Mohit Kumar
SR No. 04-01-03-10-51-21-1-19825
MTech Artificial Intelligence

Assignment 1: Video Prediction using Optical Flow

In this assignment we will predict a future video frame given the two past frames using optical flow. We will estimate the forward optical flow (flow from Frame n to Frame n+1) and assume linearity of motion to predict the future frame (Frame n+2) on corridor and sphere datasets.
To estimate the optical flow we will use two popular classical methods:

1. Lucas-Kanade optical flow algorithm
2. Discrete Horn-Schunck optical flow algorithm

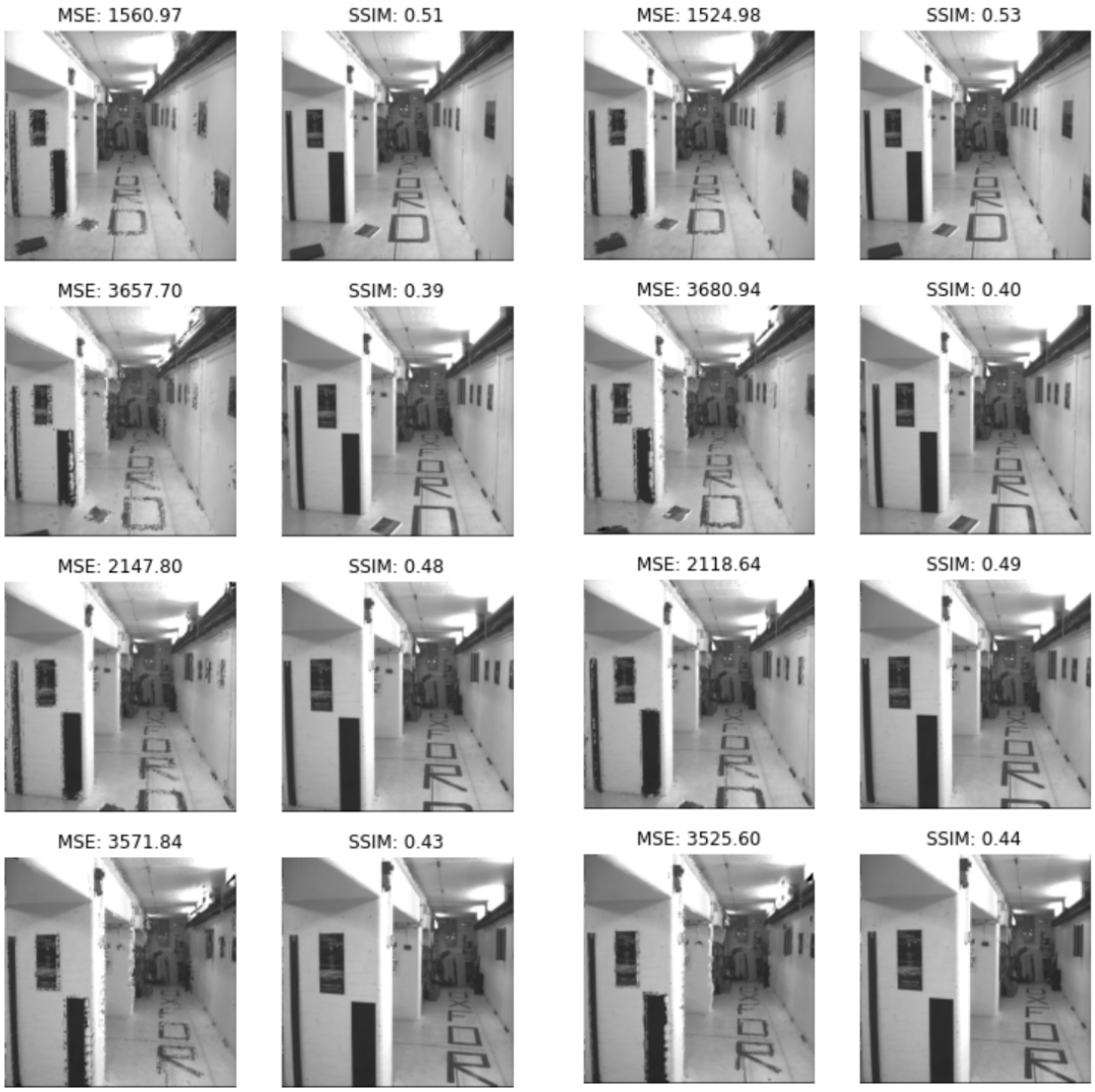
Since it was mentioned in the problem statement that we can use any library functions to estimate the optical flow. I have used the implementation of single stage Horn-Schunck and Lucas-Kanade algorithms from `https://github.com/vineeths96/Video-Interpolation-using-Optical-Flow`

For the Lucas Kanade algorithm, the standard algorithm is implemented along with a small modification. When we solve for the least squares solution for $Ax = b$, we add an additional constraint that we find the solution to the linear systems only if the smallest eigenvalues of $A^T A$ is greater than a threshold τ.
I have reported mean square error and SSIM metric to compare the predicted and ground truth video frames.

