \frac{\delta W(x;p)}{\delta p^T} \Delta p - (I_t(x) - I_{t+1}(x')) \|_2^2$$

Hence $\rightarrow$

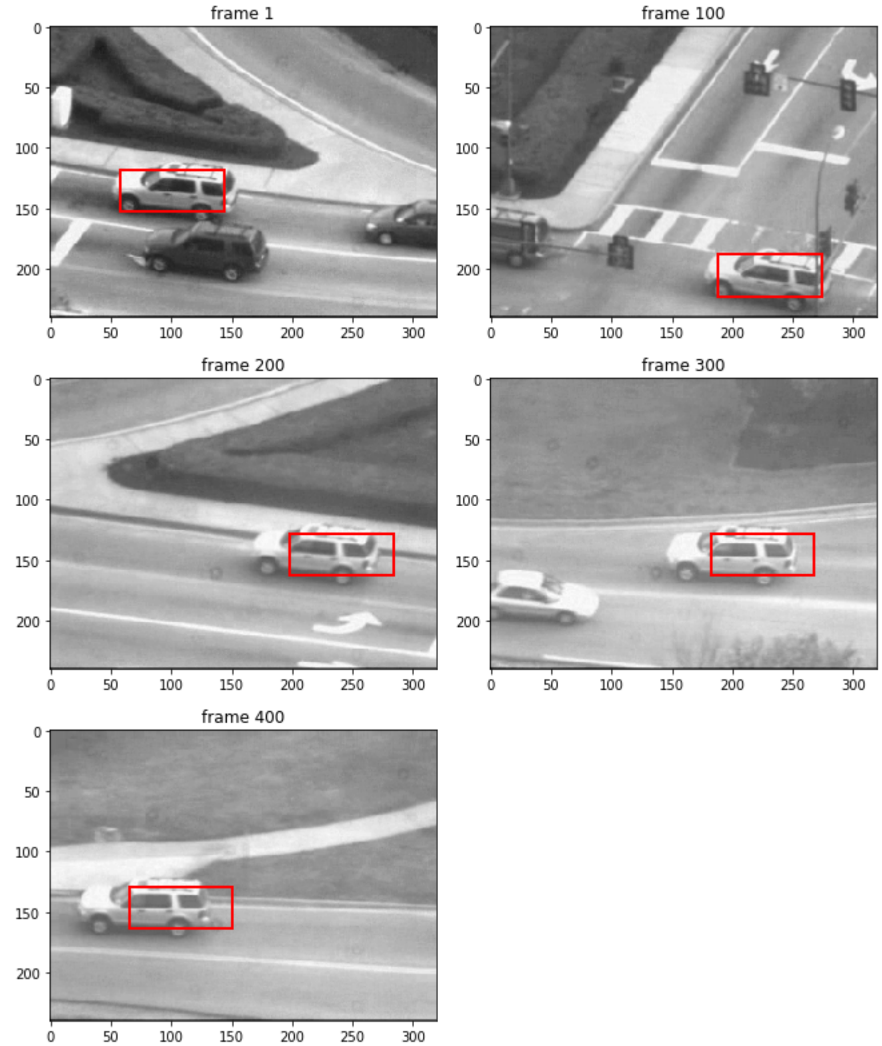
$$A = \begin{bmatrix} \frac{\delta I_{t+1}(x_1')}{\delta x_1'^T} & \cdots & 0^T \\ \vdots & \ddots & \vdots \\ 0^T & \cdots & \frac{\delta I_{t+1}(x_N')}{\delta x_N'^T} \end{bmatrix} \begin{bmatrix} \frac{\delta W(x_1;p)}{\delta p^T} \\ \vdots \\ \frac{\delta W(x_N;p)}{\delta p^T} \end{bmatrix}$$

