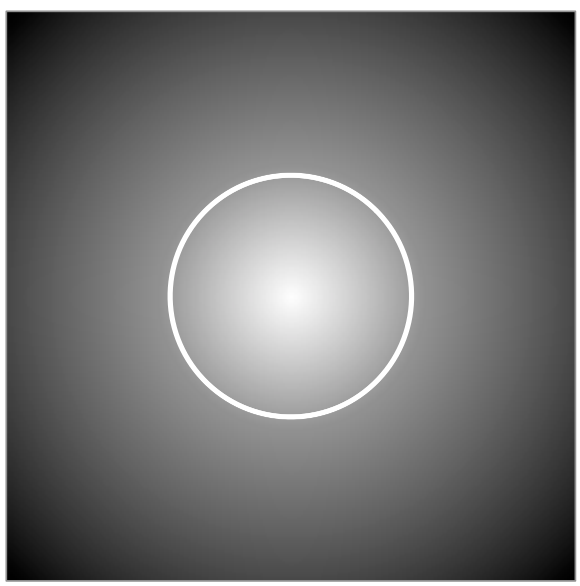
Half Maximum Flux Diameter: The Discrete Case

Wagner L. Truppel

October 17, 2012

Definition

The *Half Maximum Flux Diameter* (HMFD or HFD) of an image is defined as the diameter of the circle centered on the image's brightness centroid such that half of the total brightness of the image lies within the circle, after the background has been filtered away. In the example illustrated below, the white circle includes half of the total image's brightness and, therefore, its diameter is the image's half maximum flux diameter. Note that, although the inside is much brighter than the outside, there are many more bright pixels outside than inside. The smaller the HFD, the more focused the image is.



Computing the HFD: The continuum case

How does one compute the HFD? The definition is quite clear so, in the case of a continuous distribution of brightness, defined such that $b(r,\theta)\, rdr\, d\theta$ is the brightness falling onto the elemental area $rdr\, d\theta$ between radii r and $r+dr$ and angles θ and $\theta+d\theta$, we'd have

$$\frac{\displaystyle\int_0^{2\pi}\int_0^{R} b(r,\theta)\, rdr\, d\theta}{\displaystyle\int_0^{2\pi}\int_0^{+\infty} b(r,\theta)\, rdr\, d\theta} = \frac{1}{2}$$

when R equals the radius of the circle we're after. Suppose, for example, that the brightness per unit of area decreases according to a normal distribution, with circular symmetry, so that

$$b(r,\theta) = b_0 \exp(-ar^2)$$

where $b_0 > 0$ and $a > 0$ are parameters describing the maximum intensity and the decay rate, respectively. Then,

$$\int_0^{2\pi}\int_0^{R} b(r,\theta)\, rdr\, d\theta = \pi\,\frac{b_0}{a}\,(1-e^{-aR})$$

and

$$\int_0^{2\pi}\int_0^{+\infty} b(r,\theta)\, rdr\, d\theta = \pi\,\frac{b_0}{a}\,.$$

Therefore, imposing a ratio of 1/2 gives us

$$1-e^{-aR} = \frac{1}{2} \qquad \Rightarrow \qquad R = \frac{\ln 2}{a}\,.$$

The HFD is then $2R = 2\ln(2)/a$. Different brightness distributions generally result in different values of R.

Computing the HFD: The discrete case

Unfortunately, real-world images are not continuous but, rather, are pixelated. Each pixel occupies a small rectangular section of the image and the brightness falling over each pixel can be considered uniform. How do we compute the HFD then? The image used as an example above might actually look something like the image below.

We now have a grid of pixels, each pixel possibly not square in shape but rectangular, each of which associated with a brightness value. At least one reference on the internet

finds the discrete HFD by computing a weighted average of the distance of each pixel to the centroid, using the pixel's brightness as the weight:

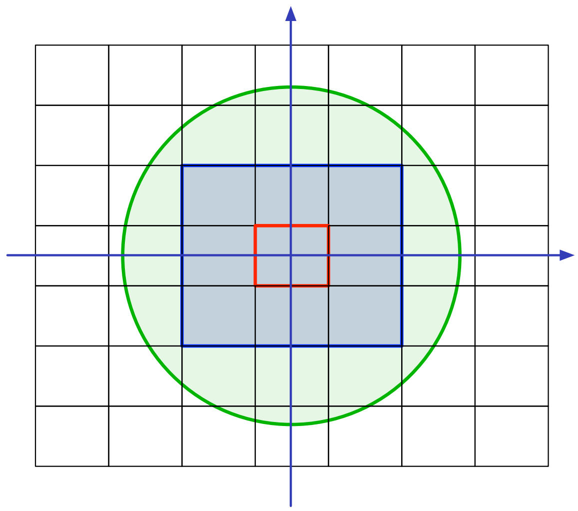
$$HFD = 2R = 2\,\frac{\sum b_i d_i}{\sum b_i}\,,$$

where both sums extend over all pixels of the image, b_i is the brightness of the i-th pixel and d_i is its distance to the brightness centroid. This method, as it turns out, is **incorrect**. It is analogous to using the expression derived above in the continuous case for *all* brightness distributions, regardless of whether or not they actually follow a normal decay. There is *no* analytical way of computing the discrete HFD without making assumptions about the brightness distributions, just as would be the case in the continuous situation. The *only* way to accurately compute the HWD is to actually use its definition.

