

VISION: Pratical work – implementation of two optical flow methods

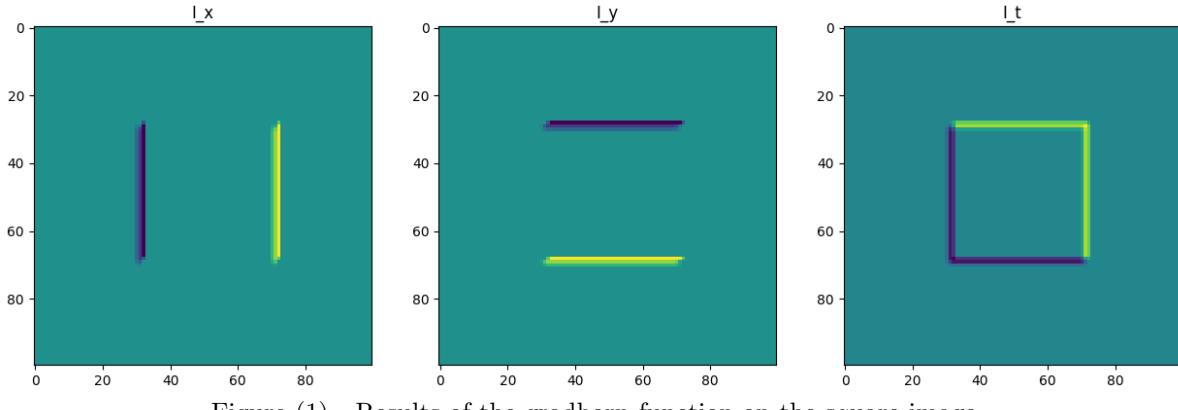
Victor PIRIOU
Nathan GALMICHE

The goals of this practical work consist of:

1. implementing the Horn-Schunck and Lucas-Kanade methods in order to compare them between each other;
2. testing them on datasets for which we have the groundtruth;
3. testing them on datasets for which we do not have the groundtruth.

1 Horn-Schunck method

1.1 Gradhorn test



The gradhorn function seems to work properly as we see on figure 1.

1.2 Test on the datasets for which we have the groundtruth

Since the use of the least square error minimization assumes that the data follow a Gaussian distribution, we decided to plot, for each dataset, the histogram of the mean errors in order to see how much this is the case.

1.2.1 Mysine dataset

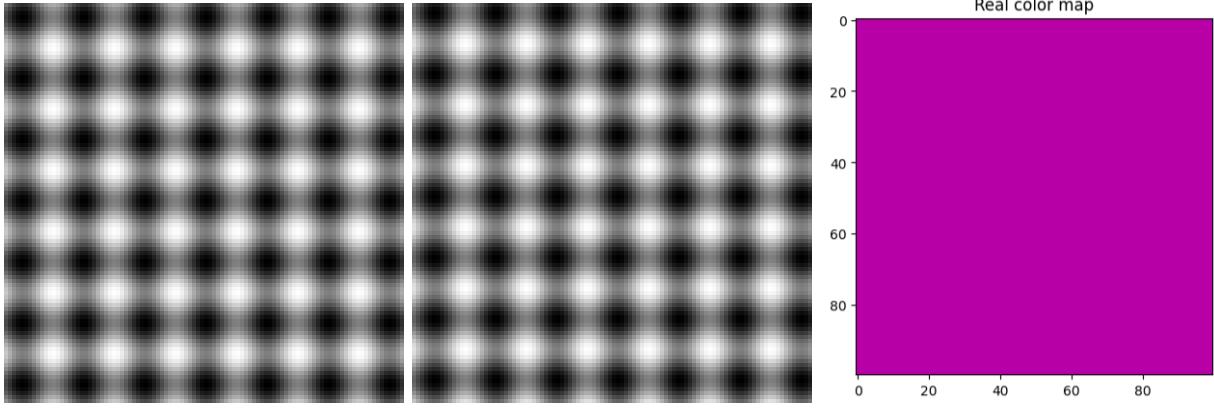


Figure (2) Mysine dataset and the ground-truth color-map.

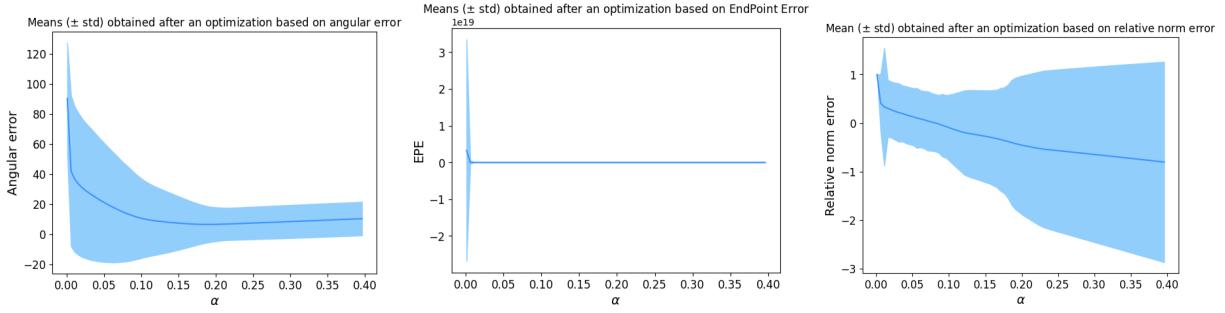


Figure (3) Mean and standard deviation of the errors obtained depending on the window size.

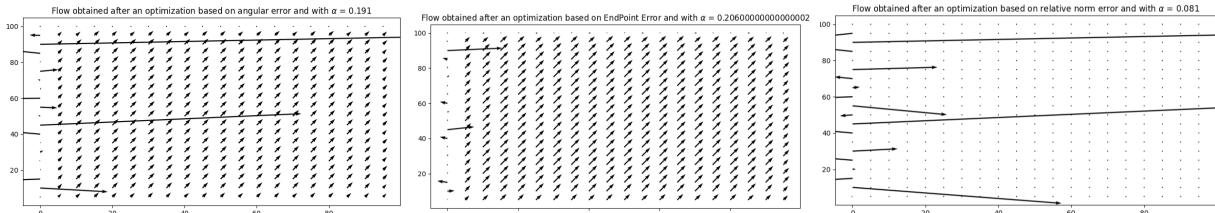


Figure (4) **Vector fields of the optical flow obtained with different optimizations.** We can notice some big outliers on the third figure which prevent us from properly seeing the other vectors that are nevertheless well estimated.

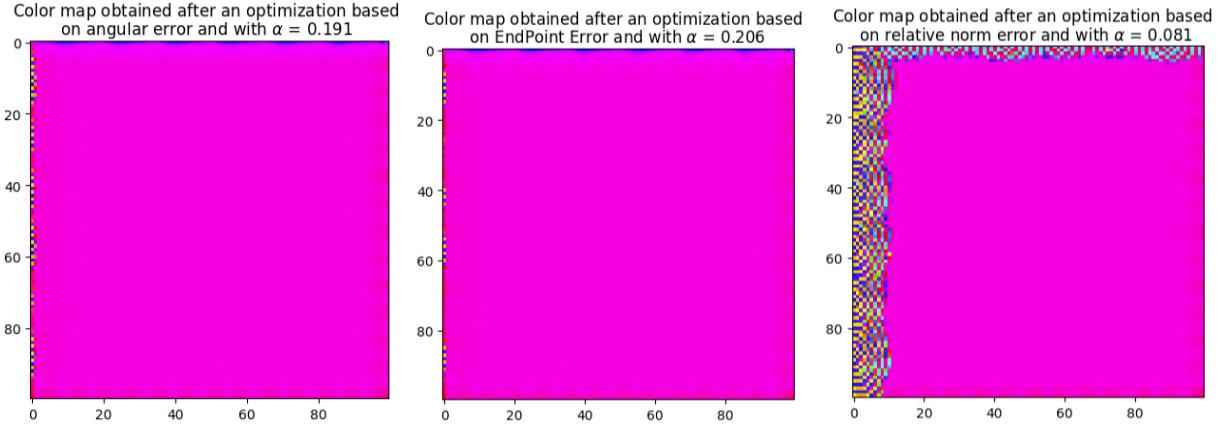


Figure (5) Color-maps obtained with different optimizations. With the relative norm error we obtain a bad result because the metric is not adapted for such optimizations.

1.2.2 Square dataset

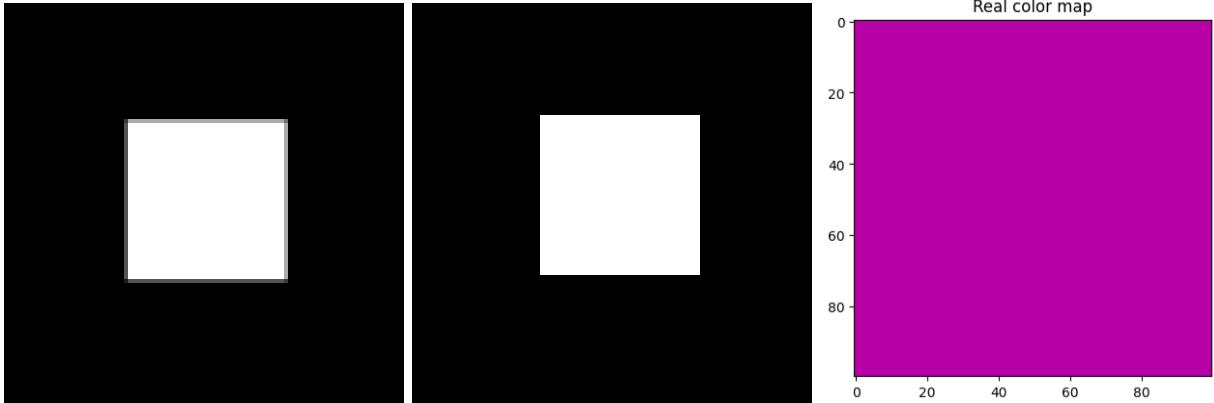


Figure (6) Square dataset and the ground-truth color-map.

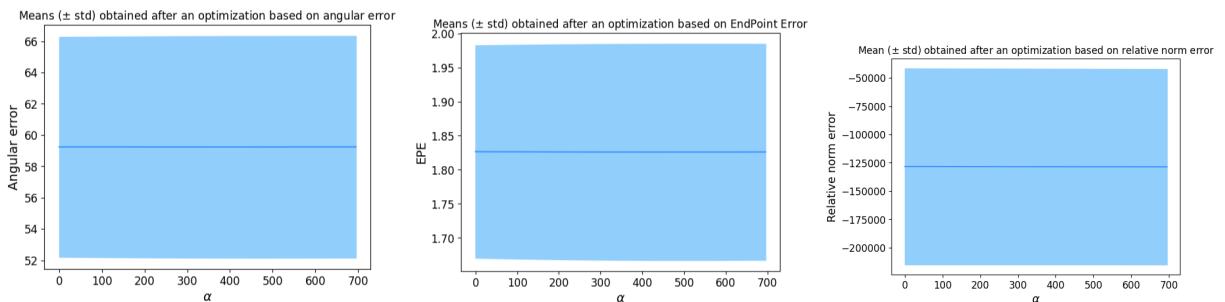


Figure (7) Mean and standard deviation of the errors obtained depending on the window size.

