Q1.1

- It is the Jacobian of the Warp:

$$\frac{\partial \boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})}{\partial \boldsymbol{p}^T} = \begin{bmatrix} \frac{\partial W_x}{\partial p_1} & \frac{\partial W_x}{\partial p_2} & \cdots & \frac{\partial W_x}{\partial p_D} \\ \frac{\partial W_y}{\partial p_1} & \frac{\partial W_y}{\partial p_2} & \cdots & \frac{\partial W_y}{\partial p_D} \end{bmatrix}$$

$$\|I_{t+1}(\boldsymbol{x}+\boldsymbol{p}) - I_t(\boldsymbol{x})\|_2^2$$
$$= \left\| I_{t+1}(\boldsymbol{x}') + \frac{\partial I_{t+1}(\boldsymbol{x}')}{I_{t+1}(\boldsymbol{x}'^T)} \frac{\partial \boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})}{\partial \boldsymbol{p}^T} \Delta\boldsymbol{p} - I_t(\boldsymbol{x}) \right\|_2^2$$
$$= \left\| \frac{\partial I_{t+1}(\boldsymbol{x}')}{I_{t+1}(\boldsymbol{x}'^T)} \frac{\partial \boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})}{\partial \boldsymbol{p}^T} \Delta\boldsymbol{p} + I_{t+1}(\boldsymbol{x}') - I_t(\boldsymbol{x}) \right\|_2^2$$
$$= \|\boldsymbol{A}\Delta\boldsymbol{p} - \boldsymbol{b}\|_2^2$$

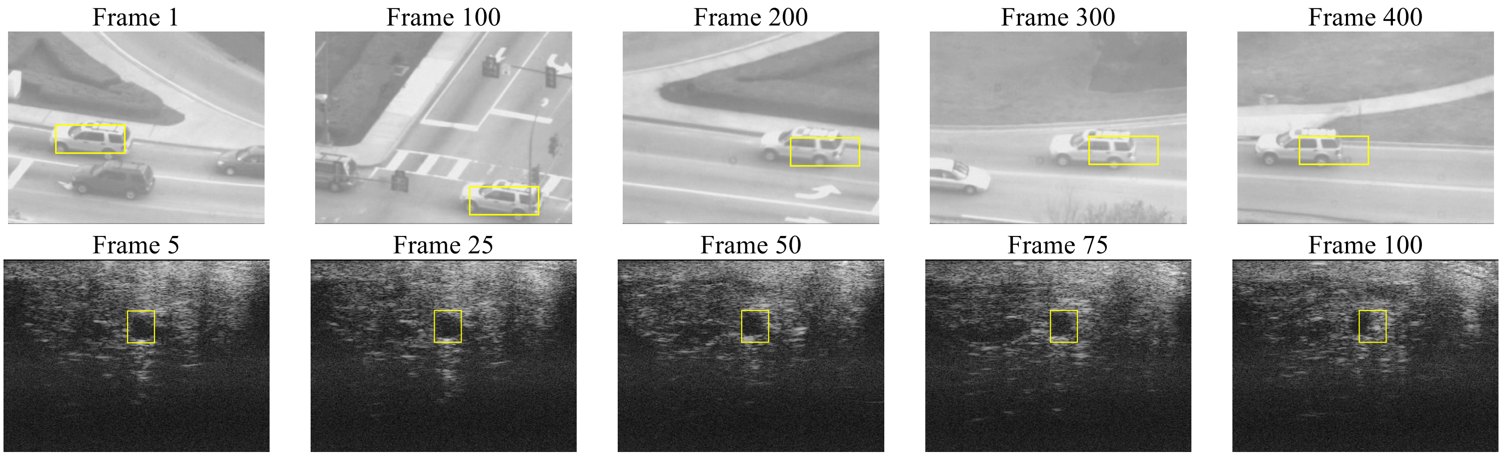
- From the derivative above, we can know that:

$$\mathbf{A} = \frac{\partial I_{t+1}(\boldsymbol{x}')}{I_{t+1}(\boldsymbol{x}'^T)} \frac{\partial \boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})}{\partial \boldsymbol{p}^T}$$
$$\boldsymbol{b} = I_t(\boldsymbol{x}) - I_{t+1}(\boldsymbol{x}')$$

- The condition that $\boldsymbol{A}^T\boldsymbol{A}$ must meet is this matrix should be non-singular or invertible.

