Exercise 09 for MA-INF 2201 Computer Vision WS24/25
05.01.2025
Submission deadline: 12.01.2025

Optical Flow

In this assignment, you are required to implement two methods (Lukas-Kanade, Horn-Schunk) to estimate the optical flow between two given images. The data for the assignment are provided in the form of some images extracted from a video, as well as ground-truth optical flows. The work should be done in the provided Python script, which consists of the following parts:

- Data loading, including images and ground truth flows (see function `load_FLO_file()`)
- Definition of the OpticalFlow class and its helper methods (already provided)
- Implementation of the Lukas-Kanade and the Horn-Schunk methods (to be filled by you, 6 points for each method)
- Implementation of the evaluation metrics: AAE (Average Angular Error) and AEE (Average Endpoint Error). The output of both functions should contain both the average and 2D per-pixel metric maps (to be filled by you, 2 points for each metric).
- The main execution loop and the visualization code of the obtained results are provided. To complete this task, you will need to complete the `flow_map_to_bgr()` visualizer function (to be filled by you, 4 points). Requirements for the visualization:
  - Run each of the two algorithms on all two pairs of provided video frames (0001-0002, 0002-0003).
  - For each run, compute the average AAE and AEE and visualize as RGB images the obtained optical flow, ground truth optical flow, per-pixel AAE, and per-pixel AEE.
  - After visualization, summarize the computed metrics for each run in a tabular format.

Guidelines for the implementation of the optical flow algorithms:

1. **Lucas-Kanade Flow**: Write your own implementation of the Lucas-Kanade optical flow as presented in the lecture. Use a 25×25 window in the algorithm.
2. **Horn-Schunck Flow**: Write your own implementation of the Horn-Schunck optical flow using an iterative scheme based on the Jacobi method as originally proposed by Horn and Schunck[1]. The iterative update rule is defined by

$$u^{(k+1)} = \bar{u}^{(k)} - \frac{I_x(I_x\bar{u}^{(k)} + I_y\bar{v}^{(k)} + I_t)}{\alpha^2 + I_x^2 + I_y^2}, \tag{1}$$

$$v^{(k+1)} = \bar{v}^{(k)} - \frac{I_y(I_x\bar{u}^{(k)} + I_y\bar{v}^{(k)} + I_t)}{\alpha^2 + I_x^2 + I_y^2}, \tag{2}$$

[1] B.K.P. Horn and B.G. Schunck, *Determining optical flow.* Artificial Intelligence, vol. 17, pp. 185 – 203, 1981

where

$$\bar{u}^{(k)} = u^{(k)} + \Delta u^{(k)} \quad \text{and} \quad \bar{v}^{(k)} = v^{(k)} + \Delta v^{(k)}. \tag{3}$$

You can approximate the laplacian $\Delta u^{(k)}$ and $\Delta v^{(k)}$ using the normalized Laplacian kernel

$$K = \begin{pmatrix} 0 & \frac{1}{4} & 0 \\ \frac{1}{4} & -1 & \frac{1}{4} \\ 0 & \frac{1}{4} & 0 \end{pmatrix}. \tag{4}$$

Set $\alpha = 1$ and initialize $u^{(0)}$ and $v^{(0)}$ with zero. Iterate until the L_2-norm difference between two flow fields is less than ϵ, i.e.,

$$\sum_{i,j} \left(|u_{i,j}^{(k+1)} - u_{i,j}^{(k)}| + |v_{i,j}^{(k+1)} - v_{i,j}^{(k)}| \right) < \epsilon.$$

Determine the value of ϵ that yields the optimal solution.