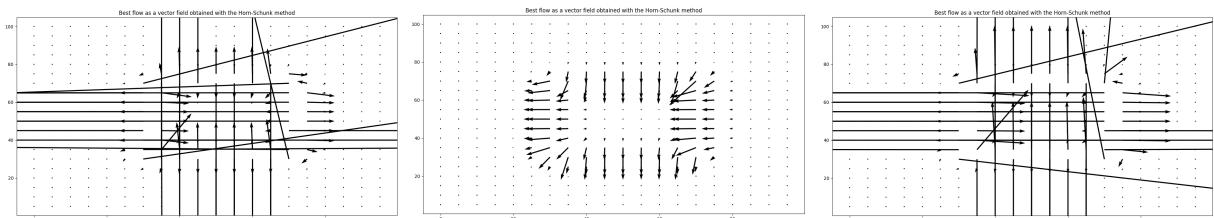


Figure (8) Vector fields of the optical flow obtained with different optimizations.

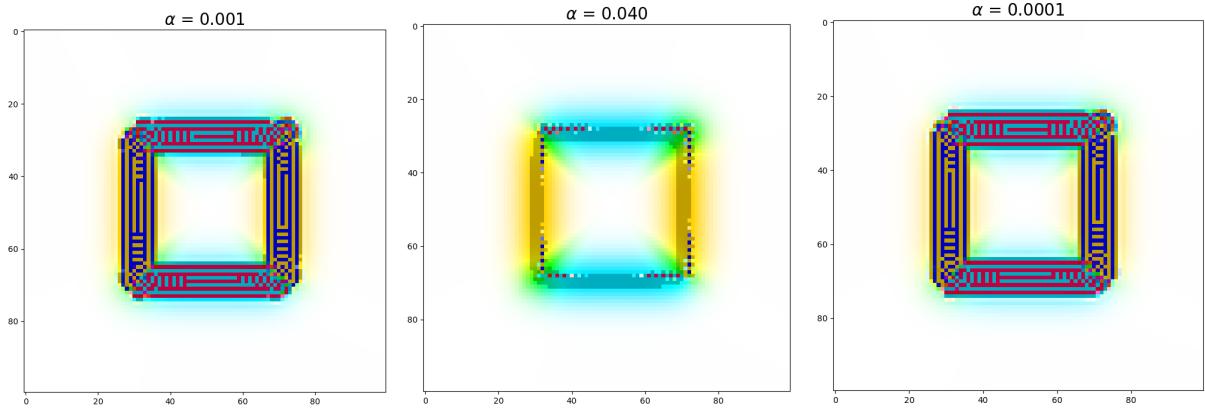


Figure (9) Color-maps obtained with different optimizations.

1.2.3 Rubberwhale dataset



Figure (10) Rubberwhale dataset and the ground-truth color-map.

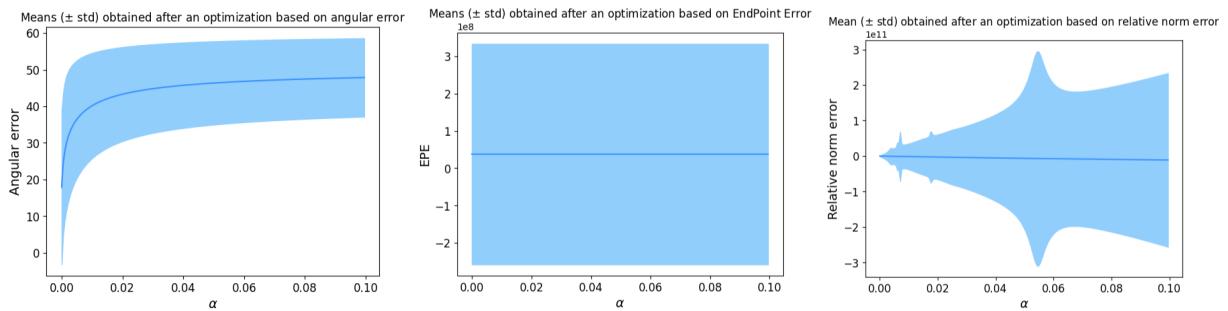


Figure (11) Mean and standard deviation of the errors obtained depending on the window size.

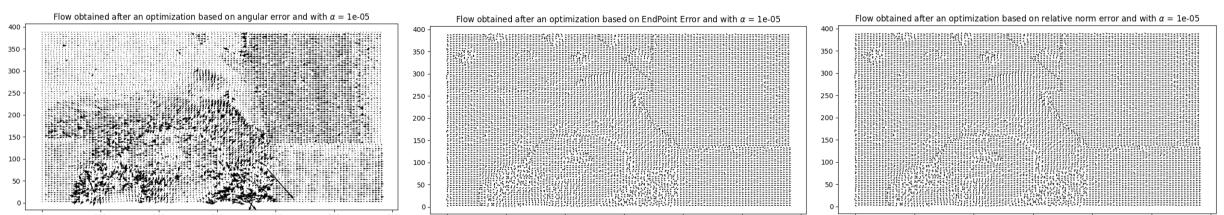


Figure (12) Vector fields of the optical flow obtained with different optimizations.

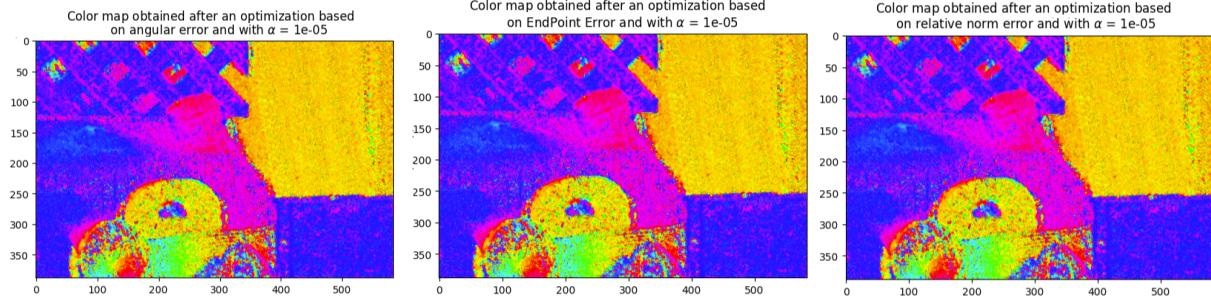


Figure (13) Color-maps obtained with different optimizations.

1.2.4 Yosemite dataset

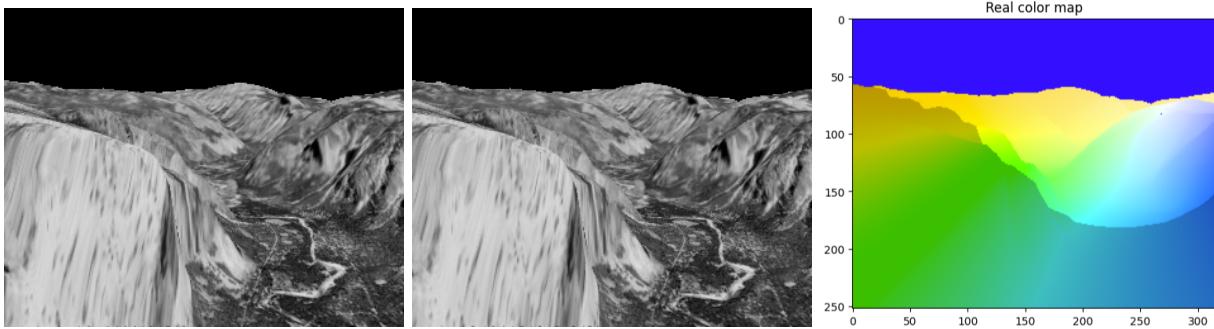


Figure (14) Yosemite dataset and the ground-truth color-map.

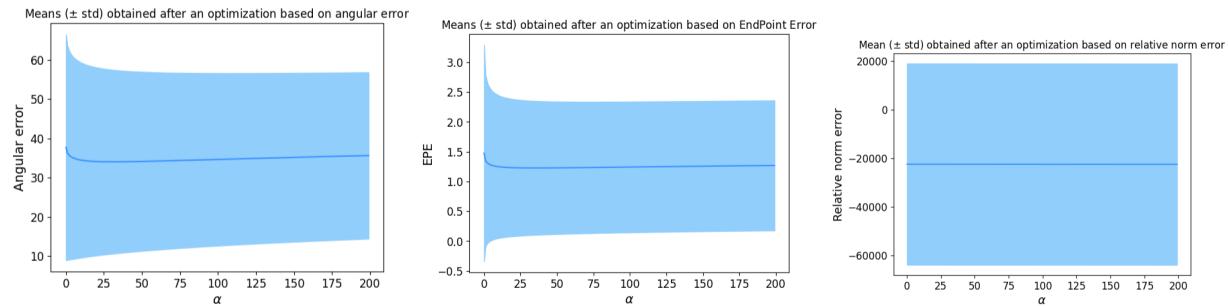


Figure (15) Mean and standard deviation of the errors obtained depending on the window size.

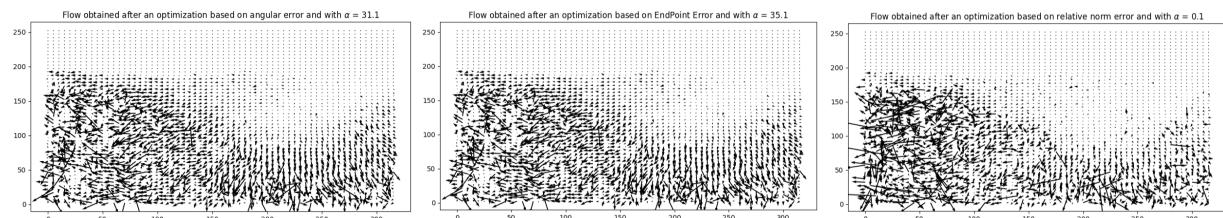


Figure (16) Vector fields of the optical flow obtained with different optimizations.