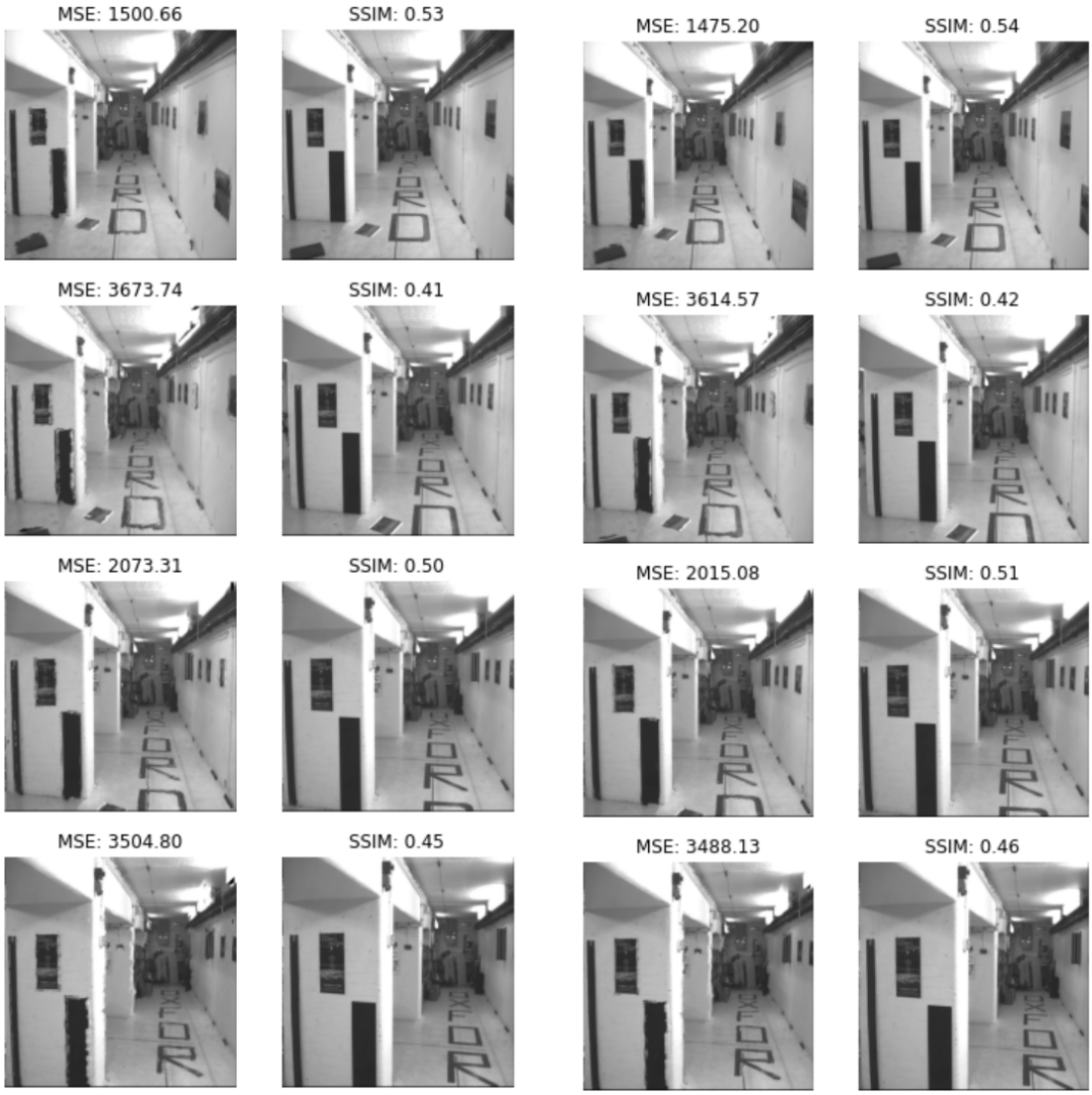
Note: In all figures in this report the left image is the predicted frame and the right image is the ground truth frame.

- **Lucas-Kanade optical flow algorithm Hyperparameters**
  I have experimented with window sizes of 5, 8, 11, 14.

