E9 208 Digital Video: Perception and Algorithms
Assignment 2

Vineeth S
M. Tech Artificial Intelligence
SR. No. 16543

Abstract

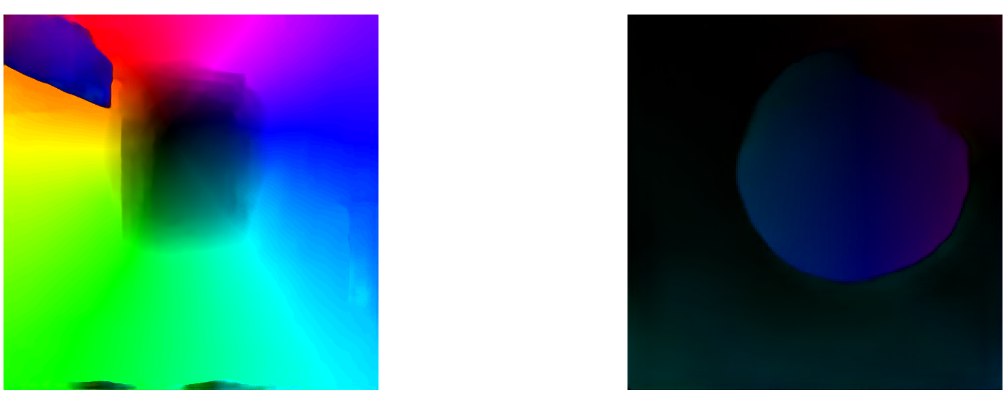
In this assignment we explore the idea of interpolating video frames using deep optical flow on Corridor dataset and Sphere dataset. To estimate the optical flow we use pre-trained FlowNet2 deep learning model and experiment by fine-tuning it. We interpolate the video frames by a method similar to one proposed in [1]. We observe that the quality of interpolated frames is comparable to original with both the Sphere data and Corridor data. Documented code will also be available at `https://github.com/vineeths96/Video-Interpolation-using-Deep-Optical-Flow`.

Problem 1: Pre-Trained Optical Flow Estimation

In this section, we discuss about the performance of frame interpolation based on pre-trained deep optical flow model. We use FlowNet2 [2] model made publicly available by NVIDIA [3]. Note that the code in *flownet2* directory of the repository contains publicly available code modified to suit our needs. This was necessary not to break the pipeline they have built for the model. We use the method proposed in [1] for interpolation of the frames.

Corridor Figures 2 and 3 show the interpolated frames for the corridor dataset with deep optical flow estimation. We can observe that the quality of reconstruction is really good compared to that of the classical optical flow based reconstruction. We are able to reconstruct the original frame back with great detail barring some error at the edges. This can be mainly attributed to pixels moving in or out of the frame.

Sphere Figures 4 and 5 show the interpolated frames for the sphere dataset with deep optical flow estimation. We can observe that the quality of reconstruction is really good and slightly exceeds the quality of the classical optical flow based reconstruction (which itself was of good quality). We are able to reconstruct the original frame back with great detail with visually no error. However, due to processing of the images the background of images was unable to be reconstructed. This does not have any impact on the objectives of our experiments.

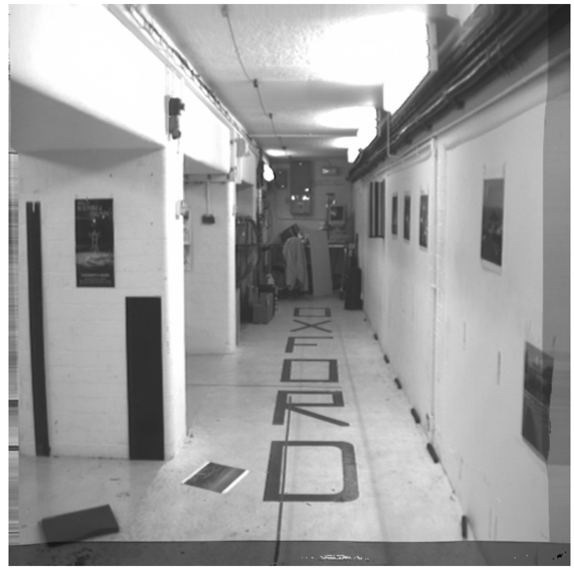


(a) Corridor (b) Sphere

Figure 1: Sample of estimated optical flows

Figure 1 shows the estimated optical flow for a random frame for the corridor and sphere data.