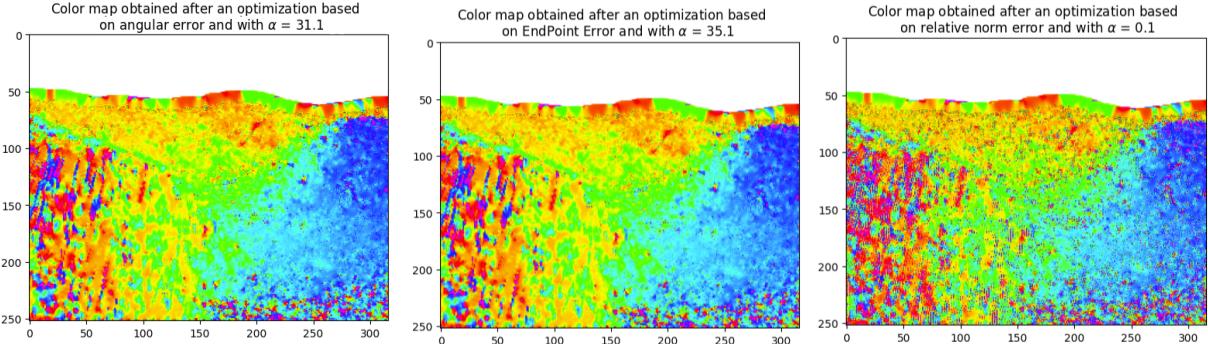


Figure (17) Color-maps obtained with different optimizations.

1.3 Test on the datasets for which we don't have the groundtruth

1.3.1 Nasa dataset

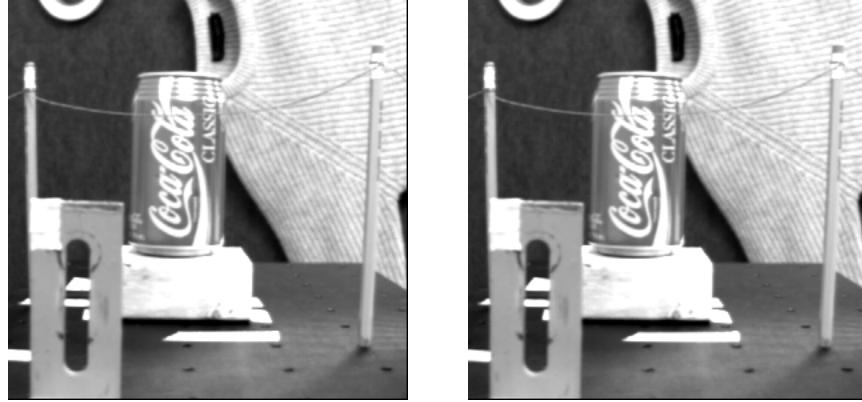


Figure (18) Images of the Nasa dataset.

This scene is made of different objects at different distances of the camera. The difference between the two images is that the second one is zoomed in. This means that the objects should appear a bit bigger and the ones that are not in the middle should be moving towards the outside of the image. Of course if it's a dezoom the opposite is possible. We expect the optical flow to be going from the center toward the outside or the opposite.

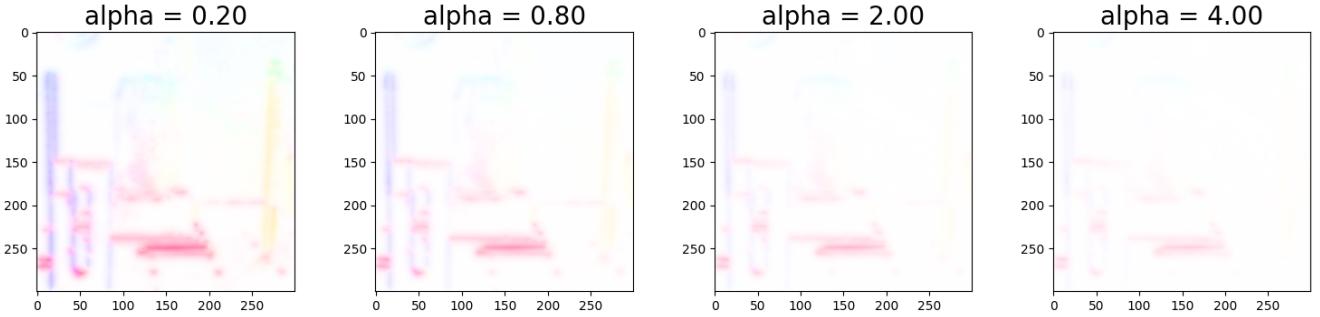


Figure (19) Color maps of the optical flows obtained with 50 iterations of the Horn and Schunk method. Here, α denotes the value of the regularization hyperparameter.

Looking at the color-maps, we first notice that iterating too much with a small regularization weight makes big flaws appear. This is because Horn-Schunck struggles where there is too much or too few information and when the regularization weight is too low, the fact that the displacement is found in random directions is not penalized enough. However to obtain good results we need to not increase the regularization weight too much. In fact, the bigger it is, the less the color-map is meaningful. As for the image, we expect there to be a focus point from which the objects are moving away. This means that the orientations of the displacements are all different around the

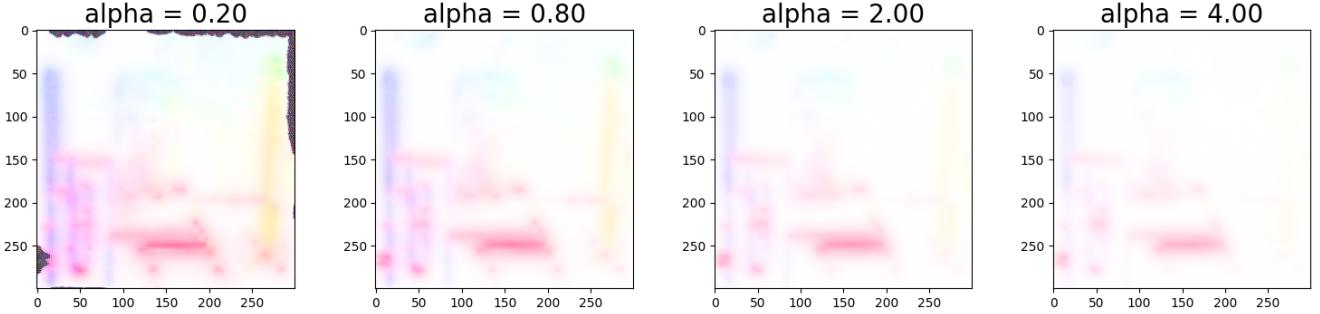


Figure (20) Color maps of the optical flows obtained with 200 iterations of the Horn and Schunk method.

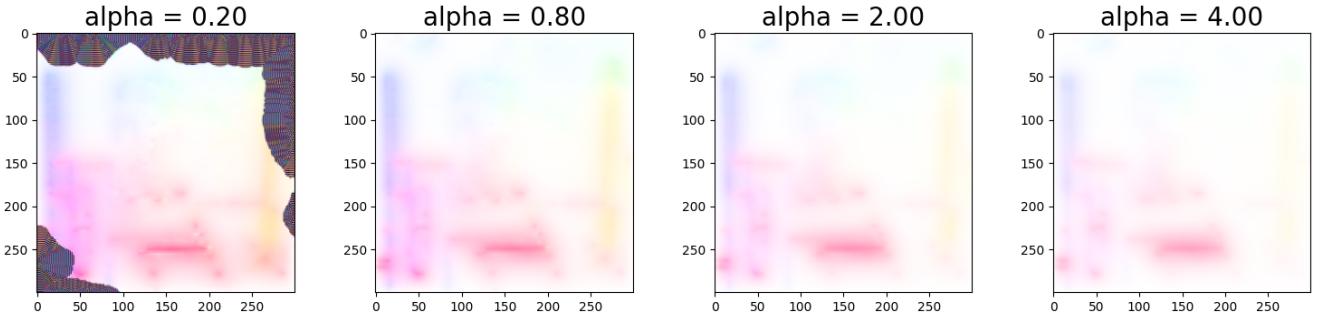


Figure (21) Color maps of the optical flows obtained with 500 iterations of the Horn and Schunk method.

image. We thus expect the colors to depend on where they are on the image. The result is quite satisfying since we have pink on the bottom left, turning to blue when we go up the image, turning to yellow when we shift to the right. With few iterations we seem to have a more precise flow but the colors are less intense so it is interesting to keep iterating some more so we do not miss any details. As mentioned before, we thus need to increase the regularization weight to avoid flaws and we obtain a pretty good color-map with 200 iterations and α set at 0.8.

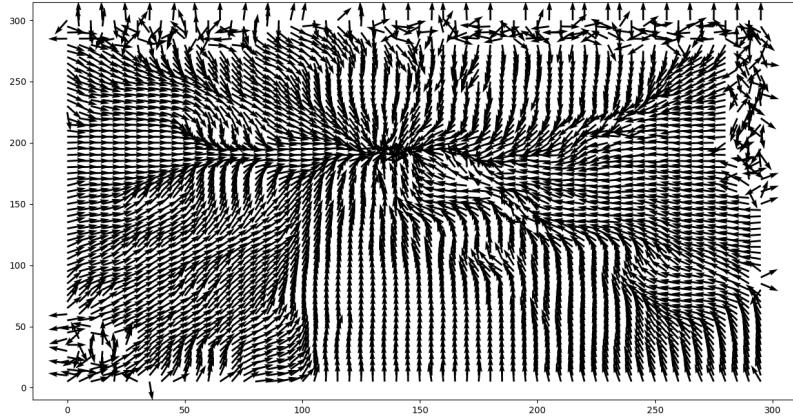


Figure (22) Best flow as a normalized vector field obtained with the Horn-Schunck method.

We show a vector field of this optical flow to check that the results make sense. As we expected we notice that the vectors are all oriented toward a single point in the middle of the image. This is the focus point. We notice slight differences in the vectors' orientations in certain zones which can be explained by the distance between the object and the camera.

1.3.2 Rubic dataset

This scene is a rubic's cube sitting on a round trey. Between the second image and the first, the trey turns a little. The patterns on the side of the trey should allow to track intensity changes, and thus the movement, as well as the corners of the cube. The rest of the image does not move.

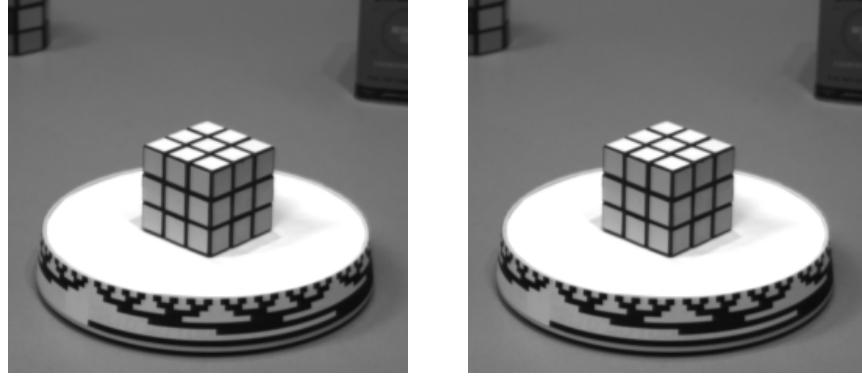


Figure (23) Images of the Rubic dataset.