– *corridor dataset*



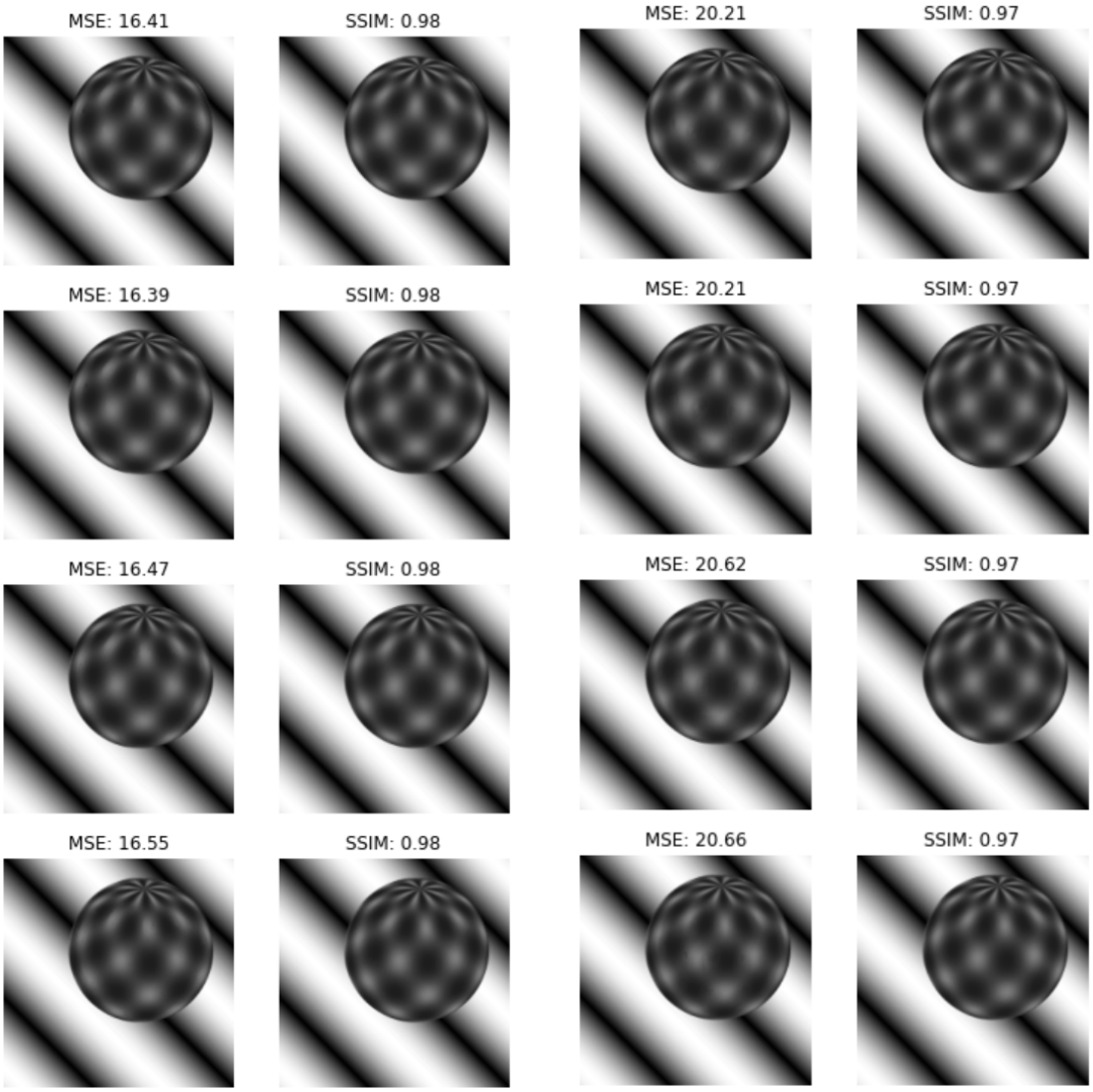
(a) window size = 5 (b) window size = 8



(a) window size = 11

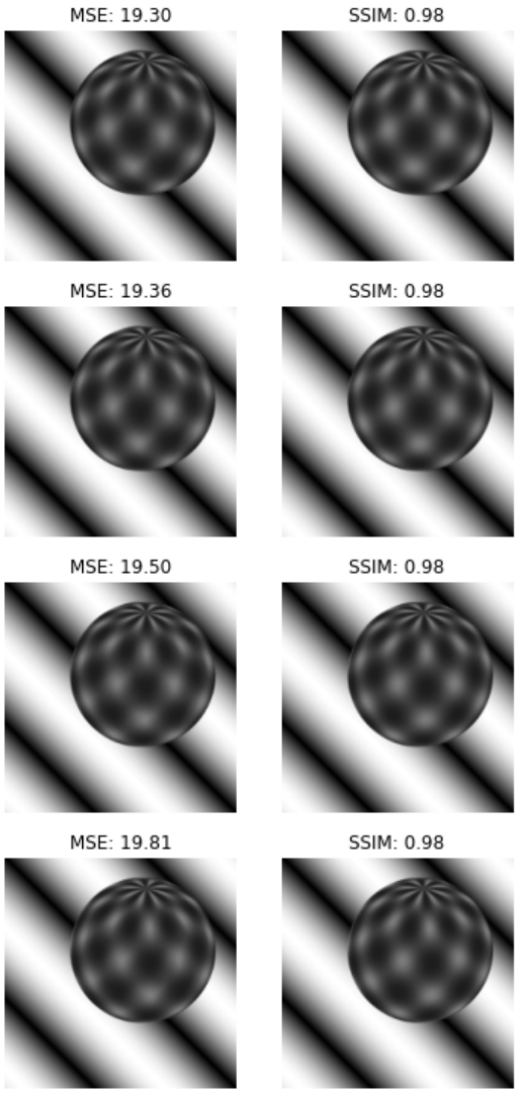
(b) window size = 14

