Optical flow

Urša Zrimšek

I. Introduction

The main goal of this project is solving the problem of optical flow estimation. This is the first fundamental step in most video processing algorithms. We approached it by implementing two well known methods: Lucas-Kanade, that is solving the problem by least squares and Horn-Schunck, that is using variational calculus. In this report we will show their comparison graphically, testing them on two consecutive images from videos.

II. Experiments

When we implemented both methods, we first tested them on a rotation of random noise image - shown on Figure 1.

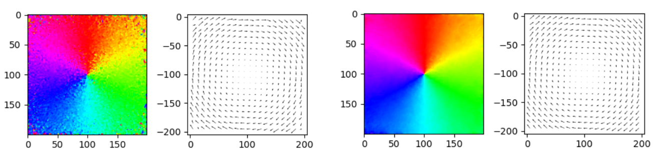

Figure 1. **Comparison of Lucas-Kanade (left) and Horn-Schunck (right) method**. In both field images we can see that methods show rotation in right direction, correctly showing smaller field in the center, where the points are less affected by rotation, and bigger field near the edges, where the points move more. From the color image that shows angle, we can see that the Horn-Schunck method is much smoother than Lucas-Kanade.

On Figure 2 we can see three pairs of images, on which we further tested the methods.

Figure 2. **Images on which we tested the methods**. In the upper row is a first image of the sequence, and second in the lower one. On the left we have a two photos from a video of an office where camera is moving and changing its angle, so that objects in front are moving to the left and those in the back are moving to the right. In the middle, we have a static camera and one skier jumping through the hole (from right to left in the photo) and another one going from left to right. On the right, the climber is reaching with his left hand into the rock and moving further from the rock with his body and the camera is moving a bit to the right, so the rock is moving to the left.

On Figure 3, we can see further comparison of both methods on Images from Figure 2, together with effect of normalization of images. In Lucas-Kanade method we used neighborhood size of 3 when we didn't normalize the image (it performed better than bigger values) and size 15 when we did. For Horn-Schunck we did 500 iterations and chose $\lambda = 0.5$.

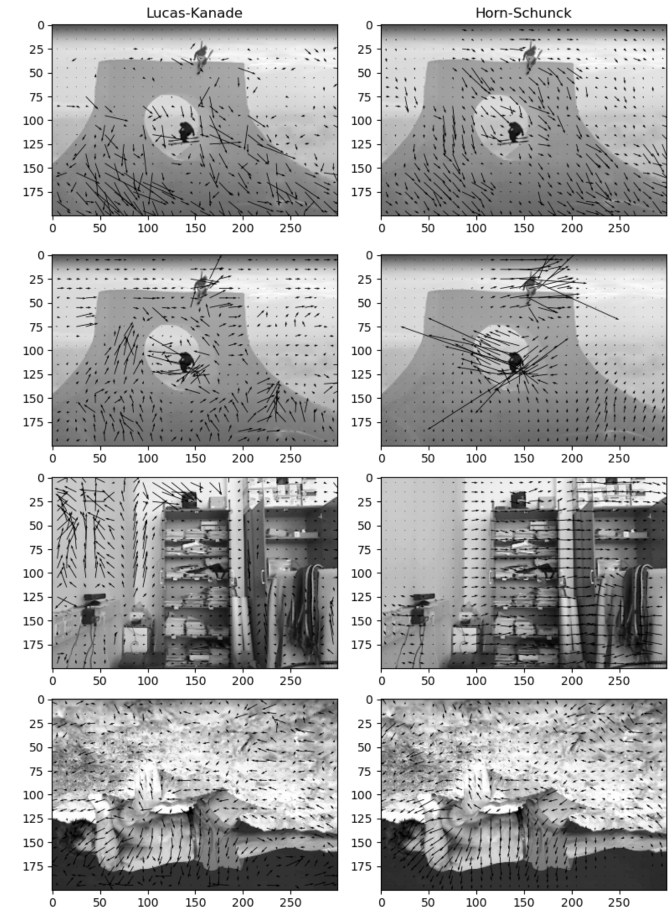

Figure 3. **Further comparison of Lucas-Kanade and Horn-Schunck method**. First image from the top shows how both methods perform if we don't normalize the images before performing the algorithms. On the second one we can see, that the normalization is a necessary step for our methods to work, so we always used it from here on. We can observe that HS method is good at recognizing the parts of the image that don't move - in contrast to LK.

