Cairo University
Faculty of Engineering
Department of Computer Engineering

Feature Extraction Module 2

Horn & Schunck

Report to explain the approach we took to make a feature extraction to group of frames

Introduction

We need a way to mesure the change of a set of frames for a specific car to see the changes in its flow so we want to calculate the optical flow to be able to describe its flow in a fast and good accuracy so we can later process it in the vif descriptor module.

Different Algorithms

We searched in the most known algorithms in making optical flow and what we found appropriate to our case are two algorithms: Horn & Schunck and lucas kanade method,

By comparing these two methods together based on[1] we determined that Horn & Schunck method are better than lucas kanade as we implemented earlier lucas kanade, we found lucas kanade is so slow as it depends on feature extraction of corners in the image, and also it’s a local optical flow but Horn & Schunck is global optical flow.

Horn & Schunck Algorithm

The proposal of Horn and Schunck[2] consists in formulating the problem of optical flow estimation as a variational problem, where the desired vector field h is defined as the minimizer of a certain energy functional J(h). This functional has two terms: a data attachment term, given by the optical flow constraint, and a regularity term that is based on the gradient of the flow:

$$J(\mathbf{h}) = \int_{\Omega} (I_x u + I_y v + I_t)^2 + \alpha^2 \left(|\nabla u|^2 + |\nabla v|^2 \right)$$

where α is a parameter to control the weight of the smoothness term compared to the optical flow constraint. The parameter α is squared so that its units are units of grey-level, and it can be regarded as the intensity of an additive Gaussian noise present in the input images.
This energy model uses quadratic functionals in both terms. This assumes that the image noise and the flow derivatives are expected to follow a Gaussian distribution. A direct consequence of this kind of functionals is that the method is very sensitive to the presence of noise and the computed flow fields are very smooth.

These shortcomings have led to the appearance of many research works that try to deal with these limitations.

The minimization of the above functional yields the following Euler-Lagrange equations:

$$I_x^2 u + I_x I_y v = \alpha^2 \mathrm{div}(\nabla u) - I_x I_t$$
$$I_x I_y u + I_y^2 v = \alpha^2 \mathrm{div}(\nabla v) - I_y I_t$$

The Laplacian can be approximated with the following expressions, which will be useful for the discretization below:

$$\mathrm{div}\,(\nabla u) \approx (\bar{u} - u),$$
$$\mathrm{div}\,(\nabla v) \approx (\bar{v} - v),$$

where u and v bar are local averages of (u, v). This approximation is analogous to the commonly used "difference of gaussians", where the Laplacian operator is approximated by the difference of blurred versions of the image, in this case the smallest blur is zero. Solving the equations above for (u, v) and re-arranging the terms, we obtain the following system of equations:

$$\left(\alpha^2 + I_x^2 + I_y^2\right)(u - \bar{u}) = -I_x\left(I_x \bar{u} + I_y \bar{v} + I_t\right)$$
$$\left(\alpha^2 + I_x^2 + I_y^2\right)(v - \bar{v}) = -I_y\left(I_x \bar{u} + I_y \bar{v} + I_t\right)$$

Writing these equations for each pixel of the input images, we obtain a sparse system of linear equations. This system can be solved efficiently with an iterative scheme.

The partial derivatives, Ix, Iy and It, are approximated using forward differences and averaging between two consecutive frames:

$$I_x \approx \frac{1}{4}\left(I_{i,j+1,k} - I_{i,j,k} + I_{i+1,j+1,k} - I_{i+1,j,k} + I_{i,j+1,k+1} - I_{i,j,k+1} + I_{i+1,j+1,k+1} - I_{i+1,j,k+1}\right)$$
$$I_y \approx \frac{1}{4}\left(I_{i+1,j,k} - I_{i,j,k} + I_{i+1,j+1,k} - I_{i,j+1,k} + I_{i+1,j,k+1} - I_{i,j,k+1} + I_{i+1,j+1,k+1} - I_{i,j+1,k+1}\right)$$
$$I_t \approx \frac{1}{4}\left(I_{i,j,k+1} - I_{i,j,k} + I_{i+1,j,k+1} - I_{i+1,j,k} + I_{i,j+1,k+1} - I_{i,j+1,k} + I_{i+1,j+1,k+1} - I_{i+1,j+1,k+1}\right)$$

The local averages (¯u, v¯) are estimated from the eight neighbors of (u, v) as:

$$\bar{u} \approx \frac{1}{6}(u^n_{i-1,j} + u^n_{i+1,j} + u^n_{i,j-1} + u^n_{i,j+1}) + \frac{1}{12}(u^n_{i-1,j-1} + u^n_{i+1,j-1} + u^n_{i-1,j+1} + u^n_{i+1,j+1})$$
$$\bar{v} \approx \frac{1}{6}(v^n_{i-1,j} + v^n_{i+1,j} + v^n_{i,j-1} + v^n_{i,j+1}) + \frac{1}{12}(v^n_{i-1,j-1} + v^n_{i+1,j-1} + v^n_{i-1,j+1} + v^n_{i+1,j+1}).$$

The coefficients of this discretization are the same as in the original paper by Horn and Schunck. In order to have a correct discretization of the Laplacian, these should be chosen so that the sum of coefficients are equal to the coefficient associated with (u, v). The solution of the above sparse system of linear equations can be obtained by means of the following iterative scheme:

$$u^{n+1} := \bar{u}^n - I_x \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{\alpha^2 + I_x^2 + I_y^2},$$
$$v^{n+1} := \bar{v}^n - I_y \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{\alpha^2 + I_x^2 + I_y^2}.$$

Implementation

Here are the steps that we implemented in the program for feature extraction module "Horn & Schunck":-

1. Inputs:
   - frame(t-1)
   - frame(t)
   - alpha: regularization constant
   - NumOfIter: number of iteration to increase accuracy
2. Compute the derivatives in fx, fy, ft
3. Initialize horizontal change matrix, vertical change matrix
4. Average flow vector to get h bar and v bar
5. Compute the new u and v

$$u \leftarrow \bar{u} - I_x \frac{I_x\bar{u} + I_y\bar{v} + I_t}{\alpha^2 + I_x^2 + I_y^2}$$
$$v \leftarrow \bar{v} - I_y \frac{I_x\bar{u} + I_y\bar{v} + I_t}{\alpha^2 + I_x^2 + I_y^2}$$

6. Go back to step 4 until numOfIter ends
7. Calculate the magnitude of the horizontal and vertical change
8. Output: horizontal, vertical and magnitude

References

[1] Pinto, Andry & Moreira, A. & Costa, Paulo & Correia, Miguel. (2013). Revisiting Lucas-Kanade and Horn-Schunck. Journal of Computer Engineering and Informatics. 1. 23-29. 10.5963/JCEI0102001.
[2] Berthold K.P. Horn, Brian G. Schunck, "Determining Optical Flow," Proc. SPIE 0281, Techniques and Applications of Image Understanding, (12 November 1981); https://doi.org/10.1117/12.965761