Again, we notice the same phenomenons mentioned with the previous dataset. We might add that the flaw phenomenon appears mainly on the borders of the image as well as on the edges.

We obtain gradients of colors on the color-maps which was expected since the objects are turning, meaning that the direction of the movement shifts along the object. It is nice to note that the intensity of the colors are way bigger for the trey than the cube. This is good because the objects revolve around the center of the tray, where the cube sits. This means that the sides of the tray are farther from the center than the rubic's cube and the displacement is thus bigger.

As we expected, the rest of the image does not impact the color-map. However the top of the trey does not either, even though it is moving. This is because the region is plain and Horn-Schunck struggles to use the various directions around the zone to evaluate the flow.

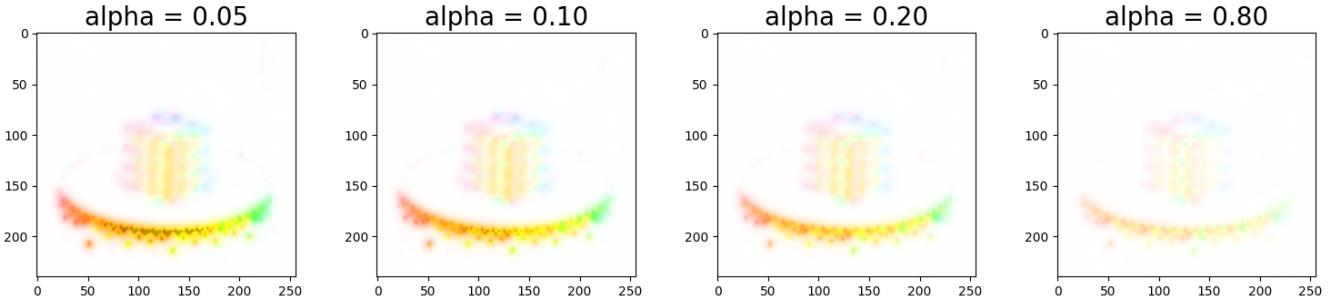


Figure (24) Color maps of the optical flows obtained with 50 iterations of the Horn and Schunk method.

By checking with vector fields, we find that the global movement corresponds to a turning object, with vectors more important when it is farther from the center.

If we normalize the vectors, we are able to see the ones around the moving objects and we can confirm that it is not interesting to keep them since they seem random.

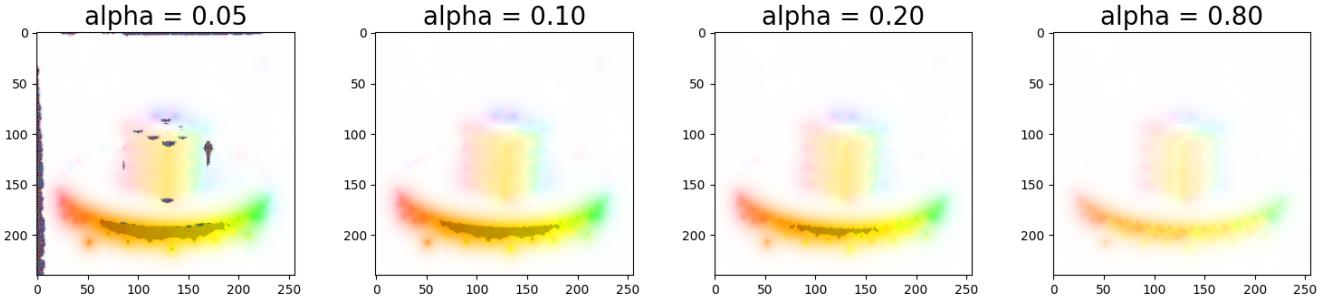


Figure (25) Color maps of the optical flows obtained with 200 iterations of the Horn and Schunk method.

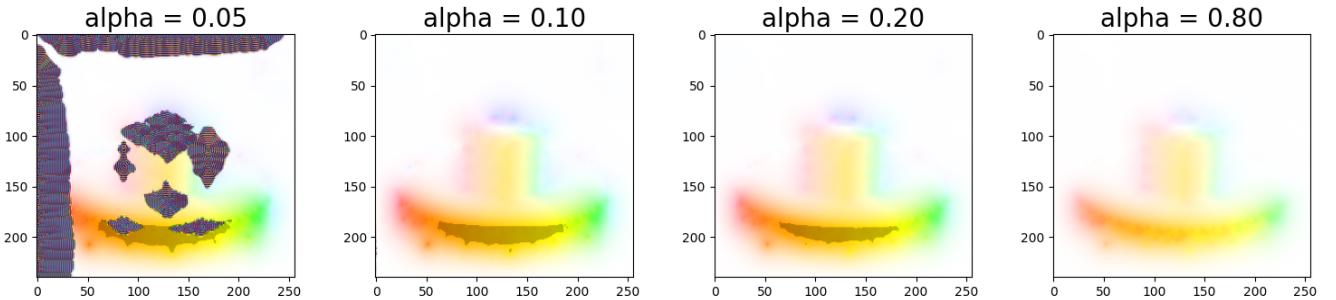


Figure (26) Color maps of the optical flows obtained with 500 iterations of the Horn and Schunk method.

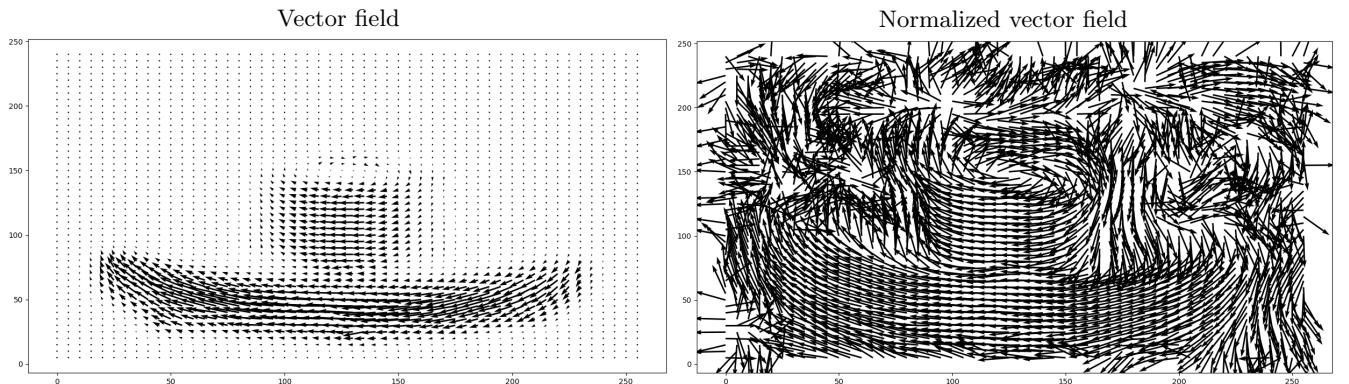


Figure (27) Best flow as vector fields obtained with the Horn-Schunck method.

1.3.3 Taxi dataset

This scene is a street where there are parked cars as well as moving cars. Between the first and second image, the cars on the left, middle and right each move in different directions. The rest of the scene does not move.



Figure (28) Taxi dataset.

We first see on the color maps that some movement is found around the moving objects. This is probably due to noise and we will mention it again in the next part.

We notice that the two cars that are moving towards the left are well recognized and clearly appear on the color-map. However, the third car moving towards the right is less visible. This is most likely due to its color which is black. It makes the car plainer and we see that the method struggles to catch on these zones, even more so when the regularization weight is high.

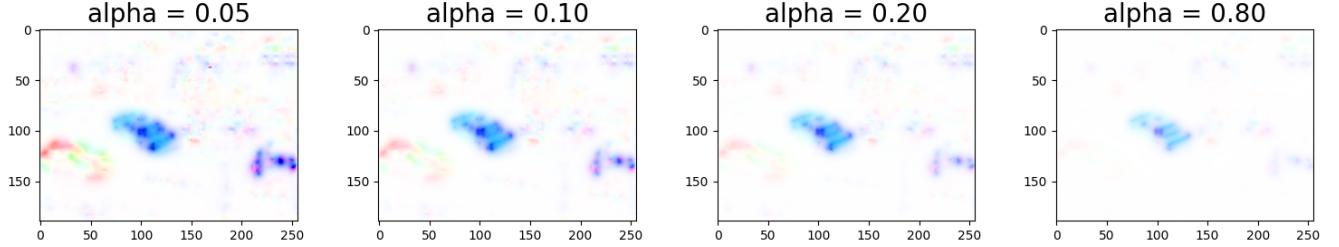


Figure (29) Color maps of the optical flows obtained with 50 iterations of the Horn and Schunk method.

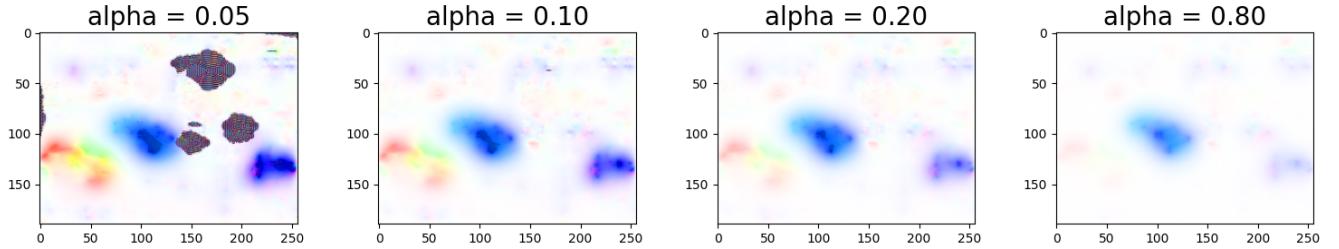
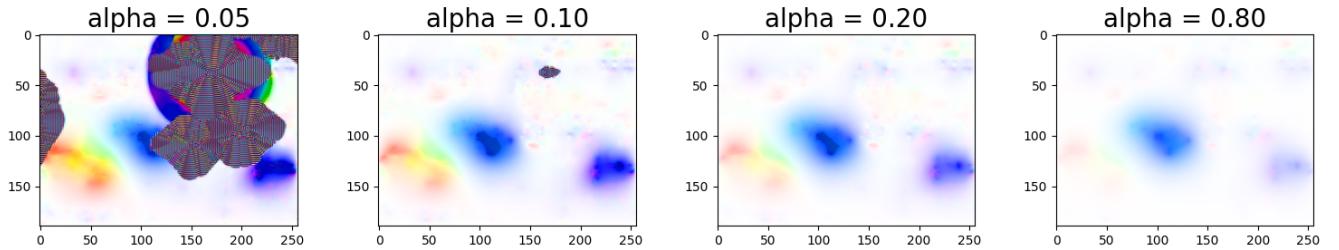


Figure (30) Color maps of the optical flows obtained with 300 iterations of the Horn and Schunk method.



