Working with Images

1 The mean, variance, and median images

A video $\mathcal{V} = \{I_1, I_2, \ldots, I_T\}$ is a sequence of images where I_t is an image captured at time t (i.e., video frame). We can represent each image in the video as $M \times N$ matrix I_t. We can also think of the video as a spatio-temporal function $I(x, y, t)$ as the one shown in Figure 1.

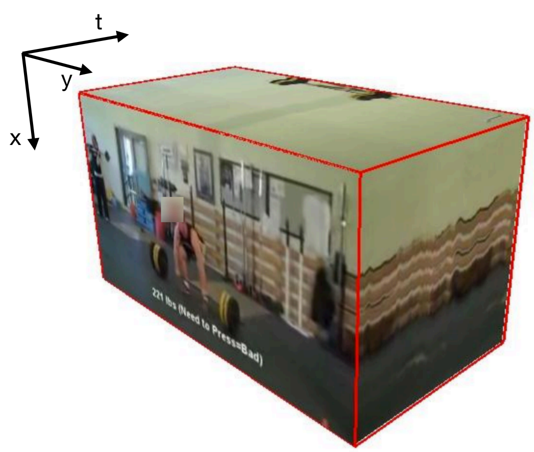

Figure 1: Video as spatio-temporal volume (Figure from: `http://www.mikelrodriguez.com/representing-videos-using-mid-level-discriminative-patches/`)
.

We can calculate the temporal average image by averaging all image matrices:

$$\bar{I} = \frac{1}{N}\sum_{t=1}^{T} I_t, \tag{1}$$

```
v = zero(rows, cols, depth)
v = readvideo(filename)
m = mean(v, dim, temporal)
```

which is equivalent to averaging the function $I(x, y, t)$ over time, i.e.:

$$\bar{I} = \frac{1}{N}\sum_{t=1}^{T} I(x, y, t). \tag{2}$$

The temporal variance image is given by:

$$S = \frac{1}{N} \sum_{t=1}^{T} \left(I_t - \bar{I} \right)^2 . \tag{3}$$

The temporal *median image* is another interesting result we can calculate in videos that have been captured by a static camera. To calculate the median image, we can calculate the median of each pixel in the video along the temporal dimension.

Given a video as a spatio-temporal volume (i.e., function), we can calculate some interesting quantities such as partial derivatives. For example, we can calculate the partial-derivative with respect to time, i.e.:

$$\frac{\partial I\left(x, y, t\right)}{\partial t} \approx I\left(x, y, t + \Delta t\right) - I\left(x, y, t\right), \tag{4}$$

which can help us detect differences between two consecutive images (e.g., motion, defects).

2 Pixel operations

2.1 Log enhancement

The logarithm operator can be applied to images to enhance the low-intensity pixel values while maintaining high-intensity pixel values mostly unchanged. The log operator is given by:

$$Q = c \log\left(1 + I\right), \tag{5}$$

where I is an image. We can choose the scaling constant c so that the pixel values in Q are in the range $[0, 255]$. If R is the maximum pixel value in image I, then:

$$c = \frac{255}{\log\left(1 + I\right)}. \tag{6}$$

It can be a good idea to apply log transformations to astronomy images to enhance low-intensity pixel regions that cannot be distinguished by visual inspection.

2.2 Linear blending operator

The blending between two images can be achieved by the following simple linear combination:

$$C = \left(1 - \alpha\right) I_1 + \alpha I_2, \tag{7}$$

where $\alpha \in [0, 1]$. For example, for $\alpha = 0.5$, the combined image C is a equal mixture of the pixel intensities (or colors) in I_1 and I_2. As we vary the value of α from 0 to 1, the mixture proportions change resulting in a temporal cross-dissolve sequence of images.

The linear blending operator in Equation 7 is also used for combining images weighted by an alpha mattes. In this case, the α parameter becomes a map of the same dimension of the dimension of the input images. An example of an alpha matte is given in Figure 2.

Figure 2: Alpha mattes with wide feathering (Figure from: `https://cs.brown.edu/courses/csci1290/labs/lab_compositing/index.html`)

The alpha matte in Figure 2 can be created by sigmoid function, i.e.:

$$s\left(x, y\right) = \frac{1}{1 + e^{-x}} \tag{8}$$