(a) Frame 1



(b) Interpolated Frame 1

(c) Frame 3

(d) Interpolated Frame 3

(e) Frame 5

(f) Interpolated Frame 5

Figure 2: Pre-Trained Optical flow frame interpolation: Corridor

(a) Frame 7

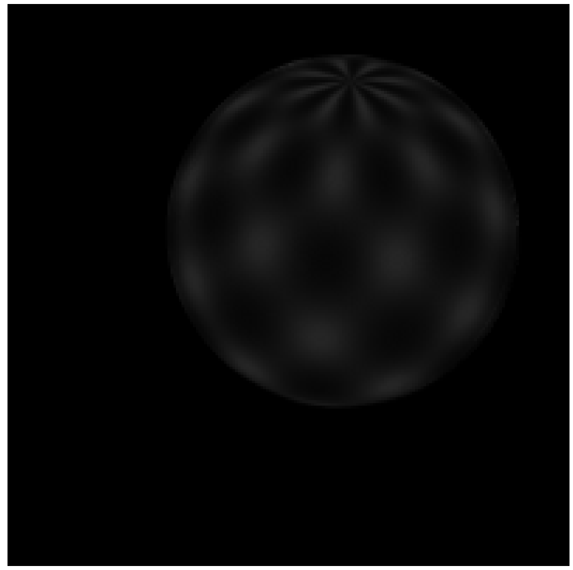
(b) Interpolated Frame 7

(c) Frame 9

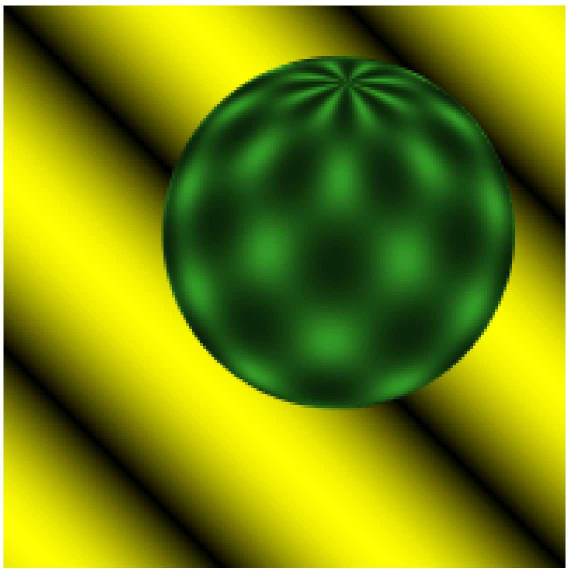
(d) Interpolated Frame 9

Figure 3: Pre-Trained Optical flow frame interpolation: Corridor

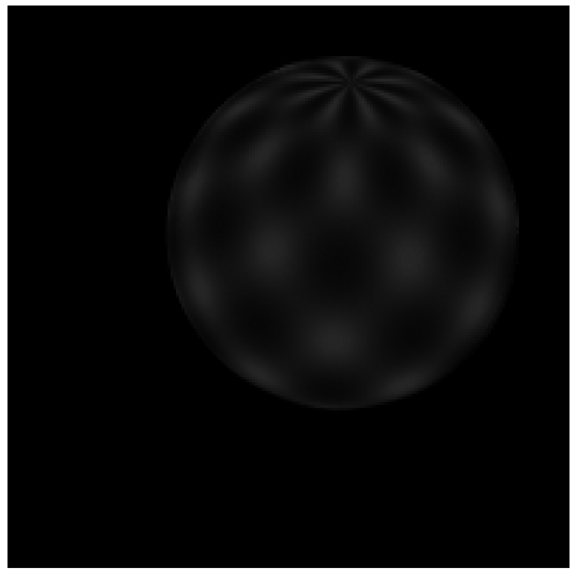
(a) Frame 1



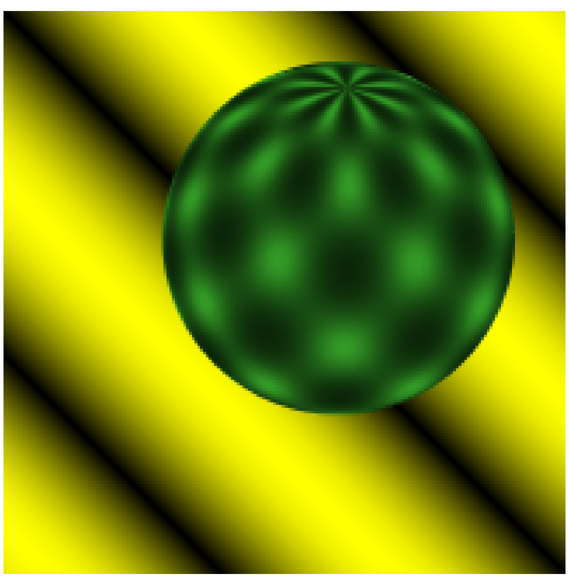
(b) Interpolated Frame 1



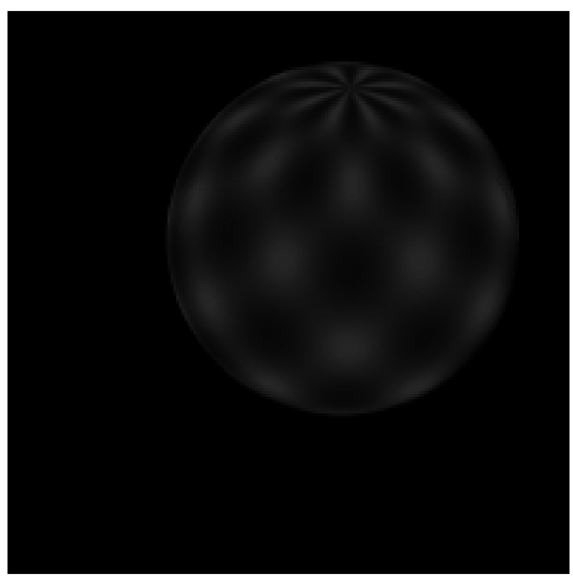
(c) Frame 3



(d) Interpolated Frame 3

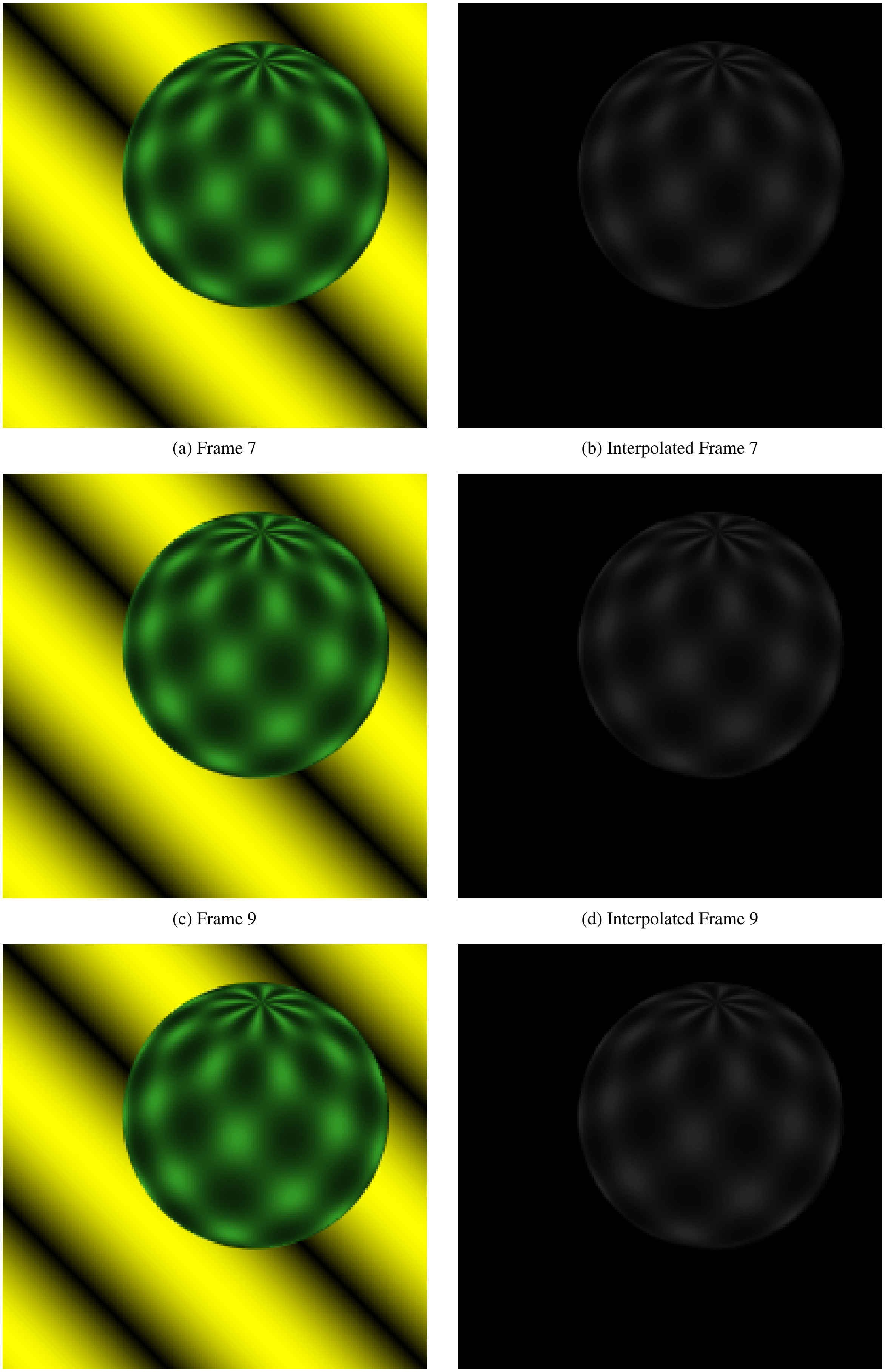


(e) Frame 5



(f) Interpolated Frame 5

Figure 4: Pre-Trained Optical flow frame interpolation: Sphere