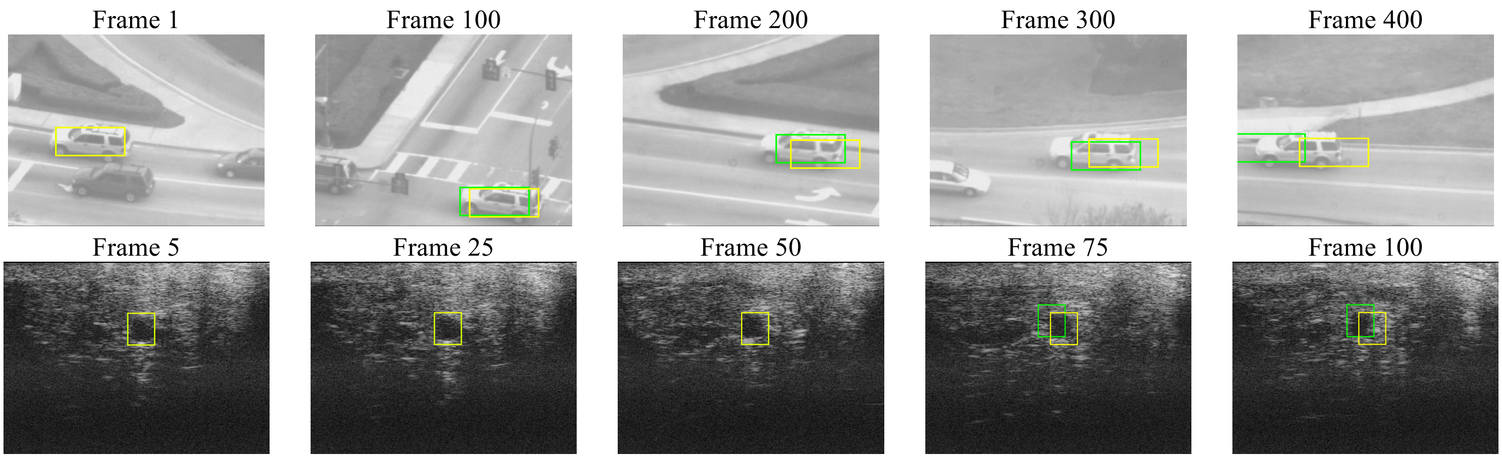
Q1.3

The results have been attached here:



Q1.4

The results have been attached. The yellow rectangles are created with the baseline tracker in Q1.3 and the green ones with the tracker in Q1.4. We can see that there is significant improvement after the correction.



Q2.1

$$I_{t+1}(\boldsymbol{x}) = I_t(\boldsymbol{x}) + \sum_{k=1}^{K} w_k B_k(\boldsymbol{x})$$

$$\sum_{k=1}^{K} w_k B_k(\boldsymbol{x}) = I_{t+1}(\boldsymbol{x}) - I_t(\boldsymbol{x})$$

$$w_1 B_1(\boldsymbol{x}) + w_2 B_2(\boldsymbol{x}) + \cdots + w_k B_k(\boldsymbol{x}) + \cdots + w_K B_K(\boldsymbol{x}) = I_{t+1}(\boldsymbol{x}) - I_t(\boldsymbol{x})$$

$$B_k(\boldsymbol{x}) w_1 B_1(\boldsymbol{x}) + B_k(\boldsymbol{x}) w_2 B_2(\boldsymbol{x}) + \cdots + B_k(\boldsymbol{x}) w_k B_k(\boldsymbol{x}) + \cdots + B_k(\boldsymbol{x}) w_K B_K(\boldsymbol{x})$$
$$= B_k(\boldsymbol{x})(I_{t+1}(\boldsymbol{x}) - I_t(\boldsymbol{x}))$$

Since B_k's are orthobases

$$w_1 0 + w_2 0 + \cdots + w_k \|B_k(\boldsymbol{x})\|^2 + \cdots + w_K 0 = B_k(\boldsymbol{x})(I_{t+1}(\boldsymbol{x}) - I_t(\boldsymbol{x}))$$

$$w_k = \frac{B_k(\boldsymbol{x})}{\|B_k(\boldsymbol{x})\|^2}(I_{t+1}(\boldsymbol{x}) - I_t(\boldsymbol{x}))$$

Q2.3

The results have been attached, the green rectangles are the results of LucasKanadeBasis and the yellow rectangles are results of LucasKanadeInverseCompositional.