To solve the problem of flow recognition on parts of the image that don't really move (in LK method), we will try to determine where optical flow can't be estimated reliably. We do that by analyzing the matrices (we have one for every pixel of the image) $A^T A = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$. To get the displacement vectors (d), we are solving systems of shape $A^T A d = A^T b$. That explicitly means dividing them by the determinant of the matrix above, so all the points in which the determinant is equal to 0 will be unreliable (to be able to do the calculations, we changed zeros for 10^{-10}). Further we will look at the eigenvalues of this matrix. If they are too small, it means that the matrix is singular, and we won't be able to invert it properly (where determinant is 0 also falls in this criteria), and if the ratio between the bigger and smaller one is too great, it means that $A^T A$ is not well conditioned. In both of those cases the optical flow can't be estimated reliably in that point. In practice, that means that either the pixels in the neighborhood are too similar (both eigenvalues small), or are changing only in one direction (big ratio). On Figure 4 we can see the output of the improved Lucas-Kanade method, that returns displacement vectors only at points that are reliable, and the mask that shows us where that is (white parts).

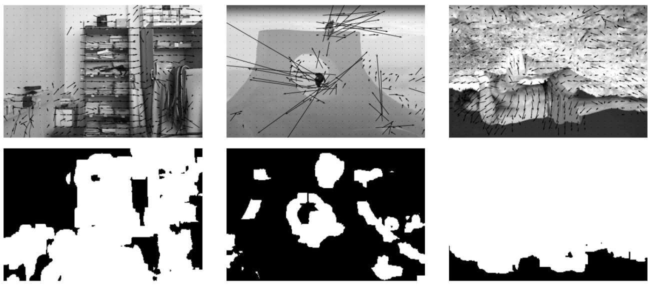

Figure 4. **Estimating reliability of Lucas-Kanade**. As we wanted to achieve, homogenous spaces without any edges are filtered out, as we clearly can see on the wall on the left picture and water behind the climber on right picture. The same happens in the middle, the field that remains on the background (in lower right part) is due to changes in lightning.

The reason for longer arrows in the middle picture is that their size is scaled automatically in the function for drawing them, so we should overlook it. We left it like this for purpose of clarity and standardization because the time in between our images is not the same - we can't set it to constant scale.

In the Lucas-Kanade algorithm, we need to determine the size of neighborhoods over which we sum the derivatives. In Horn-Schunck we need to determine the number of iterations and regularization parameter λ. We can see the comparisons of different values on Figures 5 and 6.

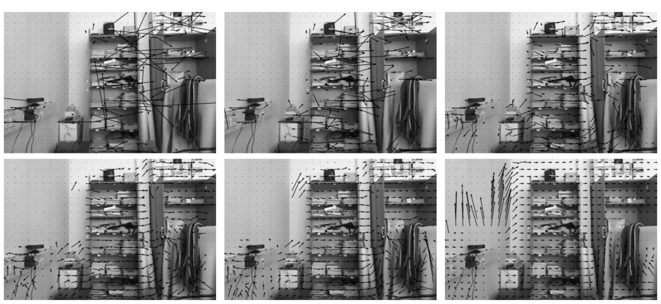

Figure 5. **Comparison of different neighborhood sizes in Lucas-Kanade**. On this six images are the results of LK algorithm (improved with reliability estimation), with neighborhood sizes of 3, 5, 10, 15, 20 and 50. On first three figures the flow is too messy, then it is becoming smoother and smoother, due to direction being result of summations over bigger and bigger neighborhoods. On the last picture we can see, that to big neighborhood could mean that a change in a small part of the image could result in a field over big part of the image - it is expected, since we are summing over 50×50 neighborhood on 300×200 image. The optimal value of this parameter could change with larger images, depending also on the size of observed moving objects.

