Q1.1

* It is the Jacobian of the Warp:
* From the derivative above, we can know that:
* The condition that must meet is this matrix should be non-singular or invertible.

Q1.3

The results have been attached here:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Frame 1 | Frame 100 | Frame 200 | Frame 300 | Frame 400 |
|  |  |  |  |  |
| Frame 5 | Frame 25 | Frame 50 | Frame 75 | Frame 100 |
|  |  |  |  |  |
|  |  |  |  |  |

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.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Frame 1 | Frame 100 | Frame 200 | Frame 300 | Frame 400 |
|  |  |  |  |  |
| Frame 5 | Frame 25 | Frame 50 | Frame 75 | Frame 100 |
|  |  |  |  |  |
|  |  |  |  |  |

Q2.1

Since s are orthobases

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 could be once pre-computation two, which would be slightly quicker than in the Lucas-Kanade algorithm, which would calculate 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 , which saves a lot compare to the classic Lucas-Kanade algorithm, which is .