The vector field confirms this as we see vector patches on the three cars but the left one as way less and they are way smaller. We notice again the results of the noise in other places of the image.

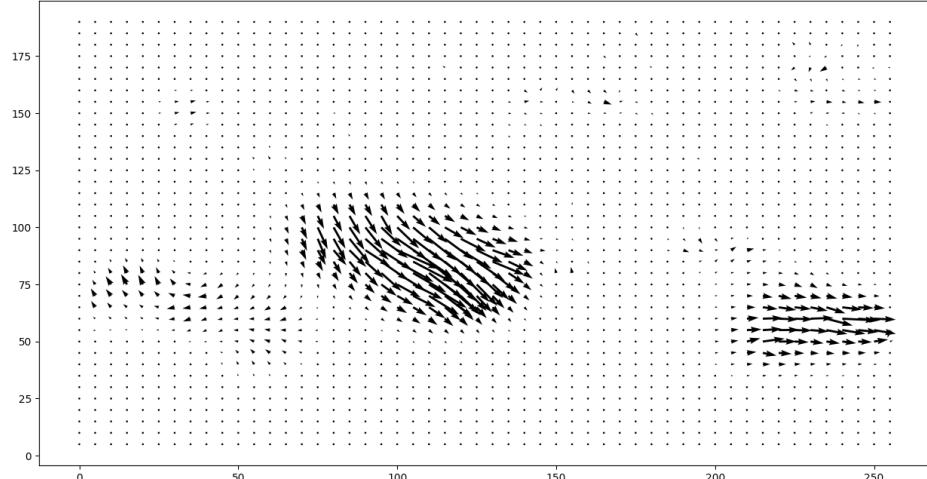


Figure (31) Best flow as a vector field obtained with the Horn-Schunck method.

2 Lucas-Kanade method

2.1 Tests on the datasets for which have the ground-truth

2.1.1 Mysine dataset

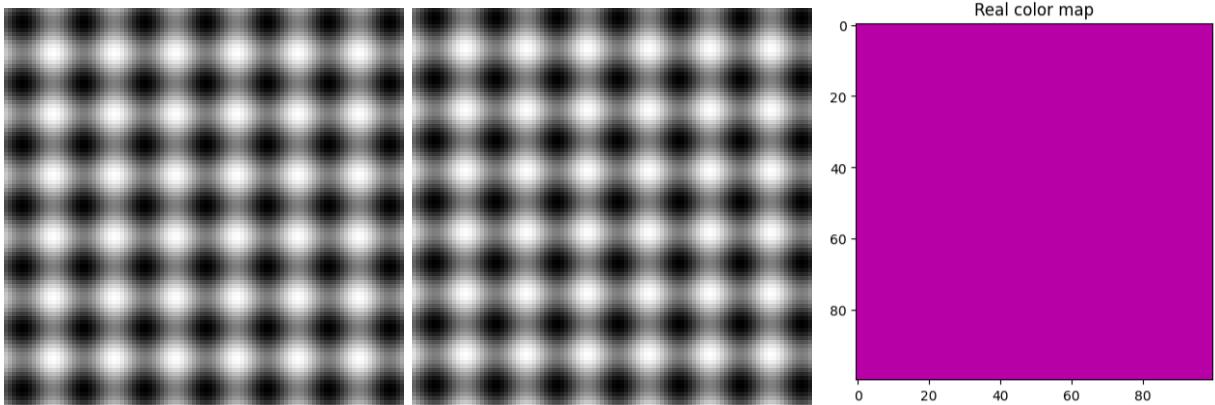


Figure (32) Mysine dataset and the ground-truth color-map.

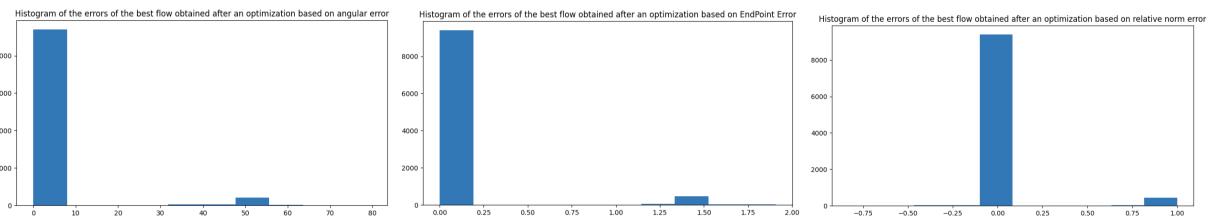


Figure (33) Histograms of the errors obtained after different optimizations.

We can notice that the errors are minimized with a very small window. This fact was expected because these are textured images so a big window would amplify the filtering effect.

We can also see that the flow vectors of the first row and column are missing. Again, this was expected given the fact that the real movement goes towards the top-right corner of the image.

Despite the fact that the errors distribution does not follow a Gaussian distribution, the results are excellent.

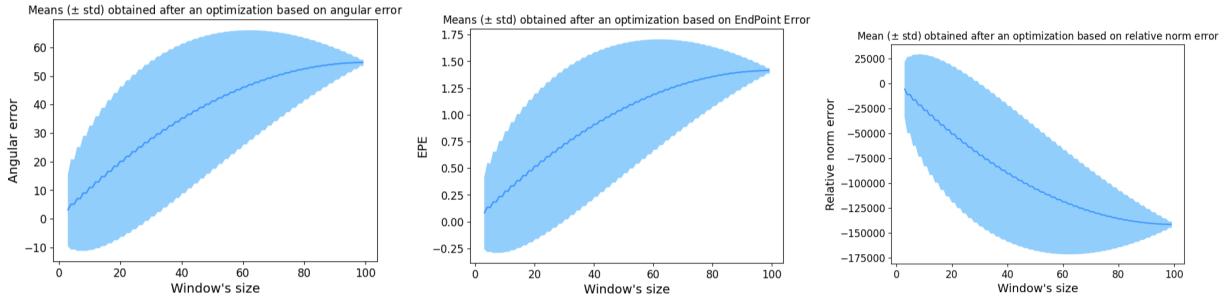


Figure (34) **Mean and standard deviation of the errors obtained depending on the window size.** The very high values of the relative norm errors are due to the fact that absolute norm errors is amplified by a small velocity.

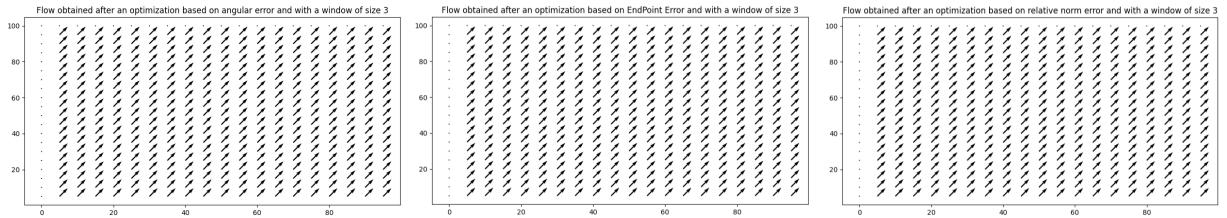


Figure (35) Vector fields of the optical flow obtained with different optimizations.

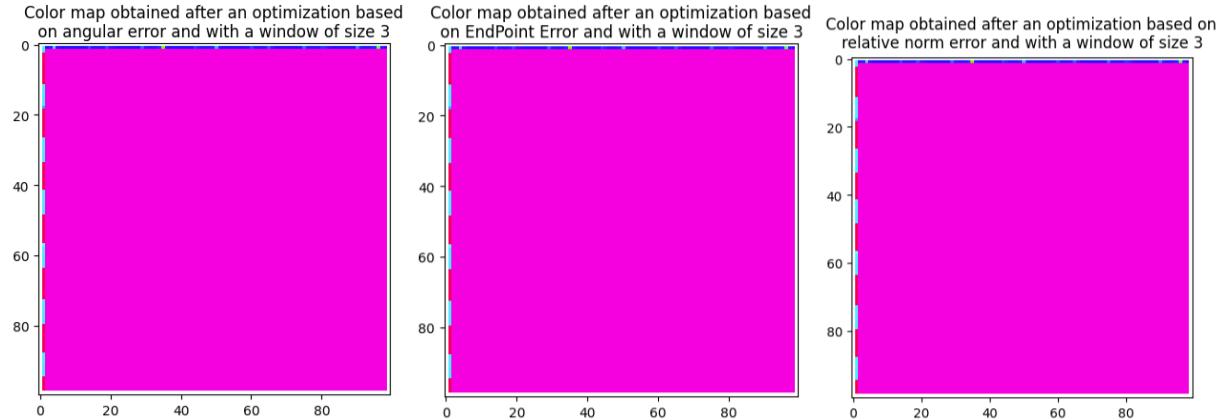


Figure (36) Color-maps obtained with different optimizations.

2.1.2 Square dataset

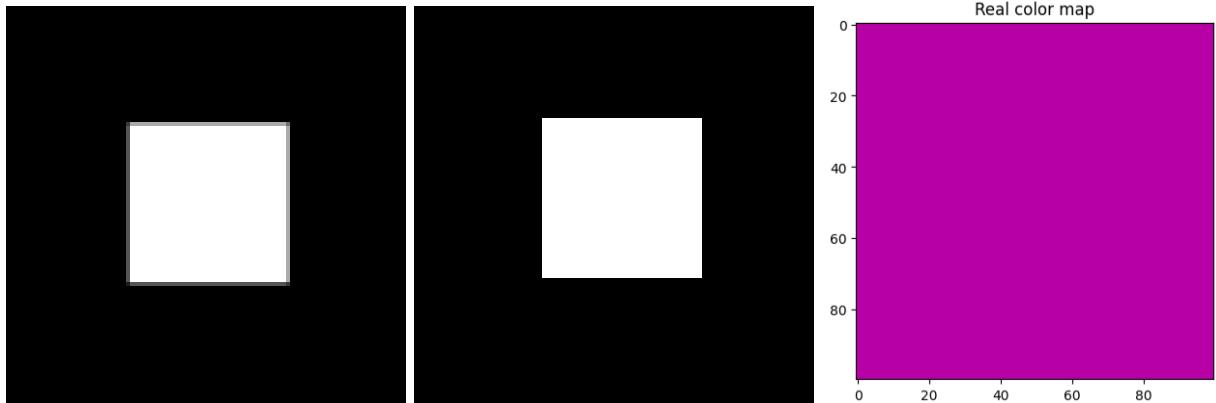


Figure (37) Square dataset and the ground-truth color-map.