From this experiment we chose 15 as the best for neighborhood size, observing also other images included in folder figures (not enough space to show here), 300 as optimal number of iterations (that really depends on the purpose of use) and 0.5 as the optimal regularization value.

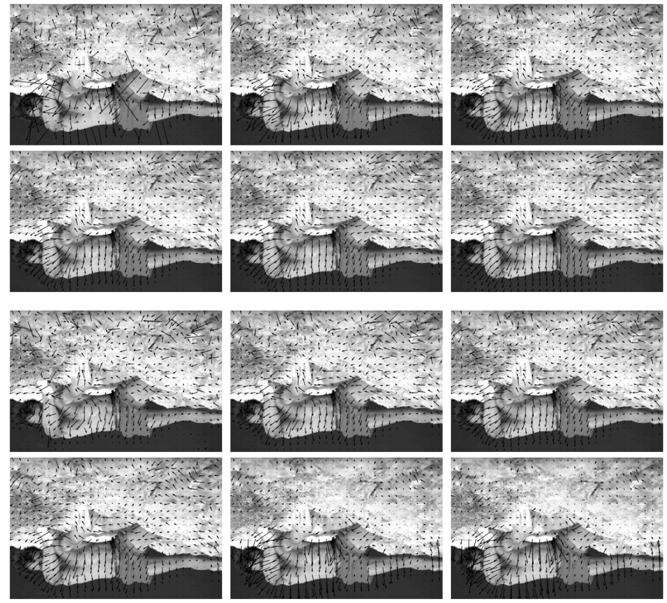

Figure 6. **Comparison of parameters in Horn-Schunck**. On the top six images we are comparing results of different number of iterations. We have results after 10, 100, 200, 300, 500 and 1000 iterations. I would say they converge enough after 200 steps (if time is a factor), to be sure we can take 300 steps. In the last two we can see that more is not better, since the field begins to spread into empty space. On the lower 6 images we are comparing values of $\lambda \in \{0.01, 0.1, 0.25, 0.5, 10, 20\}$. We can see that with bigger values the field is becoming smoother, but if we take too big value, we loose information on some parts (it looks like the rock is not moving).

Another big criteria in video analysis algorithms is also the execution time. We compared both methods, and Lucas-Kanade is significantly faster. It needs approximately $0.015s$ for execution, compared to Horn-Schunck that needs $0.22s$ for 300 steps. Even if we would take only 200 steps, it is still too slow for real time video analysis. We optimized it by initializing it with output of Lucas-Kanade, and then performing only 30 steps, which takes $0.037s$ altogether. We can see the result on Figure 7. The consequence is that we loose a big plus of HS method - no field on the background. We could fix this by initializing it with output of LK after determining where it is reliable, but such LK needs $0.45s$, so we again don't fix the time issue.

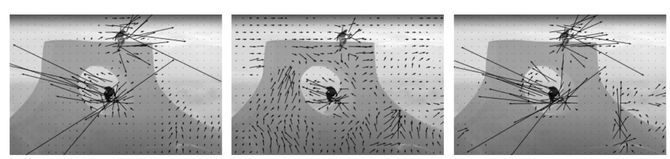

Figure 7. **Comparison of Horn-Schunck without and with initialization**. On the left we have result of normal HS, in the middle we initialized it with complete output of LK, and on the right only with the reliable part.

III. Conclusion

In this report we showed that Horn-Schnuck method has much better performance, but is also more than ten times slower, which has a big impact in real time video analysis. That's why the choice of the best algorithm completely depends on the purpose of use.