Frame 1

Frame 100

Frame 200

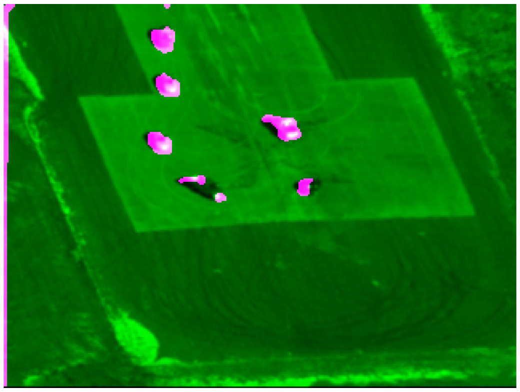
Frame 300

Frame 400

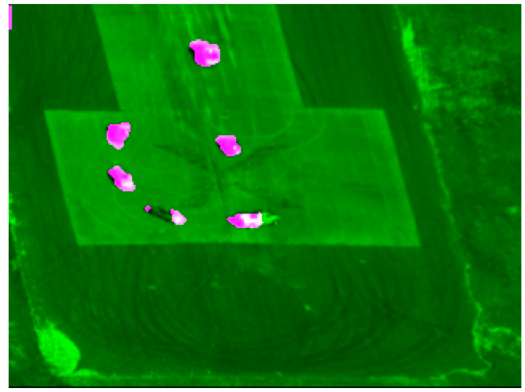
Q3.3

Lucas-Kanade Tracking with Apprearance Basis:

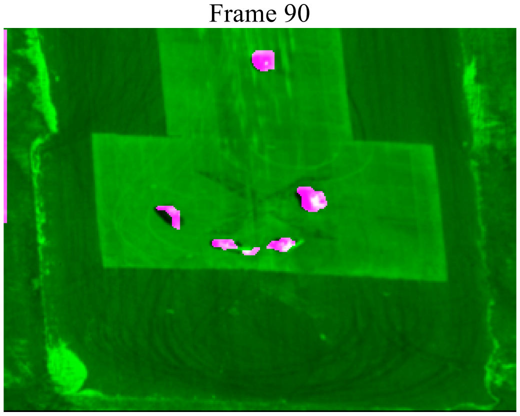
Frame 30



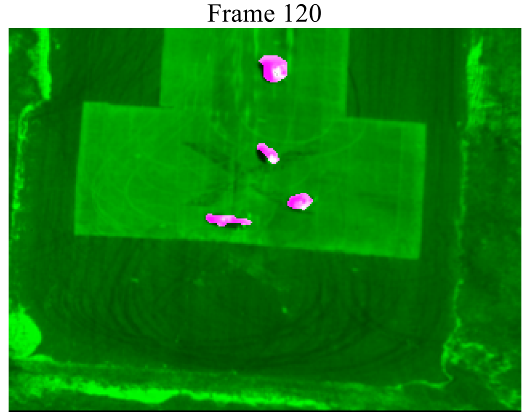
Frame 60



Frame 90

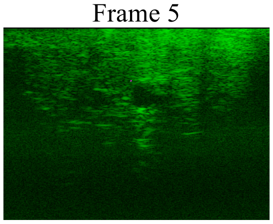


Frame 120

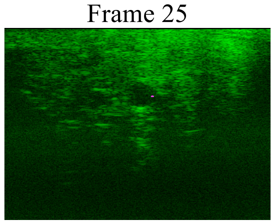


Lucas-Kanade Tracking of Affine Motion:

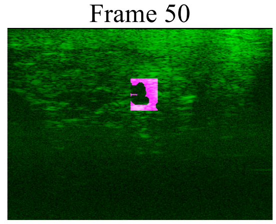
Frame 5



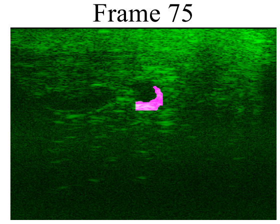
Frame 25



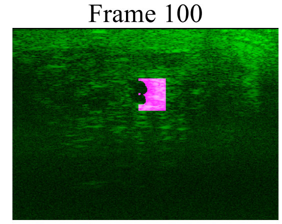
Frame 50



Frame 75



Frame 100



Q4.1

The inverse compositional algorithm is farm more computationally efficient than Lucas-Kanade algorithm, because the most time consuming step in Lucas-Kanade algorithm s the computation of the Hessian can be performed once as a pre-computation in inverse compositional algorithm. And the evaluation of the gradient of ∇T could be once pre-computation two, which would be slightly quicker than in the Lucas-Kanade algorithm, which would calculate ∇T in each iteration. The only additional cost is not substantial and could always negligible. As total, the overall cost of the inverse compositional algorithm is $\mathrm{O}(n^2\mathrm{N})$, which saves a lot compare to the classic Lucas-Kanade algorithm, which is $\mathrm{O}(n^3 + n^2\mathrm{N})$.