Here, we can notice that the errors are minimized with a window whose size is approximately equal to the size

of the square.

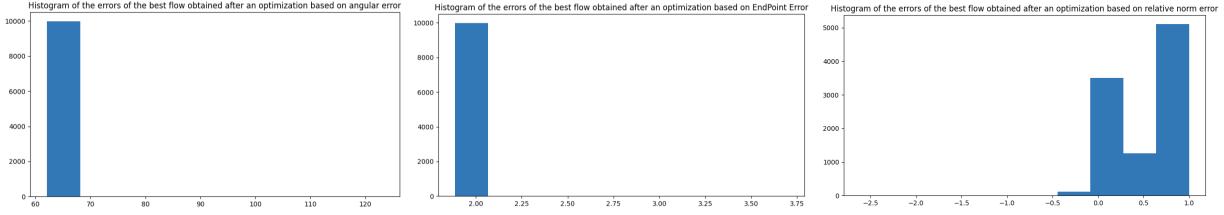


Figure (38) Histograms of the errors obtained after different optimizations.

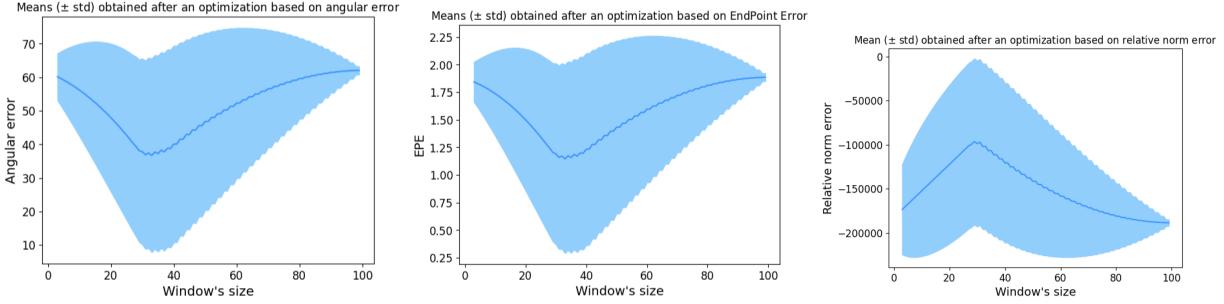


Figure (39) Mean and standard deviation of the errors obtained depending on the window size.

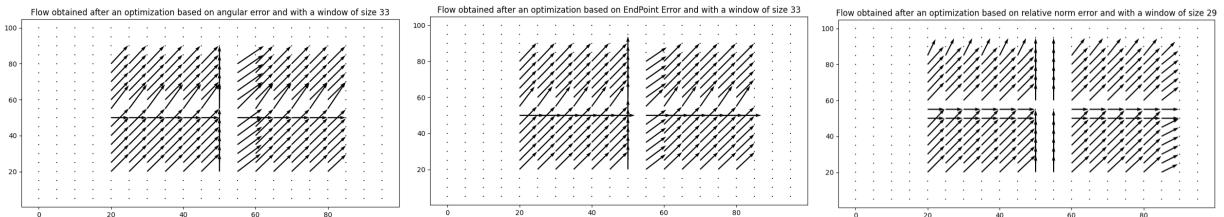


Figure (40) Vector fields of the optical flow obtained with different optimizations.

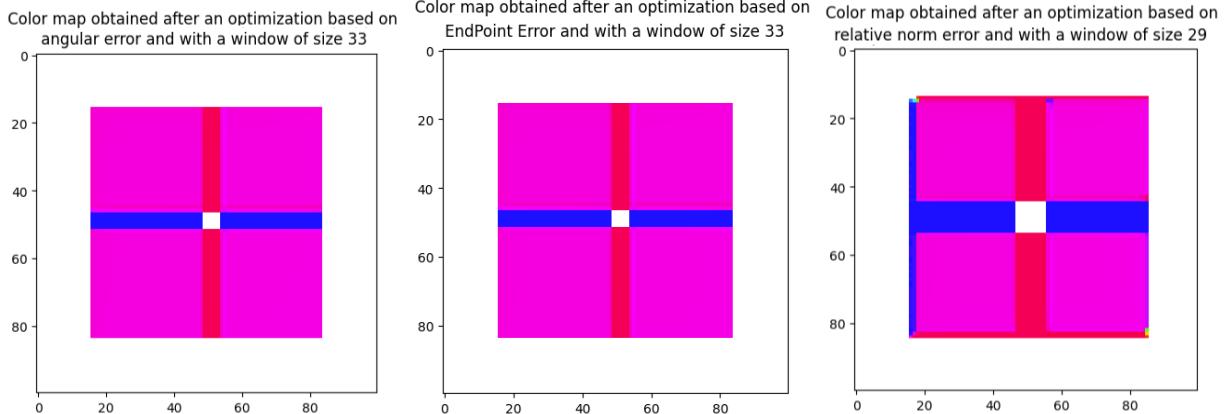


Figure (41) Color-maps obtained with different optimizations.

2.1.3 Rubberwhale dataset



Figure (42) Rubberwhale dataset and the ground-truth color-map.

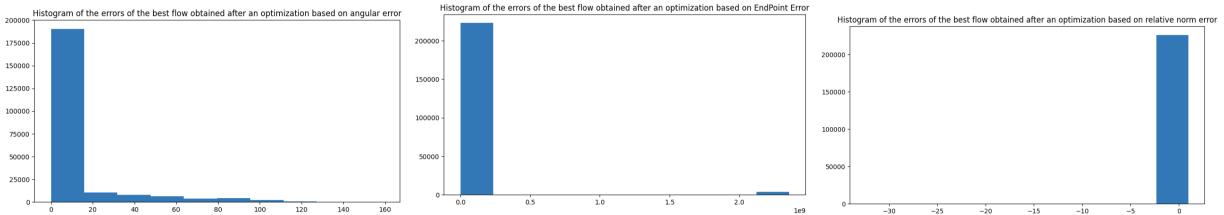


Figure (43) Histograms of the errors obtained after different optimizations.

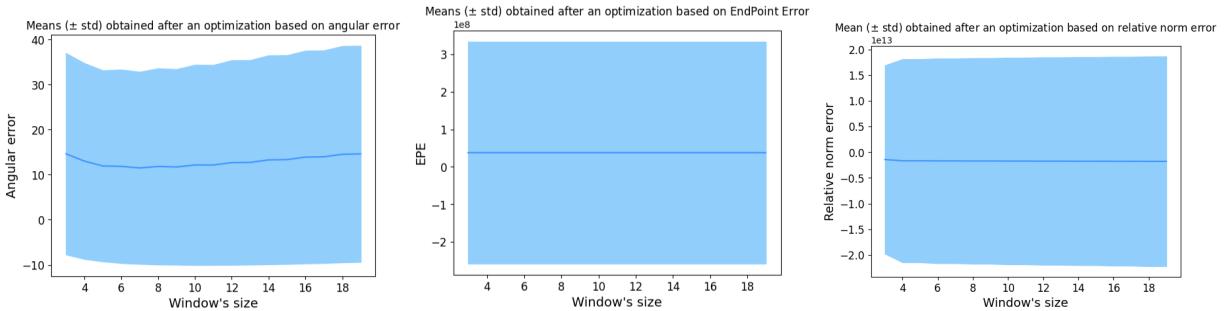


Figure (44) Mean and standard deviation of the errors obtained depending on the window size.

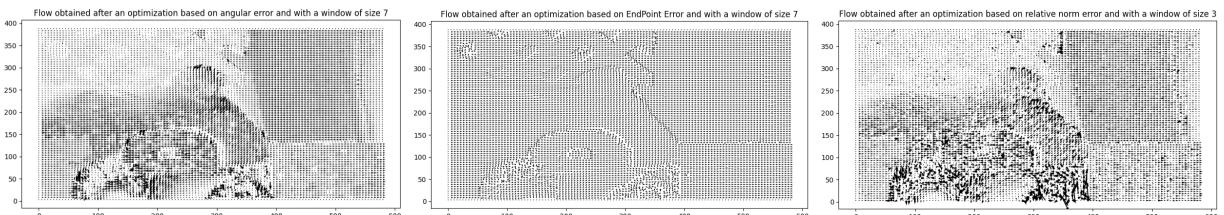


Figure (45) Vector fields of the optical flow obtained with different optimizations.

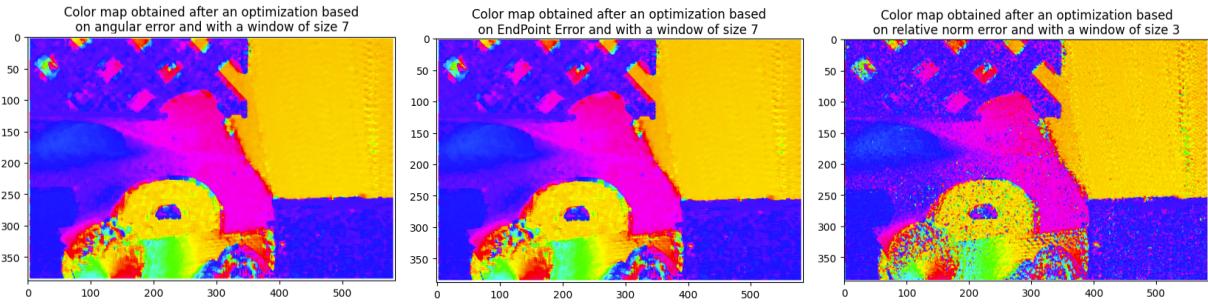


Figure (46) Color-maps obtained with different optimizations.

