What did you learn in CSE 152?

Owen Jow

November 5, 2018

1 Low-Level Vision

After a multi-week review of linear algebra, you were given an introduction to *filtering*, one of the most basic computer vision operations. Filtering refers to a transformation on each pixel intensity as a function of neighboring pixel intensities. In the linear case, it can be seen as sliding a kernel of weights over the image and taking dot products to determine each pixel's new value. This is known as a *correlation* or *convolution* operation, the latter being when the kernel is flipped.

Note that I have been speaking in regard to the spatial domain so far. We can also view images as denizens of the frequency domain, which arises from the Fourier notion that we can represent any function in terms of sines and cosines. It turns out that an image is just a function (from x, y to intensity), so it can be represented in terms of sines and cosines too. Specifically, a 2D image can be represented through a *Fourier transform* as a weighted sum of 2D sines and cosines of different frequencies.[1] Those weights and frequencies we can plot in a *frequency domain image*, which is a function from x-frequency, y-frequency to the weight of the corresponding sinusoid.

We can directly tie frequency manipulation back to linear filtering using the *convolution theorem*, which tells us that convolution in the spatial domain is equivalent to point-wise multiplication in the frequency domain. Hence we can filter by converting everything into the frequency domain via FFT,[2] multiplying, and then converting back via inverse FFT.

Next, you learned about *feature matching*, which here means "finding points in 2+ images which represent the same point in 3D." This is a very important topic in computer vision because the ability to robustly determine correspondences in different views allows you to solve a large number of problems, including but not at all limited to optical flow and SfM.[3] Commonly, feature matching is broken down into three steps: keypoint localization, local feature extraction, and local feature matching.

There are many types of interest points; in this class, we covered Harris corners. A *corner* in an image is a point around which there are strong derivatives in two directions... [**WIP**]

2 Motion

KLT tracker depends on three assumptions... [**WIP**]

---

[1] For an image, *frequency* is cycles in intensity per pixel – as opposed to per second. Look up "spatial frequency!"

[2] *Fast Fourier transform*: an $n \log n$ algorithm to compute the discrete Fourier transform.

[3] *Structure from motion*: 3D structure and camera pose estimation from multiple views of a scene.

Acknowledgments

This document was directly inspired by Yining Liu's "what have you learned so far?" writeups.