(a) Frame 7
(b) Interpolated Frame 7
(c) Frame 9
(d) Interpolated Frame 9
(e) Frame 11
(f) Interpolated Frame 11

Figure 5: Pre-Trained Optical flow frame interpolation: Sphere

Problem 2: Fine Tuned Optical Flow Algorithm Estimation

In this section, we discuss about the performance of frame interpolation based on fine-tuning the pre-trained deep optical flow model. We use FlowNet2 [2] model made publicly available by NVIDIA [3]. Note that the code in *flownet2* directory of the repository contains publicly available code modified to suit our needs. This was necessary not to break the pipeline they have built for the model. We use the method proposed in [1] for interpolation of the frames. We use the unsupervised loss function proposed in [4], which is the sum of photometric loss and smoothness loss, as the loss function for fine-tuning the deep model.

$$\mathcal{L}(\mathbf{u}, \mathbf{v}; I(x,y,t), I(x,y,t+1)) = l_{photometric}(\mathbf{u}, \mathbf{v}; I(x,y,t), I(x,y,t+1)) + l_{smoothness}(\mathbf{u}, \mathbf{v})$$

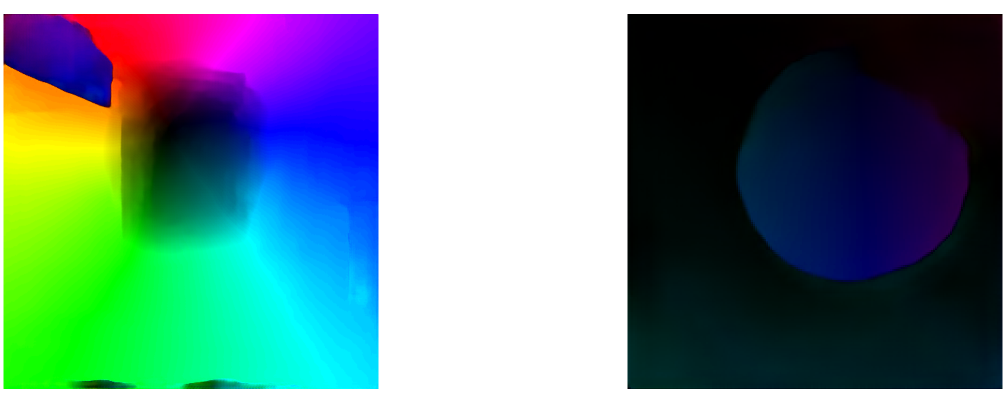
$$l_{photometric}(\mathbf{u}, \mathbf{v}; I(x,y,t), I(x,y,t+1)) = \sum_{i,j} \rho(I(i,j,t) - I(i+u_{i,j}, j+v_{i,j}, t+1))$$

$$l_{smoothness}(\mathbf{u}, \mathbf{v}) = \sum_{j}^{H} \sum_{i}^{W} \rho(u_{i,j} - u_{i+1,j}) + \rho(u_{i,j} - u_{i,j+1}) + \rho(v_{i,j} - v_{i+1,j}) + \rho(v_{i,j} - v_{i,j+1})$$

We consider the penalty function to be Charbonnier function $\rho(x) = (x^2 + \epsilon^2)^\alpha$.