where $x' = W(x;p) = x+p$

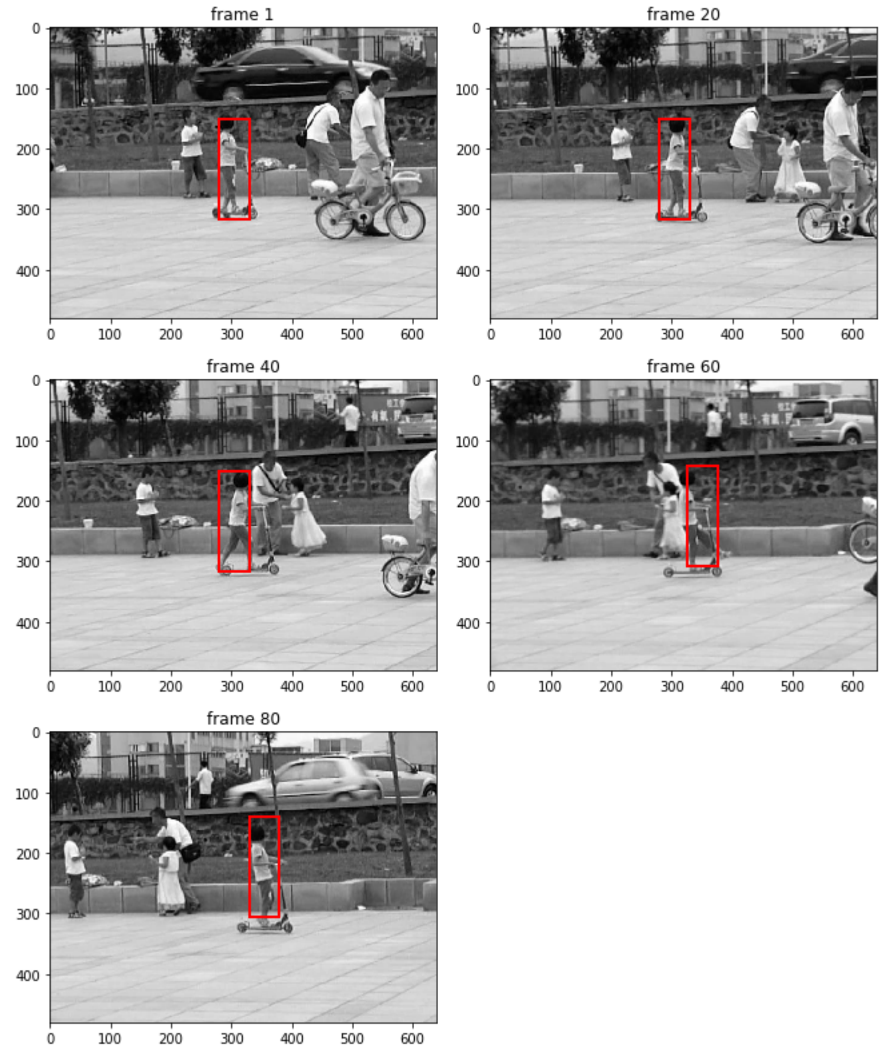
$$b = \begin{bmatrix} I_t(x_1) - I_{t+1}(x_1') \\ \vdots \\ I_t(x_N) - I_{t+1}(x_N') \end{bmatrix}$$

(c) To find the unique soln for Δp, A should span to all dimensions, that is $A^T A$ needs to be full rank, $\det(A^T A) \neq 0$