2.1.4 Yosemite dataset

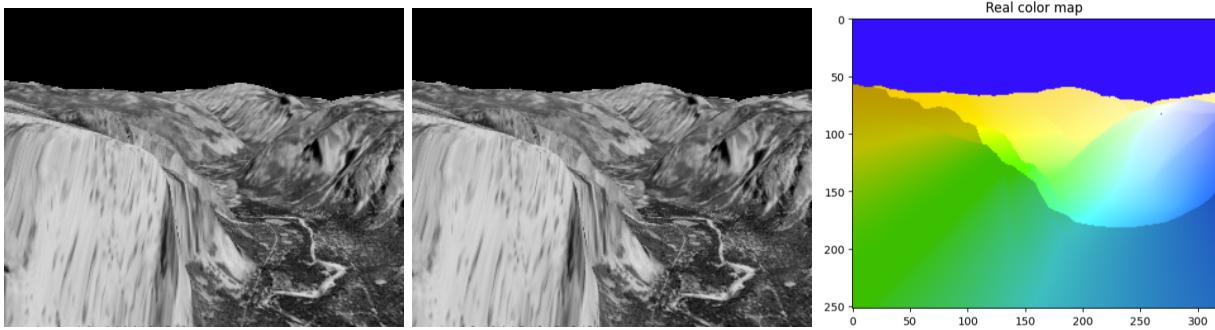


Figure (47) Yosemite dataset and the ground-truth color-map.

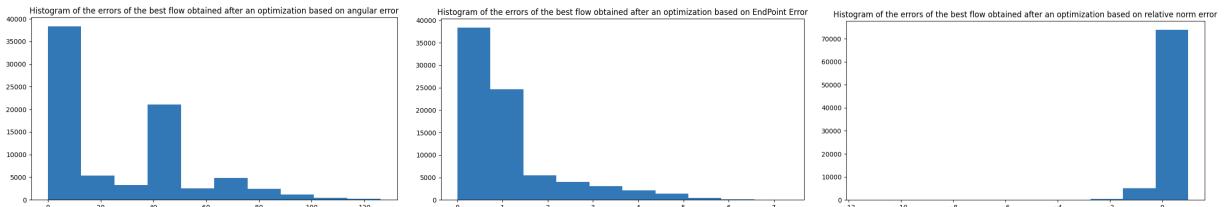


Figure (48) Histograms of the errors obtained after different optimizations.

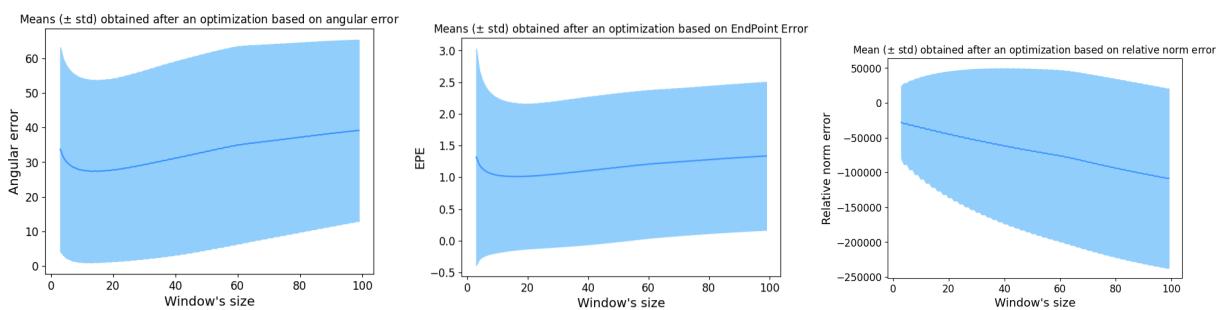


Figure (49) Mean and standard deviation of the errors obtained depending on the window size.

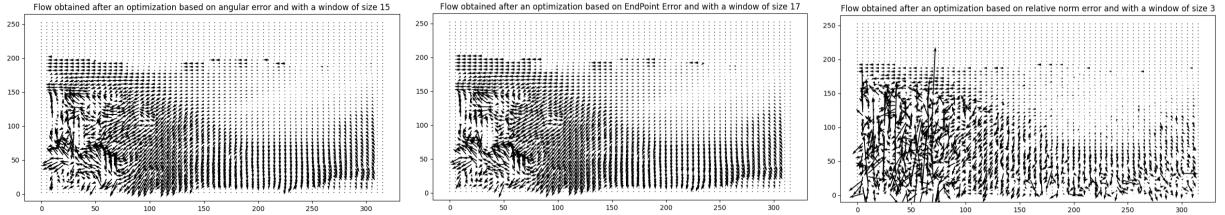


Figure (50) Vector fields of the optical flow obtained with different optimizations.

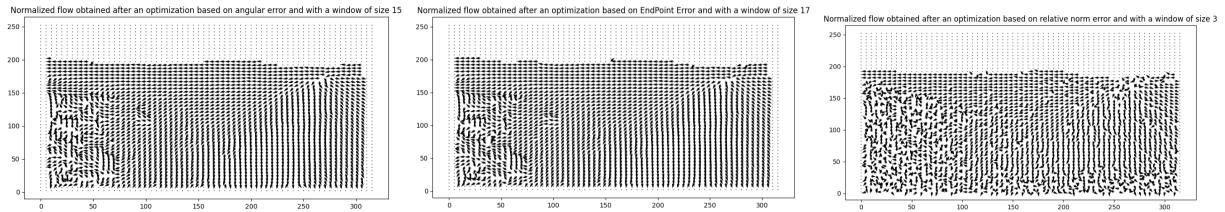


Figure (51) Normalized vector fields of the optical flow obtained with different optimizations.

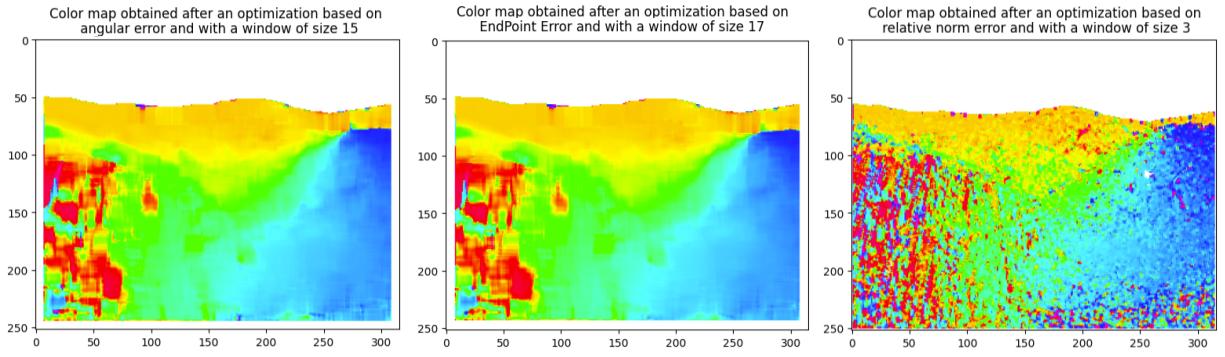


Figure (52) Color-maps obtained with different optimizations.

2.2 Test on the datasets for which we don't have the groundtruth

2.2.1 Nasa dataset

Compared to the Horn-Schunck method, the Lucas-Kanade method fills the color-map better. It is even easier to notice the gradient of color that depends on where we are on the image.

There is one flaw in the middle of the image. It most likely corresponds to the bottom of the can where we have a shadow. In fact Lucas-Kanade can struggle with edges that have a certain orientation. It is possible that it did not correctly compute the color because of this. We also notice that the Gausian window allows to smooth the map more. We obtain more details when using it.

The vector field looks pretty similar to the one we obtained with Horn-Schunck, the results are as we expected.

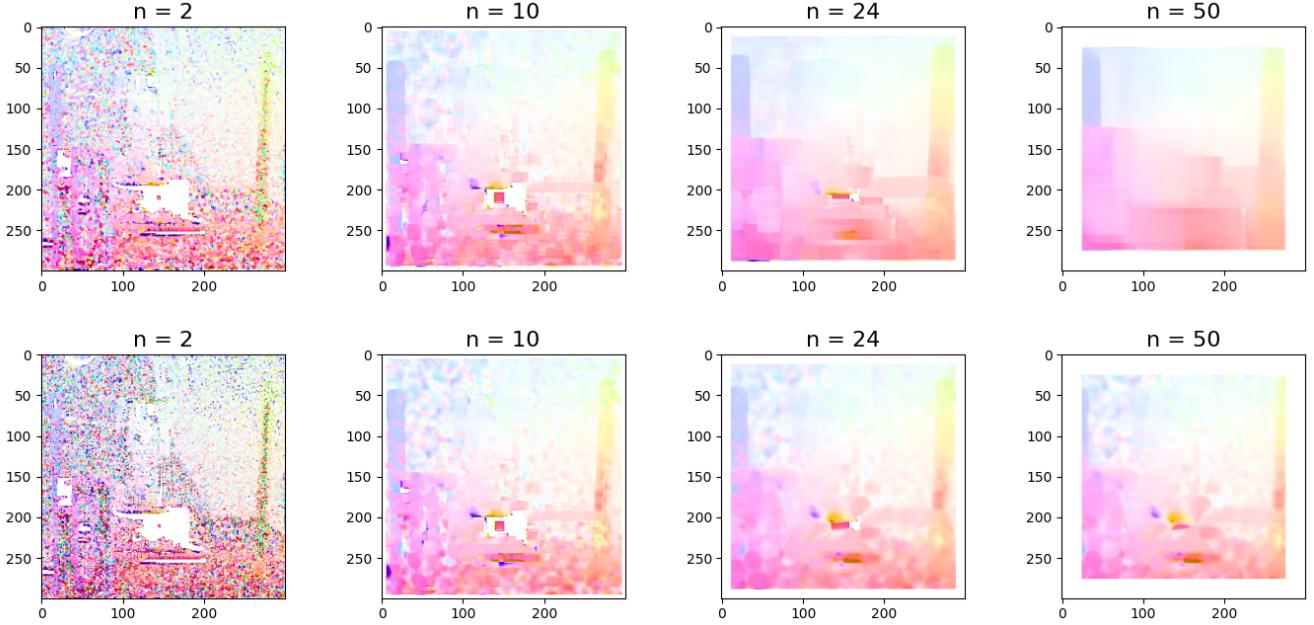


Figure (53) **Optical flow obtained with the Lucas-Kanade method. The second line uses a Gaussian kernel.** Here, n denotes the window's size.