Corridor Figures 7 and 8 show the interpolated frames for the corridor dataset with deep optical flow estimation with fine-tuning (for 10 iterations). We can observe that the quality of reconstruction is really good compared to that of the classical optical flow based reconstruction. However, we could not see any visual quality improvement over the pre-trained model results, which can be attributed to the well trained model. We observe no significant visual change in quality with respect to change in the number of iterations of fine-tuning. The images are not presented here since the results are not interesting.

Sphere Figures 9 and 10 show the interpolated frames for the sphere dataset with deep optical flow estimation with fine-tuning (for 10 iterations). We can observe that the quality of reconstruction is really good and slightly exceeds the quality of the classical optical flow based reconstruction (which itself was of good quality). Again, we could not see any visual quality improvement over the pre-trained model results, which can be attributed to the well trained model. We observe no significant visual change in quality with respect to change in the number of iterations of fine-tuning. The images are not presented here since the results are not interesting. However, due to processing of the images the background of images was unable to be reconstructed. This does not have any impact on the objectives of our experiments.



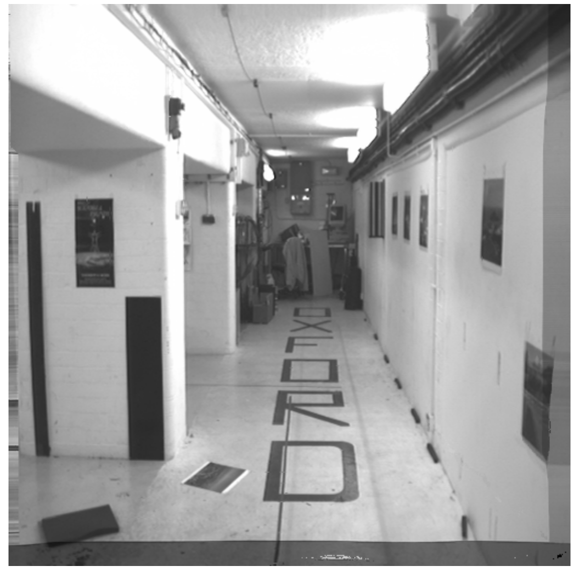
(a) Corridor

(b) Sphere

Figure 6: Sample of estimated optical flows

Figure 6 shows the estimated optical flow for a random frame for the corridor and sphere data after fine-tuning. Optical flow for all frames and all the interpolated frames are available at *results* directory of the repository. Only a subset of the result images are presented here keeping space constraints in mind.