Computing the HFD: Dealing with the grid, part I

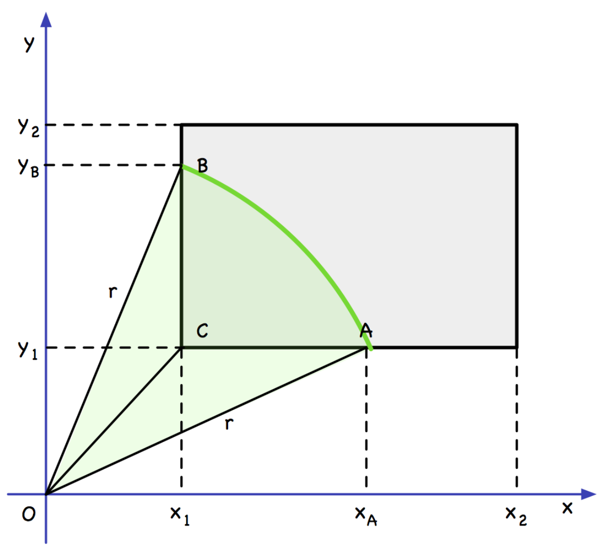
Suppose that we want to compute the total amount of brightness falling within a circle of a given radius, as illustrated below. Pixels are depicted as rectangular areas of the image and the pixel marked in red is the brightness centroid. We want to compute the total amount of brightness within the green circle.

We erect a coordinate system centered on the brightness centroid, with the x-axis extending to the right and the y-axis extending upwards. The green cirlce encompasses a



number of ‘full pixels’, pixels that are entirely contained within it, but also receives contributions from pixels that are only partially contained within it. We shall now look at how to obtain the contribution from each of those partial pixels.

So, consider the situation depicted below. The green arc is part of a circle of radius r and it crosses the pixel whose corners have the coordinates shown. Our goal is to compute the area contained within the intersection of the rectangle and the circular section.



First we need the coordinates of the intersection points A and B. It's easy to see that

$$x_A = r\,\cos(\theta_A) \quad \text{where} \quad \sin(\theta_A) = y_1/r$$

and

$$y_B = r\,\sin(\theta_B) \quad \text{where} \quad \cos(\theta_B) = x_1/r$$

where θ_A and θ_B are the angles from the x-axis towards the segments $\overline{\mathrm{OA}}$ and $\overline{\mathrm{OB}}$, respectively. Naturally, it's required that $|x_1| \leq r$ and $|y_1| \leq r$ for these angles to be real.

Next, it's apparent that the total area of the circular section AOB is the sum of three parts, the areas of the two triangles ΔOCA and ΔOCB and the area of the intersection we're interested in, ACB:

$$S_{\mathrm{AOB}} = S_{\Delta\mathrm{OCA}} + S_{\Delta\mathrm{OCB}} + S_{\mathrm{ACB}}\,.$$

Hence,

$$\begin{aligned} S_{\mathrm{ACB}} &= S_{\mathrm{AOB}} - S_{\Delta\mathrm{OCA}} - S_{\Delta\mathrm{OCB}} \\ &= \frac{1}{2}\,r^2\,|\theta_B - \theta_A| - \frac{1}{2}\,|x_A - x_1|\,|y_1| - \frac{1}{2}\,|y_B - y_1|\,|x_1|\,, \end{aligned}$$

where the angles are measured in radians. On the other hand, the area of the rectangle is

$$S_{\mathrm{rect}} = |x_2 - x_1|\,|y_2 - y_1|\,.$$

Therefore, the amount of brightness this rectangle contributes to the circle is

$$\frac{1}{2}\,\frac{r^2\,|\theta_B - \theta_A| - |x_A - x_1|\,|y_1| - |y_B - y_1|\,|x_1|}{|x_2 - x_1|\,|y_2 - y_1|}$$

times the brightness of the pixel represented by that rectangle.

Computing the HFD: Dealing with the grid, part II

The next questions of interest are how many such partial pixels are there as a function of the circle's radius r and what are their coordinates? If we can write analytical expressions for the coordinates of each partial pixel, given a radius r, then we need not perform any search at all for those pixels and the task of computing the total brightness contained within a circle of a given radius becomes very simple.

■