– *sphere dataset*

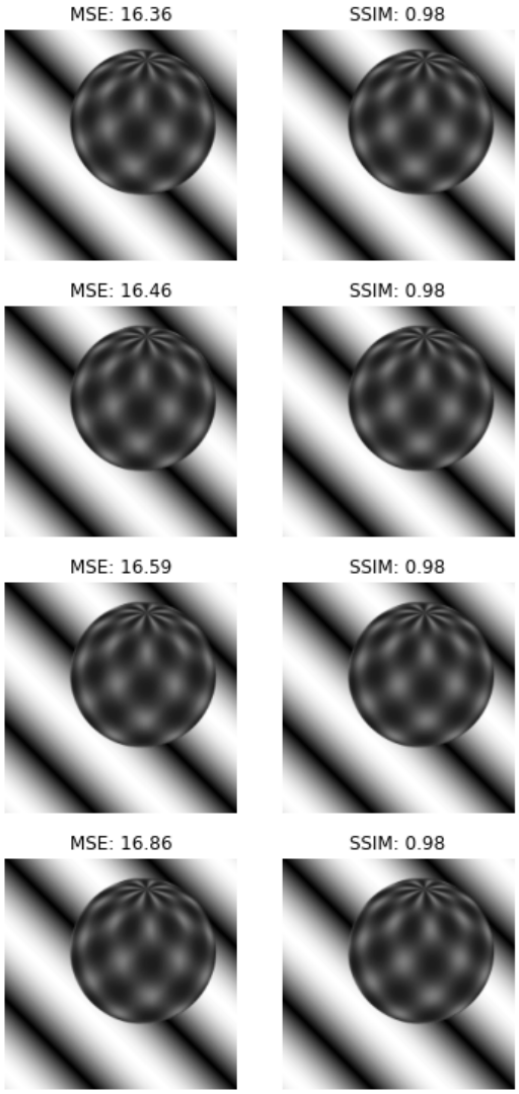


(a) window size = 5

(b) window size = 8

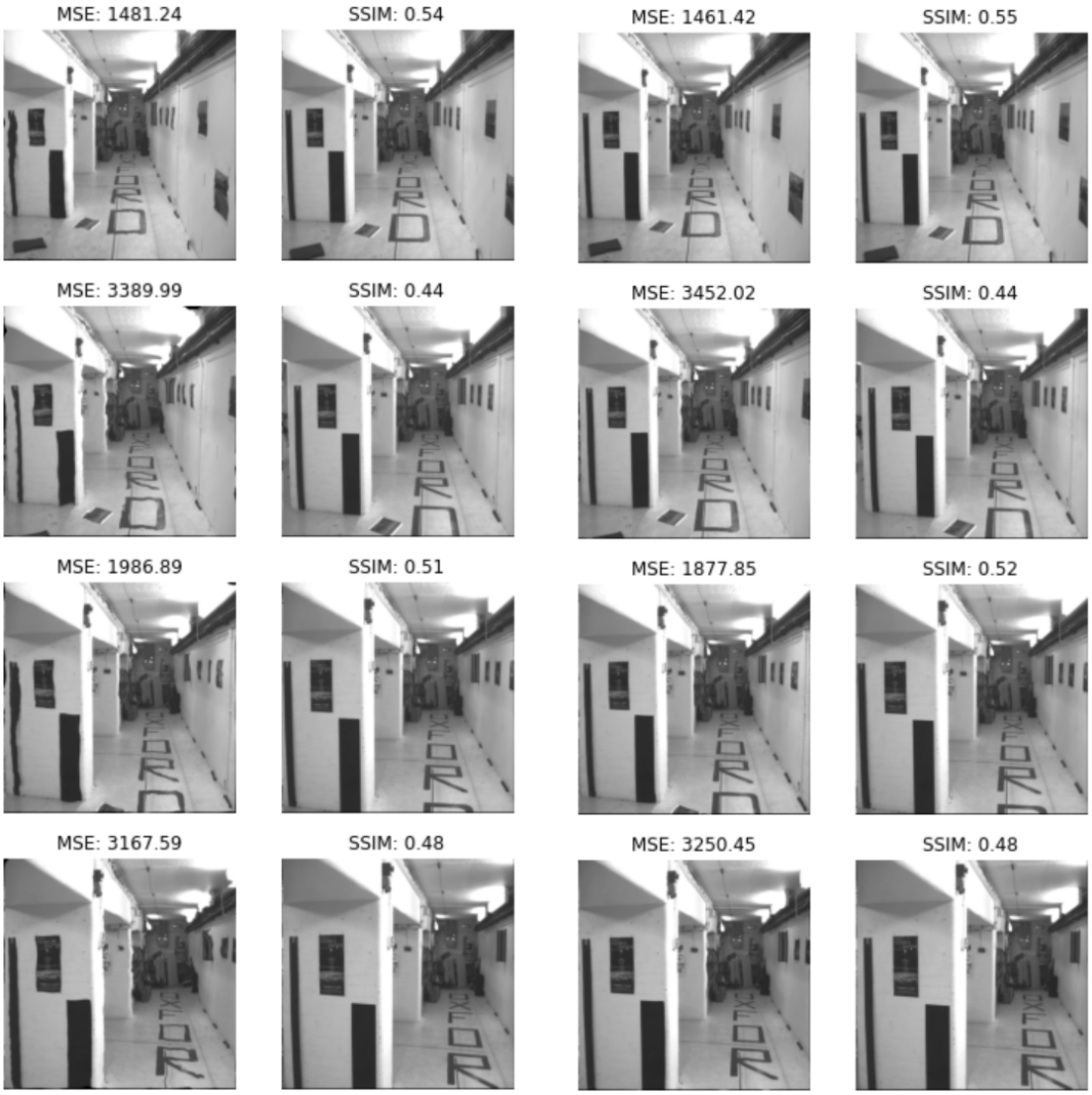


(a) window size = 11