(a) Frame 1



(b) Interpolated Frame 1

(c) Frame 3

(d) Interpolated Frame 3

(e) Frame 5

(f) Interpolated Frame 5

Figure 7: Fine Tuned Optical flow frame interpolation: Corridor

(a) Frame 7

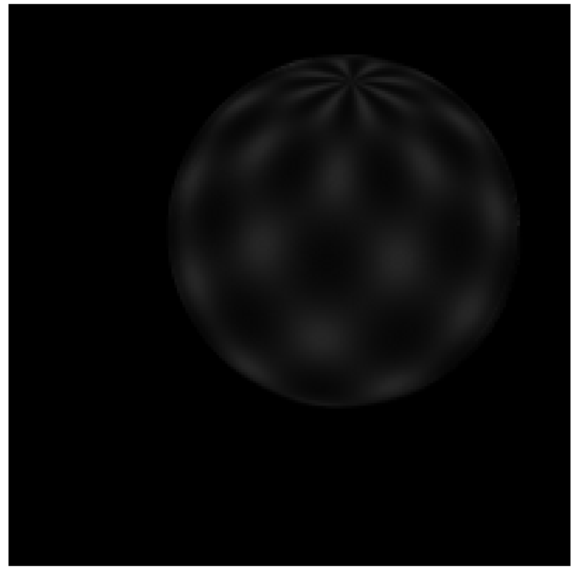
(b) Interpolated Frame 7

(c) Frame 9

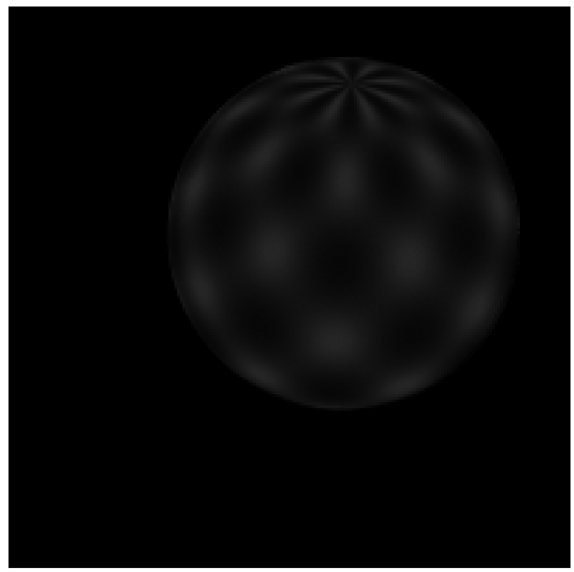
(d) Interpolated Frame 9

Figure 8: Fine Tuned Optical flow frame interpolation: Corridor


(a) Frame 1


(b) Interpolated Frame 1


(c) Frame 3


(d) Interpolated Frame 3


(e) Frame 5

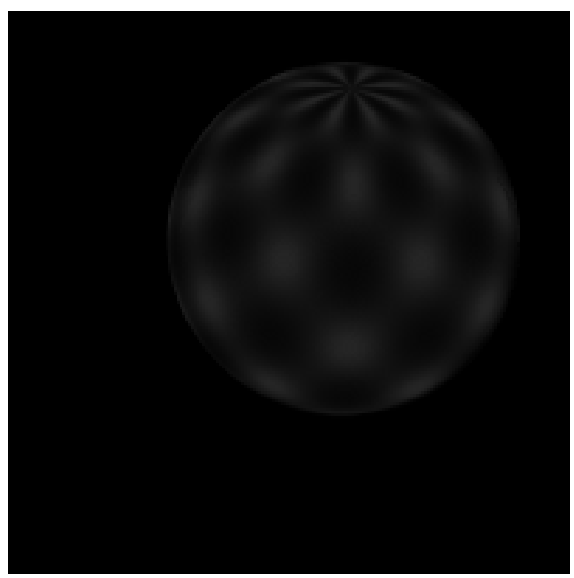
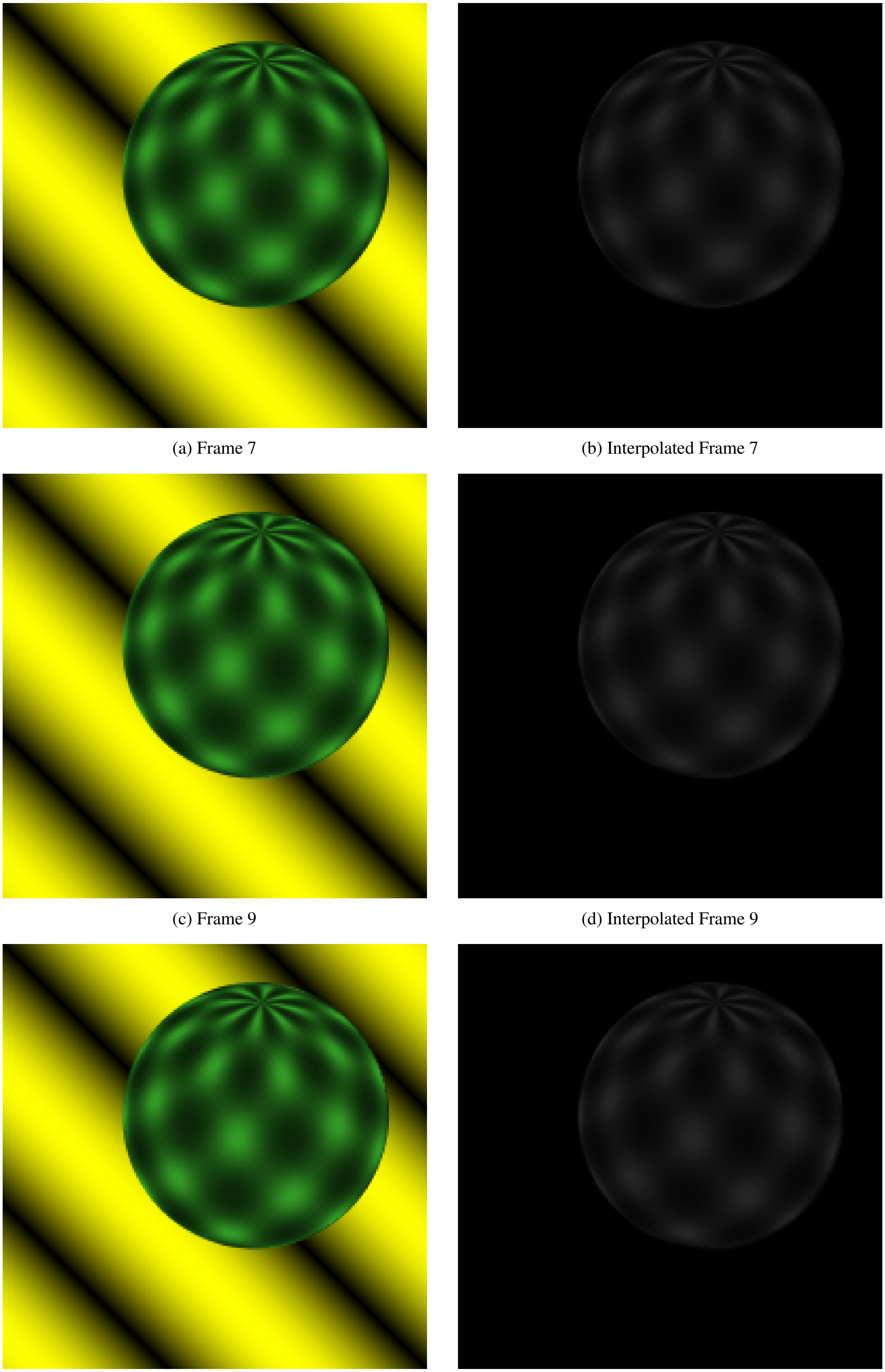