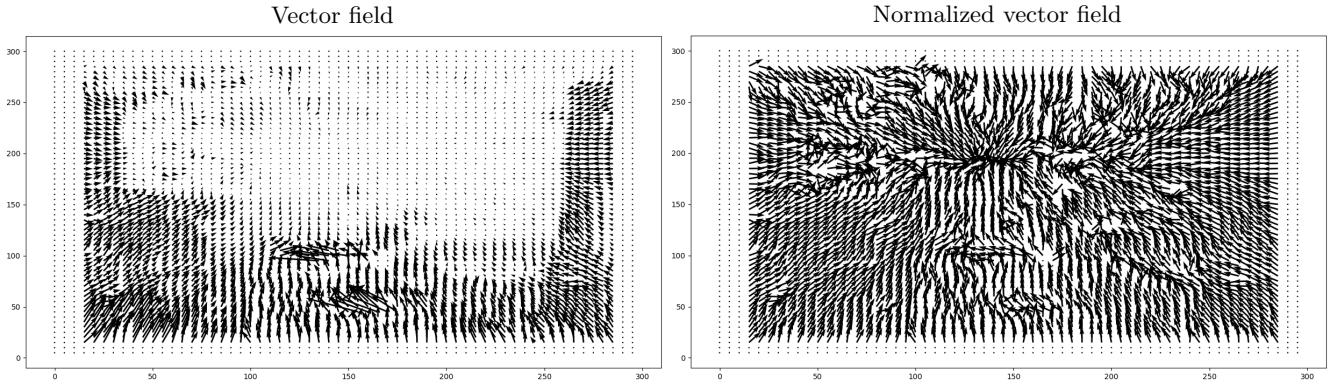


Figure (54) Best flow as vector fields obtained with the Lucas-Kanade method.

2.2.2 Rubic dataset

On this dataset, we see that Lucas-Kanade is pretty sensible to noise. In fact, the areas around the objects should appear white instead of colored like it is here. This means that the method mistook noise for displacement. It is not the case on the tray which is a lot lighter so the impact of the noise is lessened. Using bigger windows and/or Gaussian ones can help diminish the impact of the noise.

However, it is pretty effective as we obtain the expected result. We can even differentiate the squares of the rubic's cube when the window size is small enough.

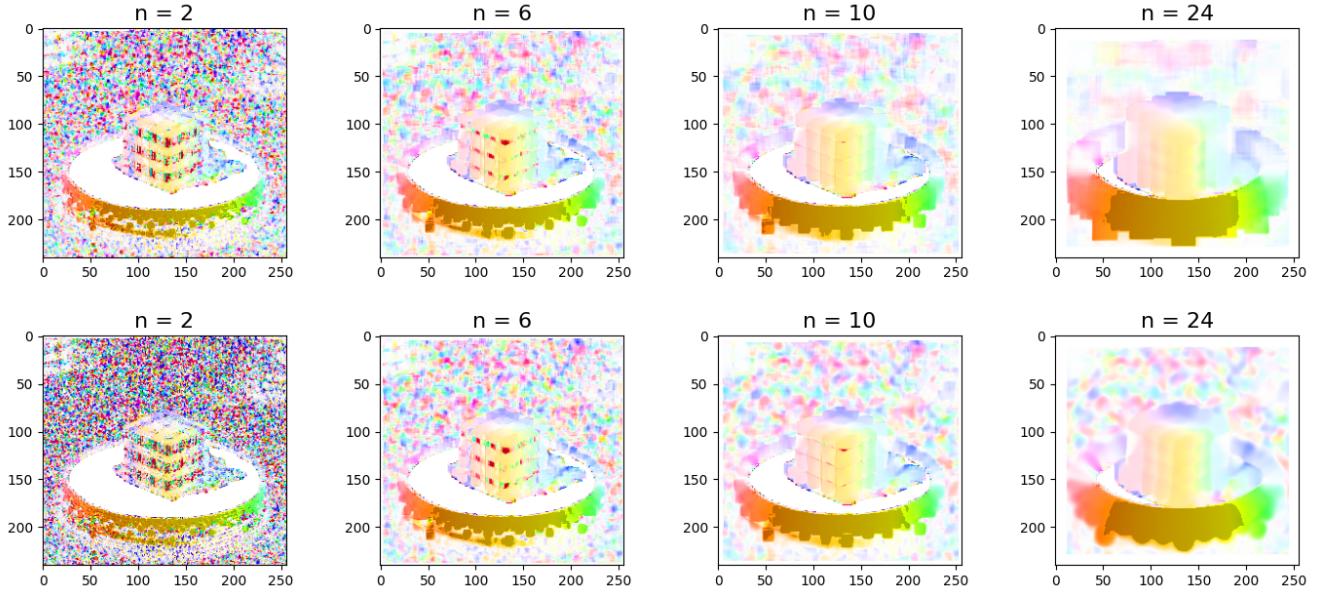


Figure (55) Color maps of the optical flows obtained with the Lucas-Kanade method.

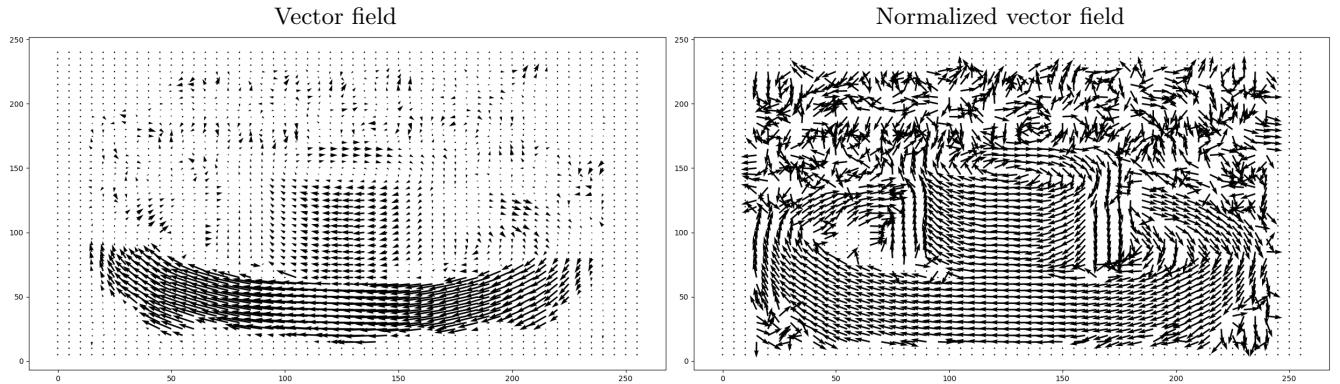


Figure (56) Best flow as vector fields obtained with the Lucas-Kanade method.

Again, we find on the vector field the noise that we had on the color-map but we could argue that there are more correct vectors on this map than with Horn-Schunck. When we look at the normalized vector, it is very obvious that we have objects turning.

2.2.3 Taxi dataset

Again, we find quite a lot of noise around the moving objects. However this time, the car on the left is very visible on the color-map. This is allowed by the fact that there is less regularization and because the windows allow to focus on precise parts and the small intensity changes on the car are more taken into account.

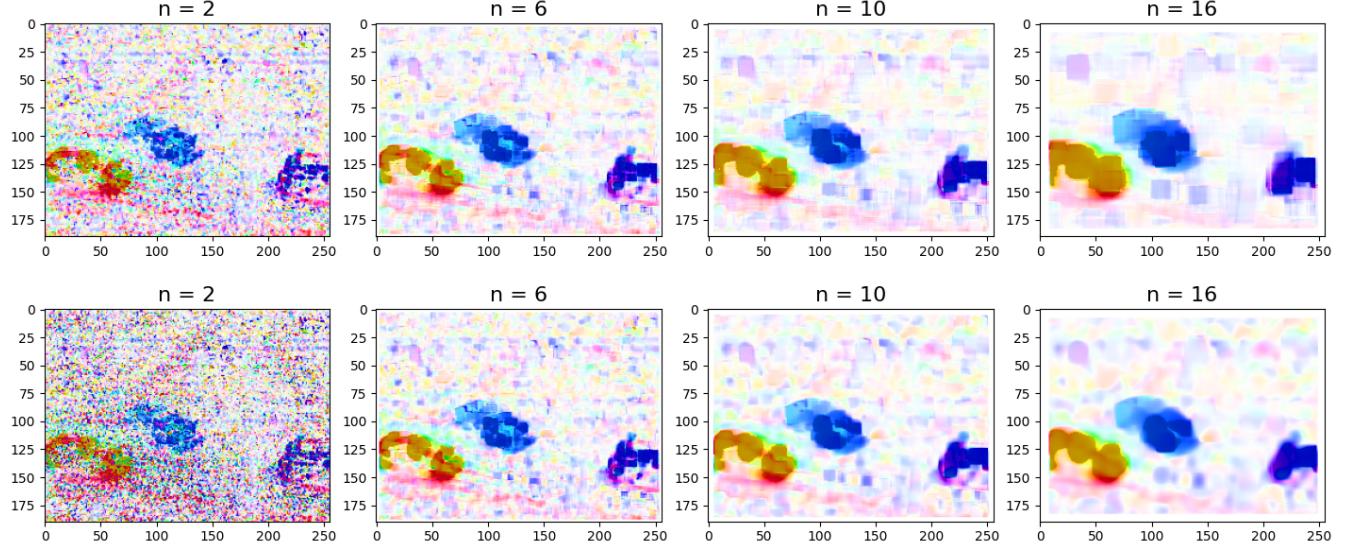


Figure (57) Color maps of the optical flows obtained with the Lucas-Kanade method.

Finally we have a good vector field with three clear patches for each cars and the right directions.

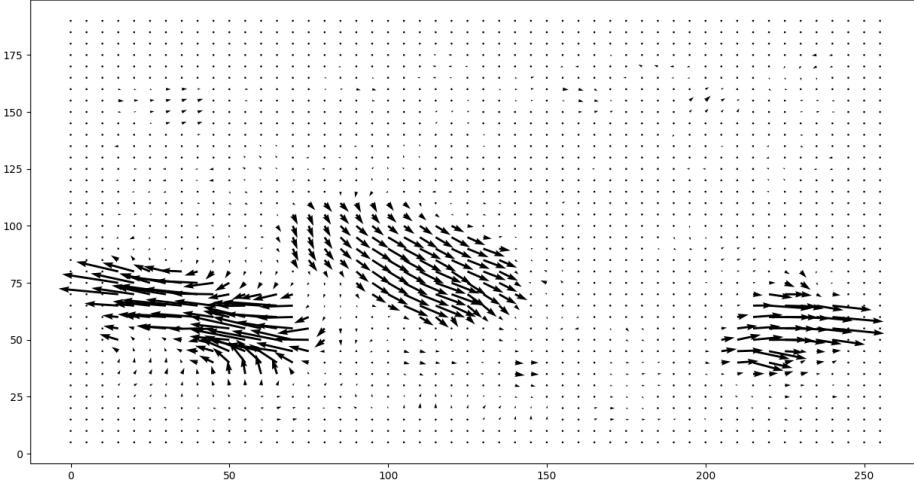


Figure (58) Best flow as vector fields obtained with the Lucas-Kanade method.

3 Conclusion

In this practical work we compared two methods to determine the optical flow. On the one hand, we have a slow method that uses regularization to smooth the displacement and evaluate it in plain zones of the images. On the other hand, we have a fast method that uses local windows to evaluate the displacement locally. From there we have to choose the method depending on our needs. There is no better method since the regularization is useful in some cases and hurtful in others.