1.3) Tracking results for Test Car Sequence

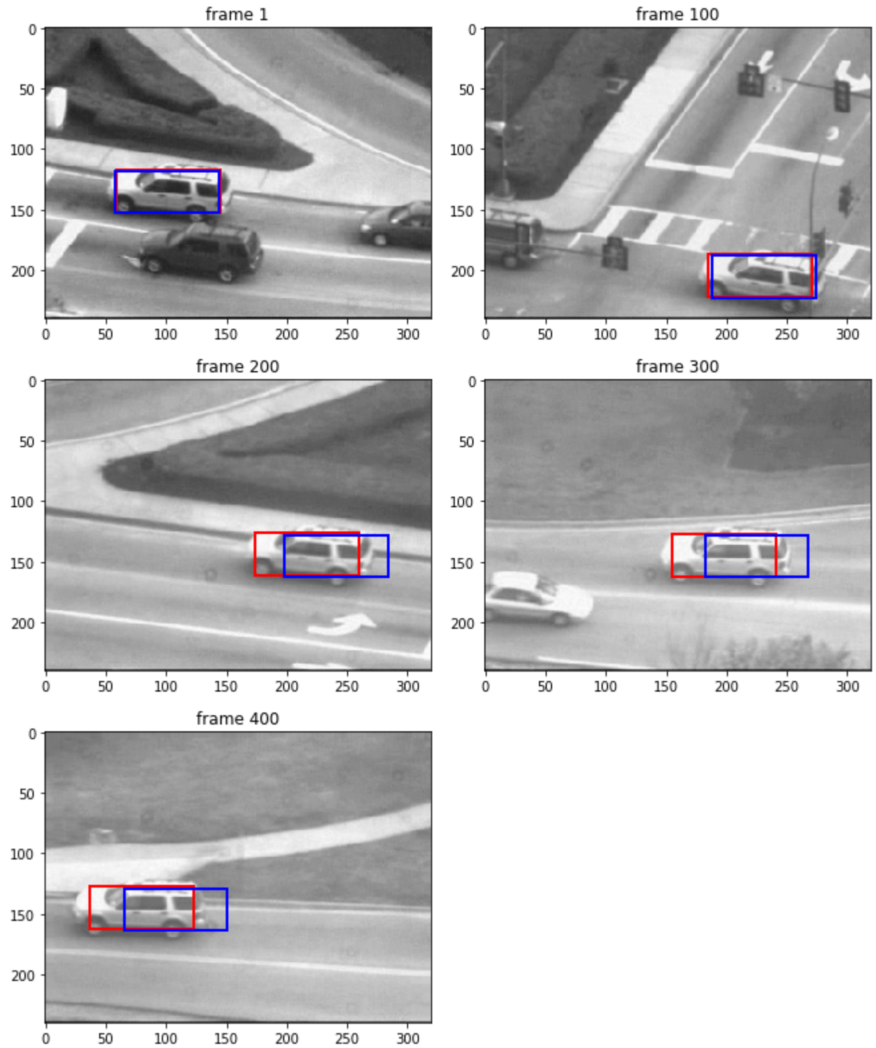


1.3) Tracking results for Test Girl Sequence:

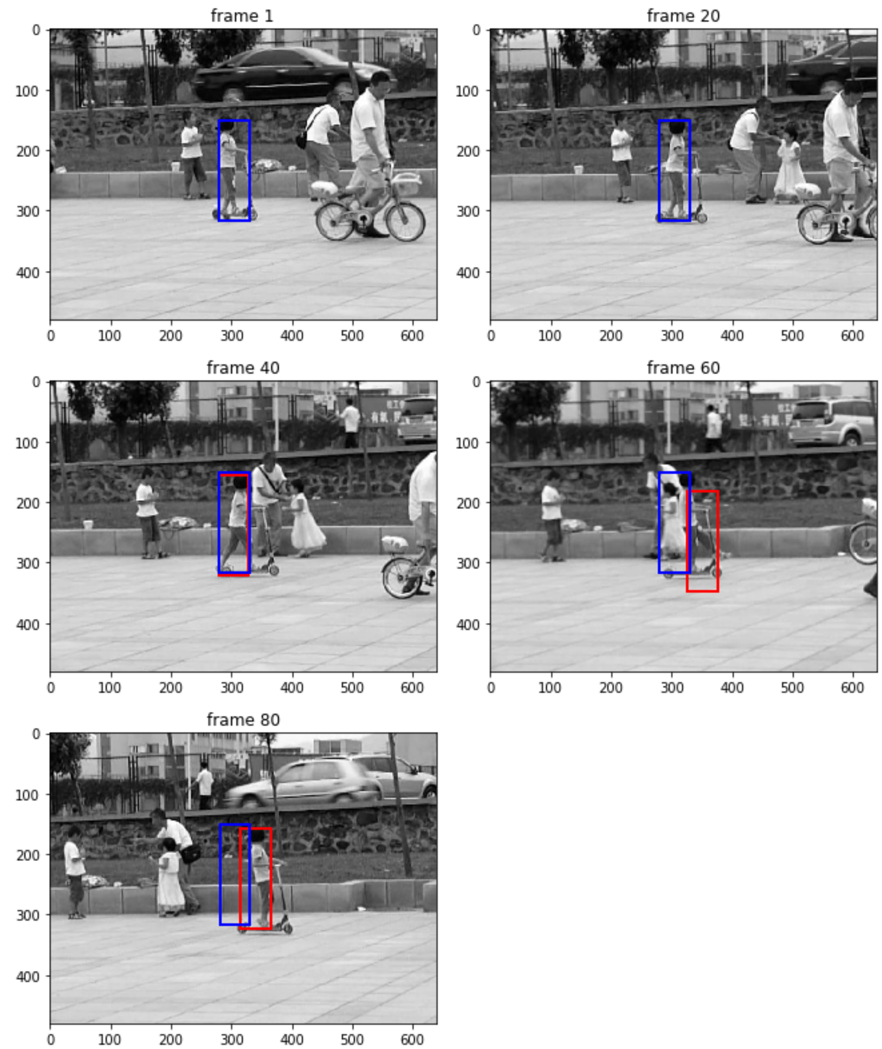


1.4) Tracking results for Test Car Sequence with Template Correction

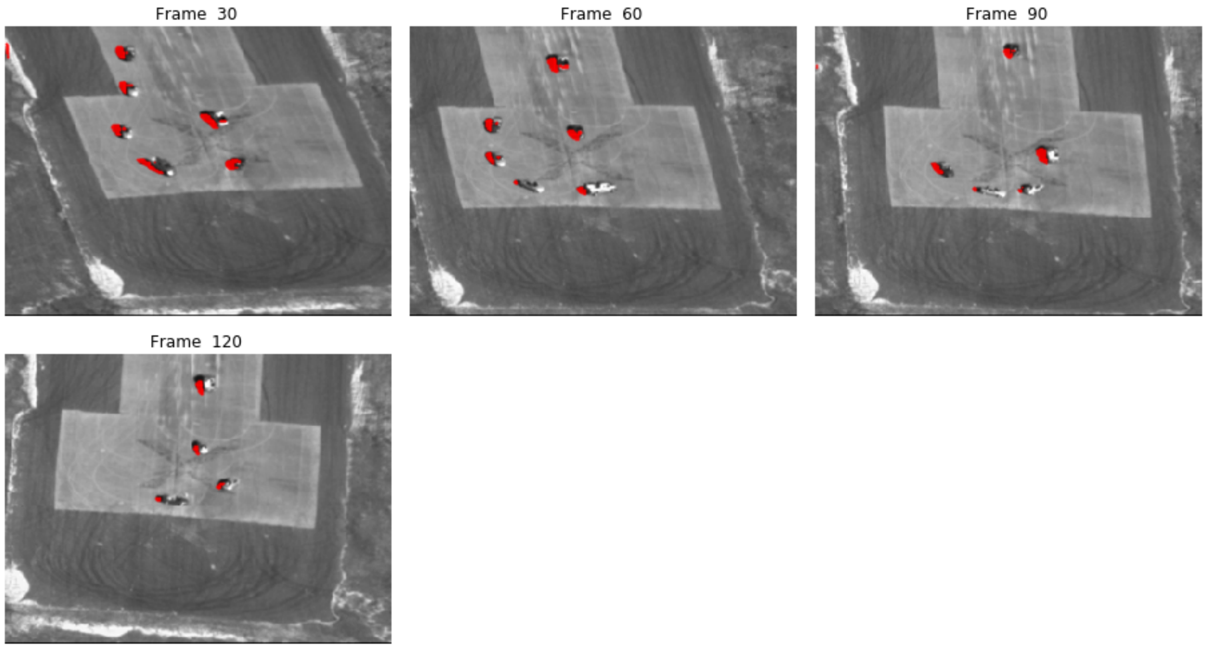
The blue rectangle represents the baseline tracker and the red rectangle represents the template correction tracker



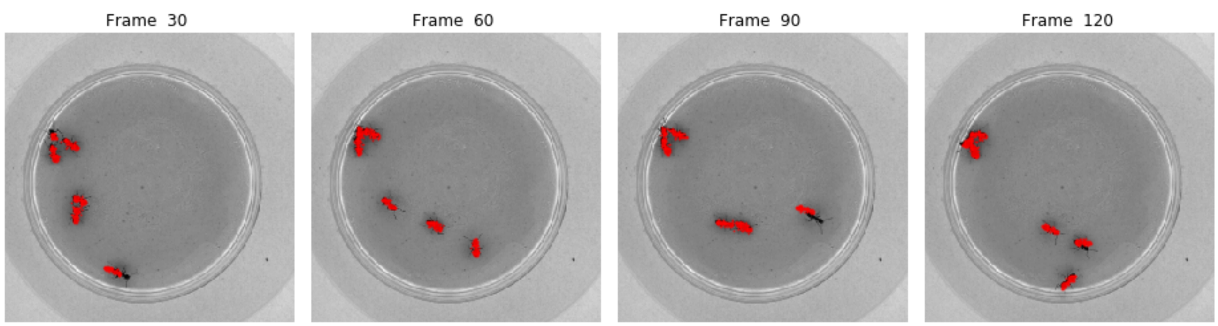
1.4) Tracking results for Test Girl Sequence with Template Correction
The blue rectangle represents the baseline tracker and the red rectangle represents the template correction tracker



2.2) The moving objects for the Aerial sequence are marked with red dots. The tolerance used for dominant motion estimation was 0.75



The moving objects for the Ant Sequence are marked with red dots. The tolerance used for dominant motion estimation was 0.75



3.1) A is a N x 6 matrix. With the conventional approach of computing Lucas Kanade Affine, it becomes computationally heavy to recompute and update A and b for every iteration of the gradient descent until optimal delta p is reached in a while loop. We can avoid re-computation of the gradient and the Hessian wrt to the Template image, since in the inverse composition we compute this once for the Template image outside the while loop. That is A can be precomputed only once and used with b which is updated in every gradient descent iteration. This leads to saving computational costs.
In my program the time taken for computation with the inverse compositional approach is half to that taken by the conventional approach.