(f) Interpolated Frame 5

Figure 9: Fine Tuned Optical flow frame interpolation: Sphere



(a) Frame 7

(b) Interpolated Frame 7

(c) Frame 9

(d) Interpolated Frame 9

(e) Frame 11

(f) Interpolated Frame 11

Figure 10: Fine Tuned Optical flow frame interpolation: Sphere

Pixel Mapping

The interpolation method proposed in [1] was used in the above algorithms, with slight modifications. We use bivariate spline approximation (scipy library) to sample intensity from the frame at non-integer indices. To a certain extent, this also takes care of pixels not getting mapped into a particular location in a frame. But this methods tends to fail around the edges of the frame. We can identify the locations of such non-mapped pixels and replace it with the the average of pixels around it. We also identify occlusions in frames and sample only from the non occluded frame.

Comparison between Classical Flow Estimation and Deep Flow Estimation

We explore frame interpolation task for two the corridor and sphere datasets with optical flow. In the previous assignment, we could observe that classical optical flow estimation performed well on the sphere dataset and poor on the corridor dataset. In this assignment, we could observe that deep optical flow estimation performed well both on the sphere dataset and on the corridor dataset. There was a significant visual improvement of frame quality in using deep optical flow for the corridor dataset. However, there was no visual quality improvement with the sphere dataset using deep optical flow. Taking the above into consideration, we can conclude that deep optical flow estimation generally tends to perform better than classical optical flow estimation but need not always guarantee superior performance.

References

[1] S. Baker et al. "A Database and Evaluation Methodology for Optical Flow". In: *2007 IEEE 11th International Conference on Computer Vision*. 2007, pp. 1–8. DOI: `10.1109/ICCV.2007.4408903`.

[2] Eddy Ilg et al. *FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks*. 2016. arXiv: `1612.01925 [cs.CV]`.

[3] Fitsum Reda et al. *flownet2-pytorch: Pytorch implementation of FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks*. `https://github.com/NVIDIA/flownet2-pytorch`. 2017.

[4] Jason J. Yu, Adam W. Harley, and Konstantinos G. Derpanis. *Back to Basics: Unsupervised Learning of Optical Flow via Brightness Constancy and Motion Smoothness*. 2016. arXiv: `1608.05842 [cs.CV]`.