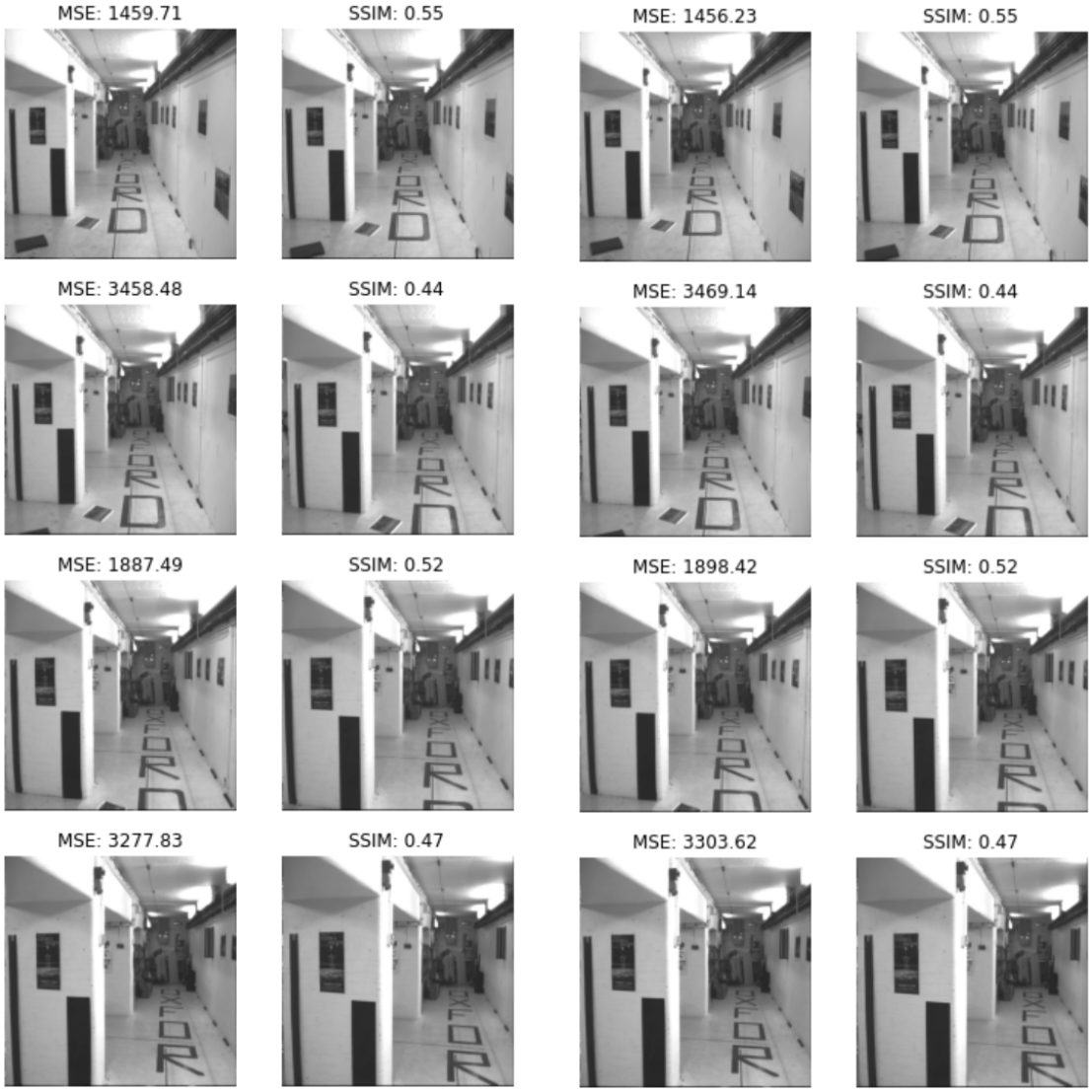


(b) window size = 14

- **Horn-Schunck optical flow algorithm Hyperparameters**
  I have experimented with lambda values of 0.01, 0.25, 0.5, 1.
  - *corridor dataset*



