Lecture 13

Gidon Rosalki

2026-01-14

1 Radon

To look inside someone's head, we could break it open, but we would prefer a non destructive method. The Radon Transform is essential for understanding multi dimensional structures (2D or 3D objects), but can only acquire its projections from lower-dimensional data (like 1D line integrals).
For example, in medical imaging (CT scans), we wish to reconstruct detailed internal images of the human body using only X-ray projections taken from various angles.
Line integral projections involve integrating an object along lines at different angles, illustrating how the function changes across specific directions. Consider this as taking lower dimensional projections of the objects in different directions, and observing the projections. The different direction projections give information about the higher dimensional object.

To perform the transform, we take the area t, and perform the line integral of each part, given as Q:

$$Q = \prod_{(i,j)\in \text{line}} t(i,j)$$

If we let

$$l(i,j) = \log(t(i,j))$$

So

$$\log(Q) = \sum_{(i,j)\in \text{line}} l(i,j)$$

When we have a projection in a given direction, this is effectively a signal. We may take the 1D Fourier transform of this signal, which will create a line through a central slice of the entire DFT image. If we do this for many directions, then we have many lines through central slices of the DFT. We can then perform 2D-IDFT on the resultant DFT, and get back a blurred image of the actual shape (since there are many gaps in the lower frequencies of the DFT image).

1.1 Filtered Back Projection

We will note that the distance between the outer edges of the lines in the DFT image increases linearly, since the size of a circle increases linearly. Let us define, for a given frequency (slice) w

$$H(\omega) = |\omega| \tag{1}$$
$$\widehat{F}(\omega) = H(\omega) F(\omega) = |w| F(\omega) \tag{2}$$

The resolution lowers if we have gaps too large between the angles of the samples.

2 Haze Removal

Our aim is to remove haze from an image.

Figure 1: Example

In computer vision, we often rely on prior knowledge about the world, referred to as a "prior". For this problem, we will utilise the *dark channel prior*. This means that if we have an image without haze, then our dark channel will be very low, but a hazy image will have a strong dark channel.
The dark channel is defined as

$$\min\{\text{RGB on a local patch}\}$$

Or more mathematically:

$$J^{\text{dark}}(x) = \min_{y \in W(x)} \left\{ \min_{c \in \{R,G,B\}} \{J^c(y)\} \right\}$$

So for example, we take 15 by 15 patches, and the value of each patch will be the minimum value of all the reds, greens, and blues, in that patch.

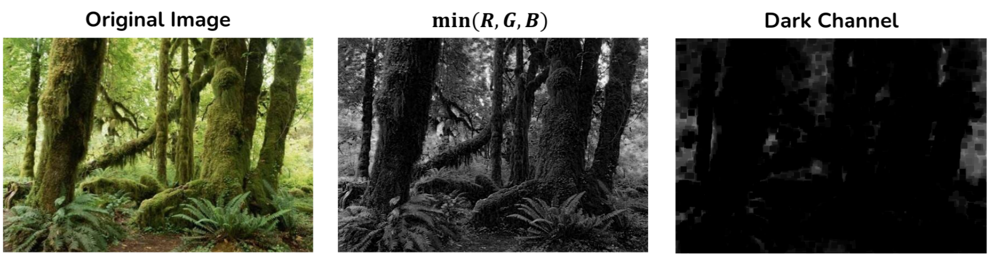

Figure 2: The image, the minimum RGB for each pixel, the dark channel

It was found (heuristically) that for haze free images, 86% of the pixels of the dark channel will be in $[0, 16]$. Almost all the pixel intensities of the dark channel are incredibly low, when the image is haze free. It is dark since in images there is lots of shade, and shade is dark, and also there are often lots of colourful objects, which will have a channel which is very dominant, but will also have a channel with a very low value, so the dark channel becomes dark.

However, if you take the dark channel of a hazy image (we are only dealing with white haze), then the dark channel is no longer dark.

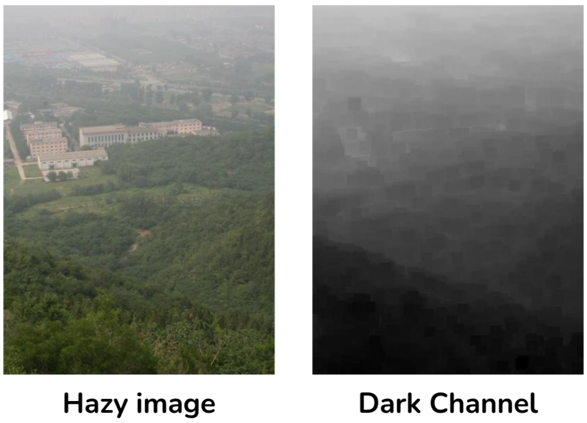

Figure 3: Hazy image, and dark channel

In order to restore, we will assume that the atmosphere is either white, or a shade of grey that is very close to white. Let us define I to be the hazy image, J to be the scene without haze, t to be the transmission, and A is the haze / atmospheric light.

$$I = J \cdot t + A \cdot (1 - t)$$

We may compute t (which is functionally the **depth** of the image, so $1 - t$ is the dark channel) by the function

$$t(x) = e^{-\beta d(x)}$$

Where d is the depth. We do not need the depth for every pixel, just one, since we may compute the dark channel, and then from the dark channel of a single pixel, we may compute β, and then we can compute the depth for every pixel in the image. From there, since we have assumed A to be constant across the image, we may reconstruct the original image J.

3 Multi Frame Super resolution

Given a set of low quality images, where each image is offset by a fraction of a pixel relative to the others, we can fuse these together into a higher resolution image.

We would like to get a uniform sampling, with sampling distance $\leq D$. Due to our limited camera resolution, we have sampled using a grid with distance $2D$. If we take 4 images, each a distance of D apart, to the right, below, and then to the left, we may interlace the four images to get the desired resolution image.

Note that we have no guarantee of the pixel offsets, that they will be exactly the amount, or the distance that we desire. They could be in the wrong directions, and even the wrong distances. There may even be rotations present.

We can assume that the observed low resolution images Y_i are generated from the high resolution image X by translating, blurring, and sampling. Given the low resolution images Y_i, we look for the high resolution image X, which has the simulated projection $P_i(X)$ that are closest to Y_i:

$$E(X) = \sum_{i=1}^{N} ||P_i(X) - Y_i||^2$$

Where P_i is the simulated known projection of image X onto the grid of the image Y_i. We may then compute P_i by motion analysis.

Steps:

1. Calculate the sub pixel motion between the low resolution images Y_i (You may use Lucas-Kanade for this)
2. Initial guess for the high resolution image X could be, for example, the average of all the aligned and enlarged low resolution images Y_i
3. For each Y_i perform the back projection process:
   (a) Compute the projection error $P_i(X) - Y_i$ (P_i is the transformation for Y_i)
   (b) Change the pixel values of X to minimise the above error by back projecting the error