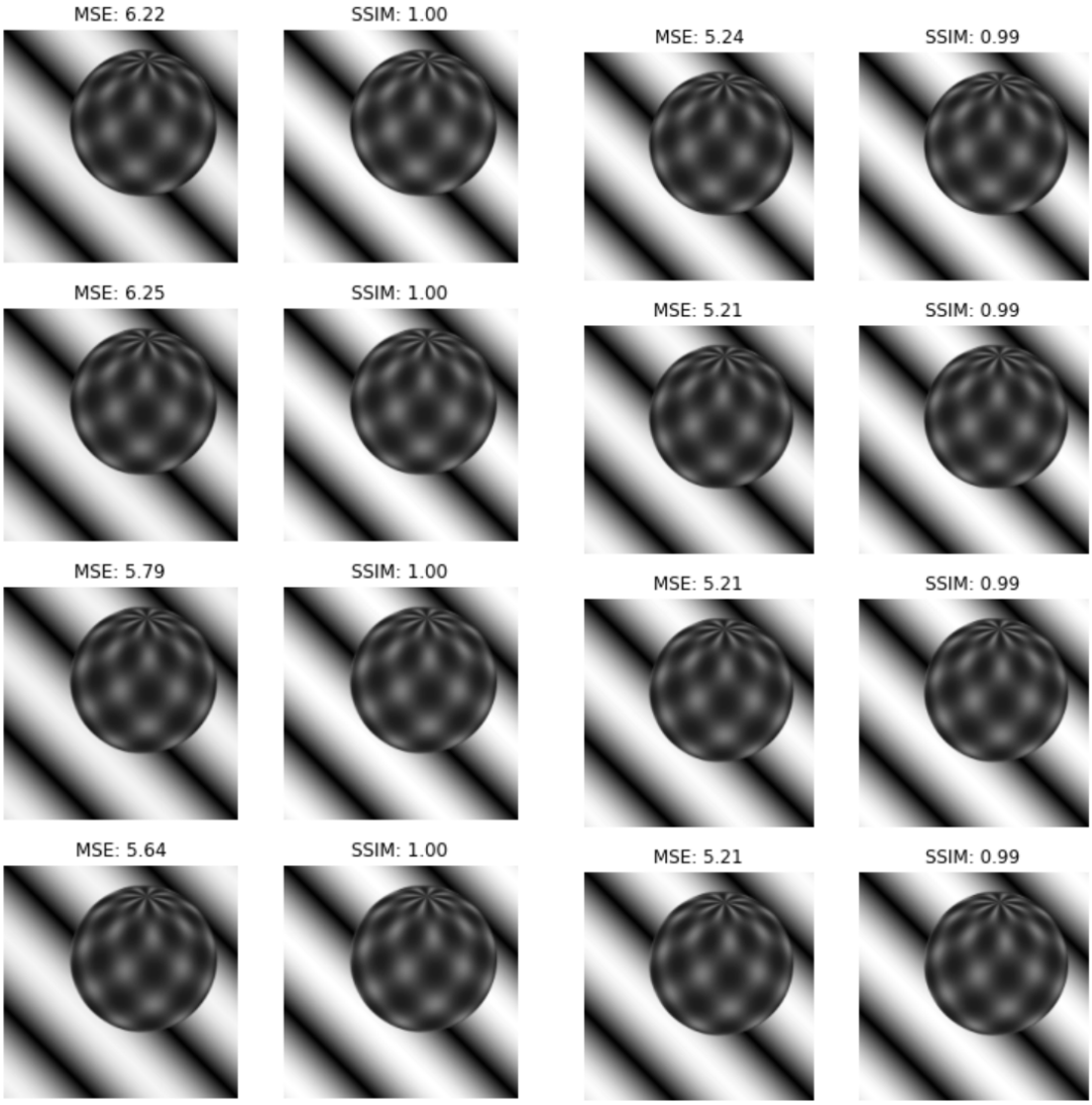
(a) lambda = 0.01 (b) lambda = 0.25



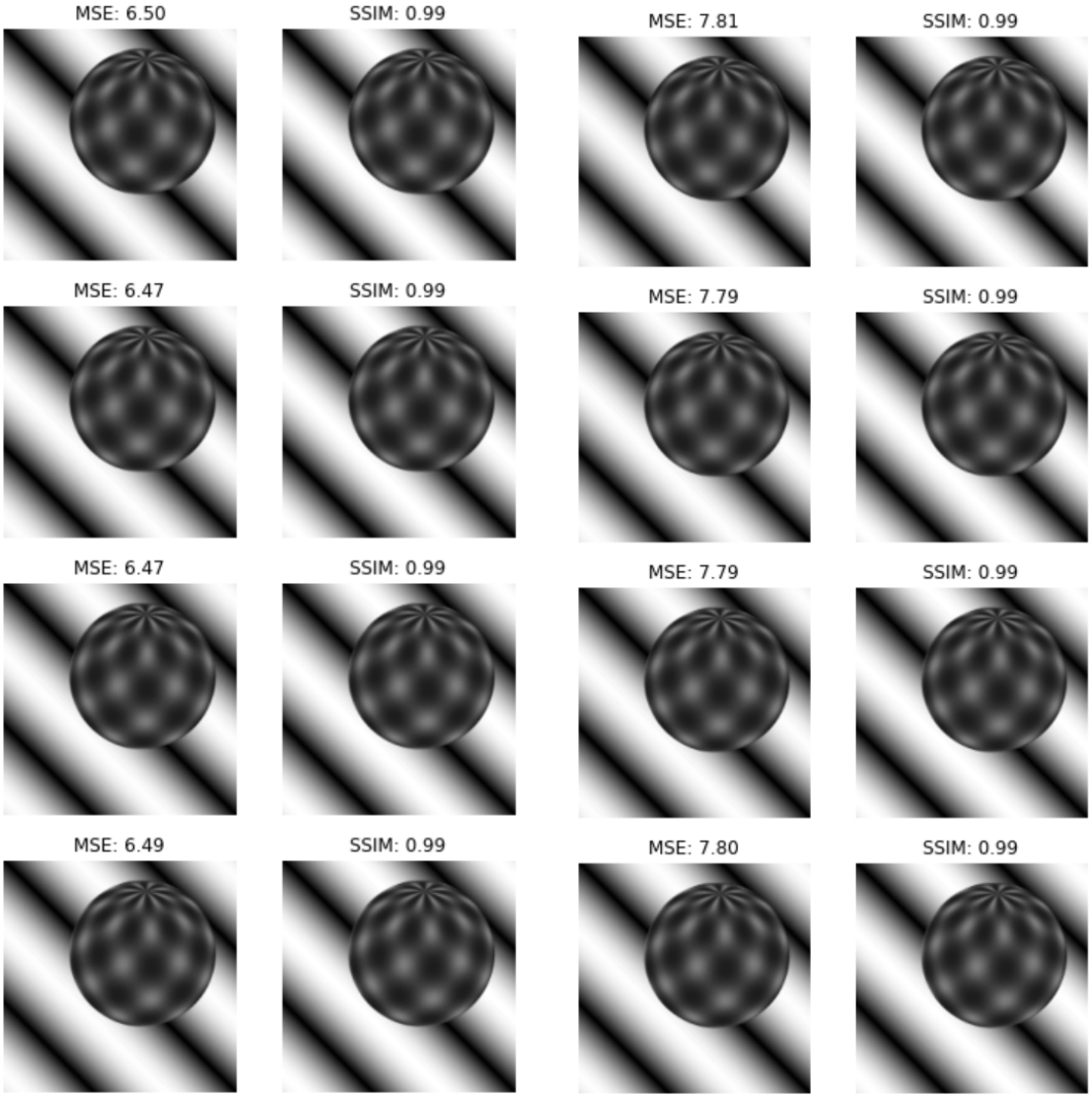
(a) lambda = 0.5

(b) lambda = 1

– *sphere dataset*



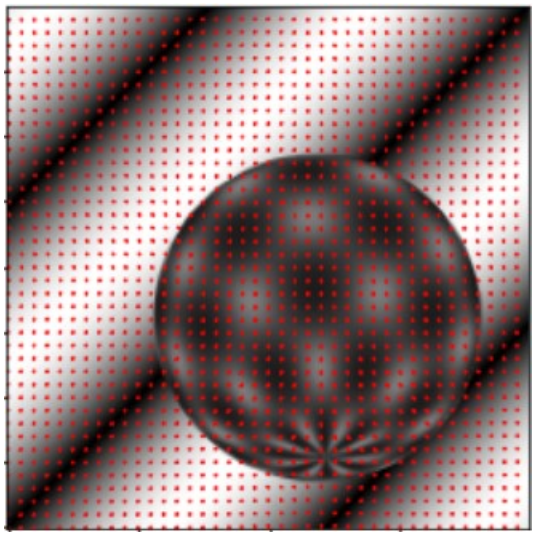
(a) lambda = 0.01 (b) lambda = 0.25



(a) lambda = 0.5

(b) lambda = 1

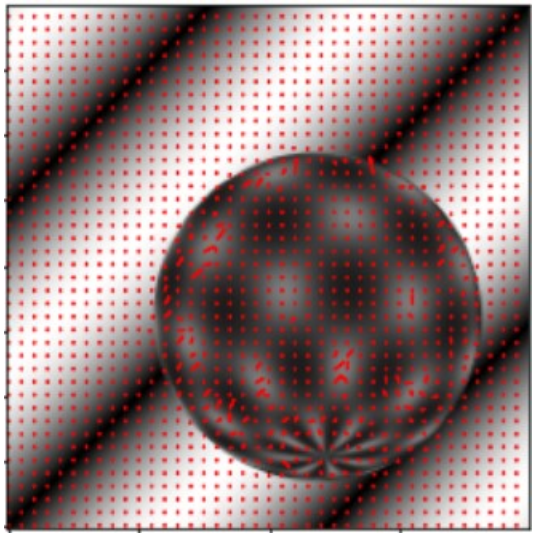
(a) Corridor



(b) Sphere

Figure 9: Estimated Optical Flow(Horn Schunck)

(a) Corridor



(b) Sphere

Figure 10: Estimated Optical Flow(Lucas Kanade)

Observations and Answers to questions in the assignment

While using the Lucas Kanade algorithm for the corridor dataset, we observe that there is an error of reconstruction at the solid edges at the boundaries. This is due to the aperture problem where at the edges, we have a single gradient direction. For the sphere dataset, the motion of the sphere is slow and we have minimal occlusions, we notice that there are no significant errors in the reconstruction, unlike the corridor dataset. From the plots, we observe that there is no significant difference in the quality of reconstructed frames by varying the window size. The quality of the predicted frames in the corridor data is really poor. This could be due to the fact that the motion is fast and pixels are moving out of/moving into the frame. The quality of predicted frames for the sphere dataset is comparable to original frame and is much better compared to that of corridor dataset. This can be attributed to the slow motion and no occlusions. The general observations regarding the reconstruction error hold true for the Discrete Horn Schunck algorithm as well. Here also we do not observe any significant difference in the quality of the predicted frame by varying the lambda value. Horn Schunck algorithm appears to perform slightly better and runs slower when compared to the Lucas Kanade algorithm. The predicted frames of corridor dataset are of low quality. The quality of predicted frames for the sphere dataset are comparable to original frame and are much better compared to that of corridor dataset. The numerical performance metrics for both the algorithms can be seen in the respective figures along with the variation in the respective hyperparameter values. We are not able to predict optical flow for pixels which are either severely occluded or disoccluded in between frames.