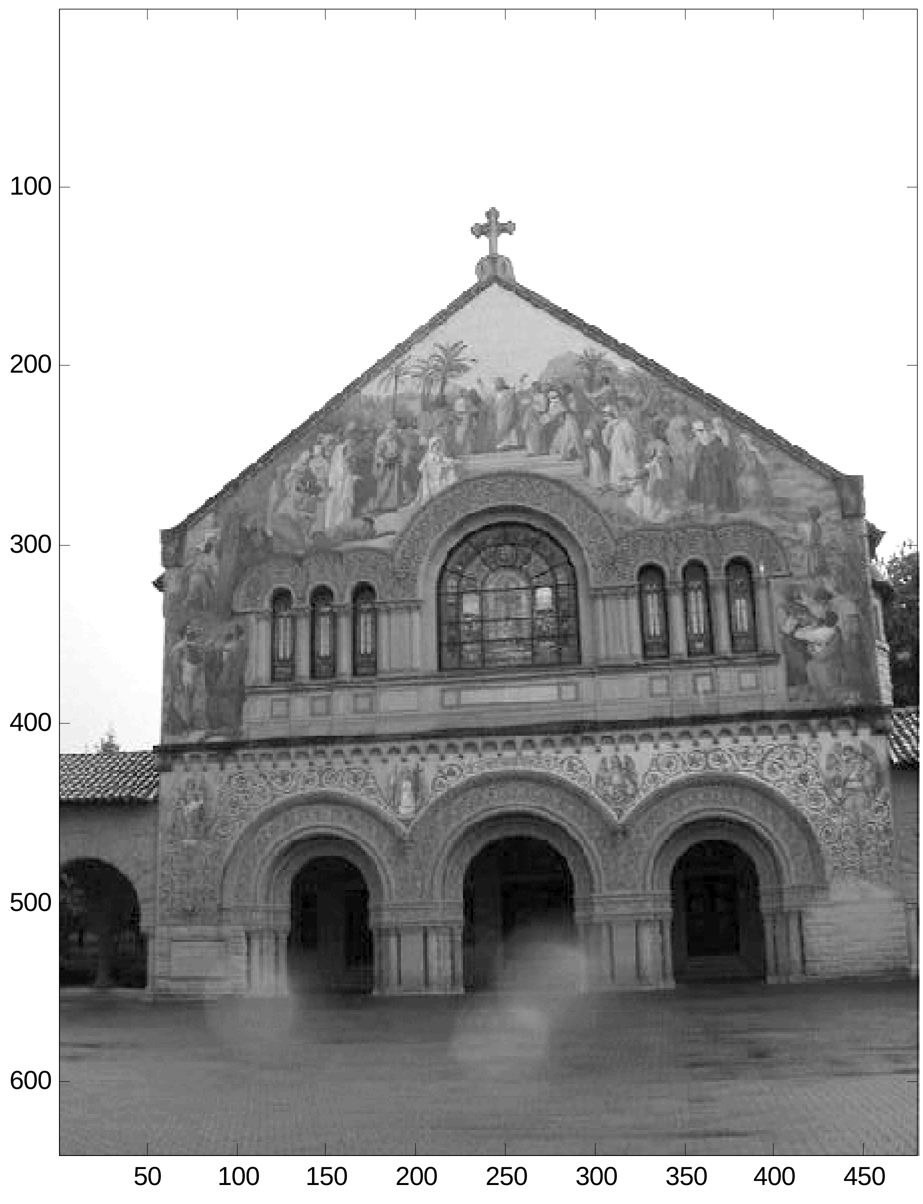
Christopher Tsai

Problem Set III – Two-Dimensional Impulse Functions

Problem #5 – Two-Dimensional Impulse Sampling

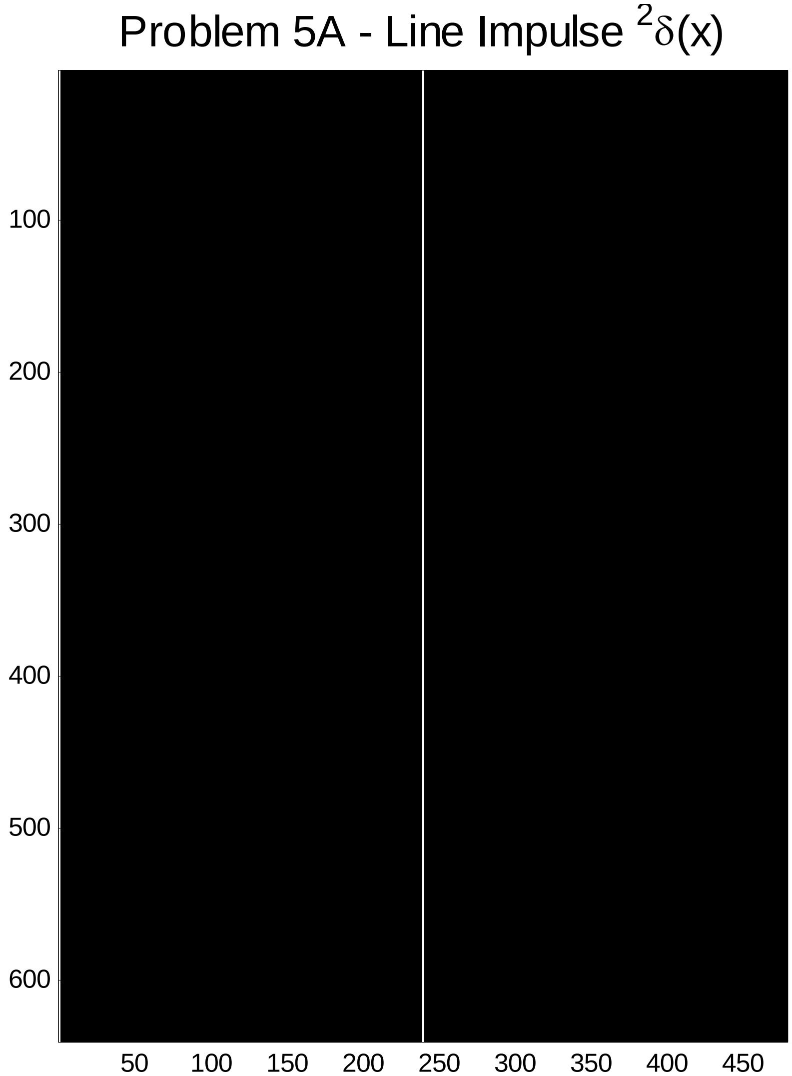
Before we begin sampling the image, decorum calls for our initial acquaintance with its form and face. Thus, let us glimpse the object of our affections. Apparently, we are sampling a photo of Memorial Church:

Problem 5 - Memorial Church Image to Be Sampled



We superimpose a Cartesian coordinate system on this image, with origin at $(x, y) = (241, 321)$. In other words, the 321st row is our x-axis, and the 241st column is our y-axis.

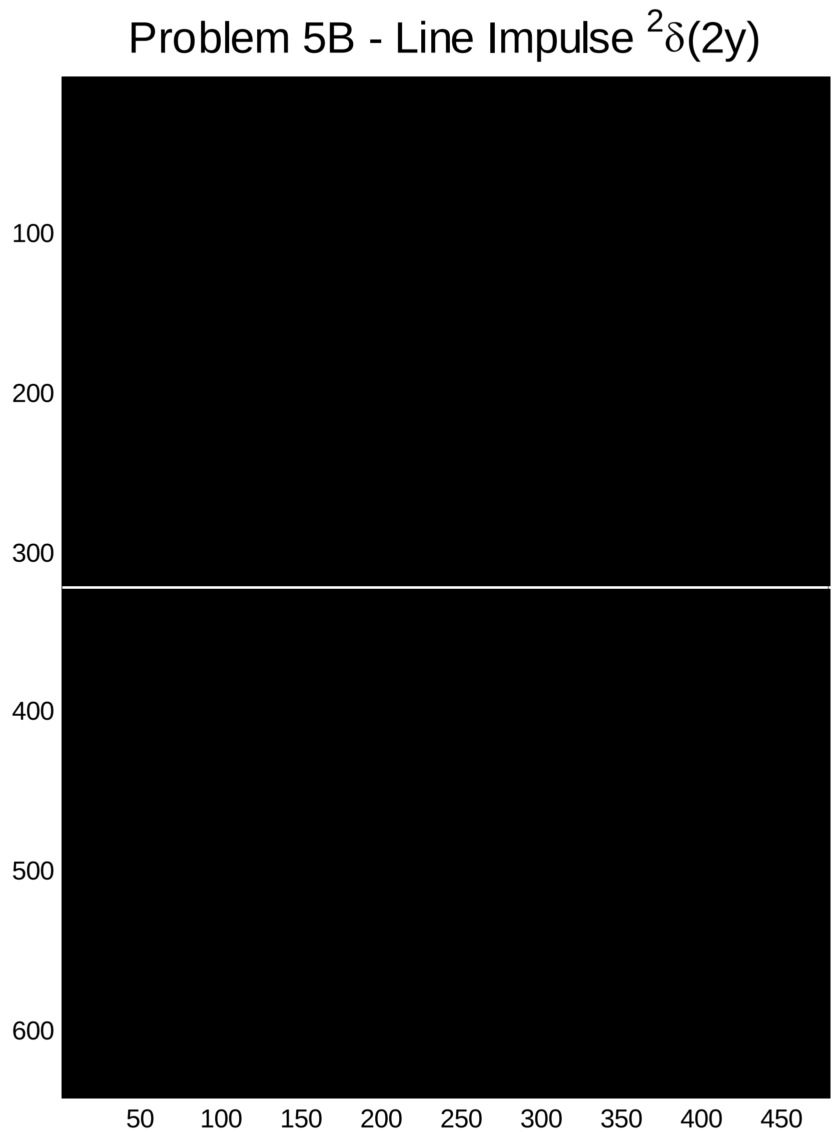
When we sample an image with the line impulse ${}^2\delta(x)$, we integrate the image function across the vertical line $x = 0$, since the impulse is zero for all other x:



Since the y-axis runs down the 241st column, across all 640 rows, our line sampling yields the value

$$\sum_{y_i=0}^{640} f(0, y_i) = 87{,}164$$

Similarly, when we tap the line impulse $^2\delta(2y) = \frac{1}{2}\ ^2\delta(y)$, we integrate the image function across the horizontal line $y = 0$, since the impulse function is zero for all other y. Thus, we ignore the image at all coordinates off the x-axis, as our discretized sampling function displays:

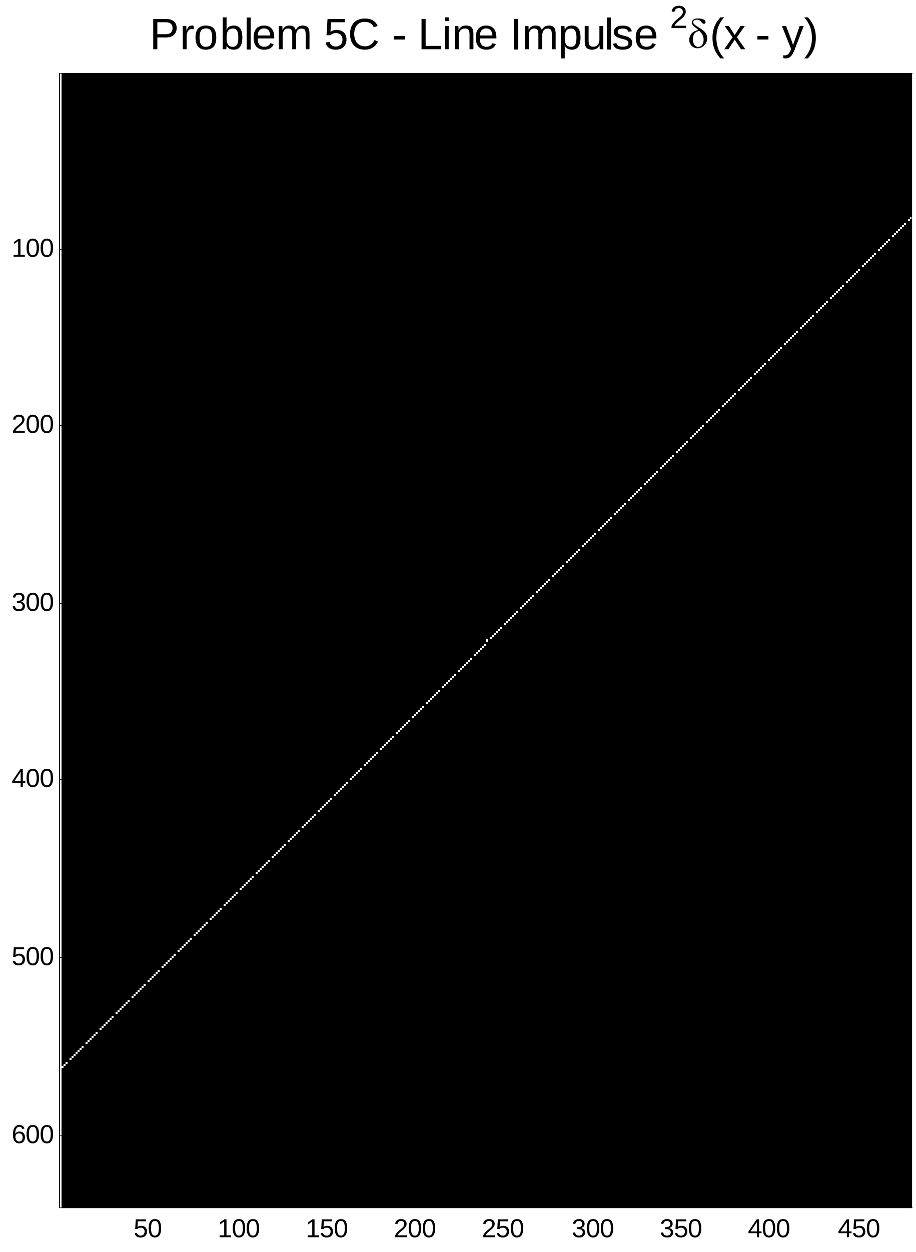


With the x-axis spanning the 321st row, across all 480 columns, our line sampling yields the value

$$\frac{1}{2}\sum_{x_i=0}^{480} f(x_i, 0) = 28{,}433$$

where we scale by $\frac{1}{2}$ due to the multiplicatively inversely scaled impulse argument.

Sampling with the line impulse $^2\delta(x-y)$ integrates – or, rather, accumulates – the image along the line $y = x$, which passes through the origin, amassing values that reside in positions with equal x and y coordinates. Using this criterion as a mask for our image, we observe the sampling:



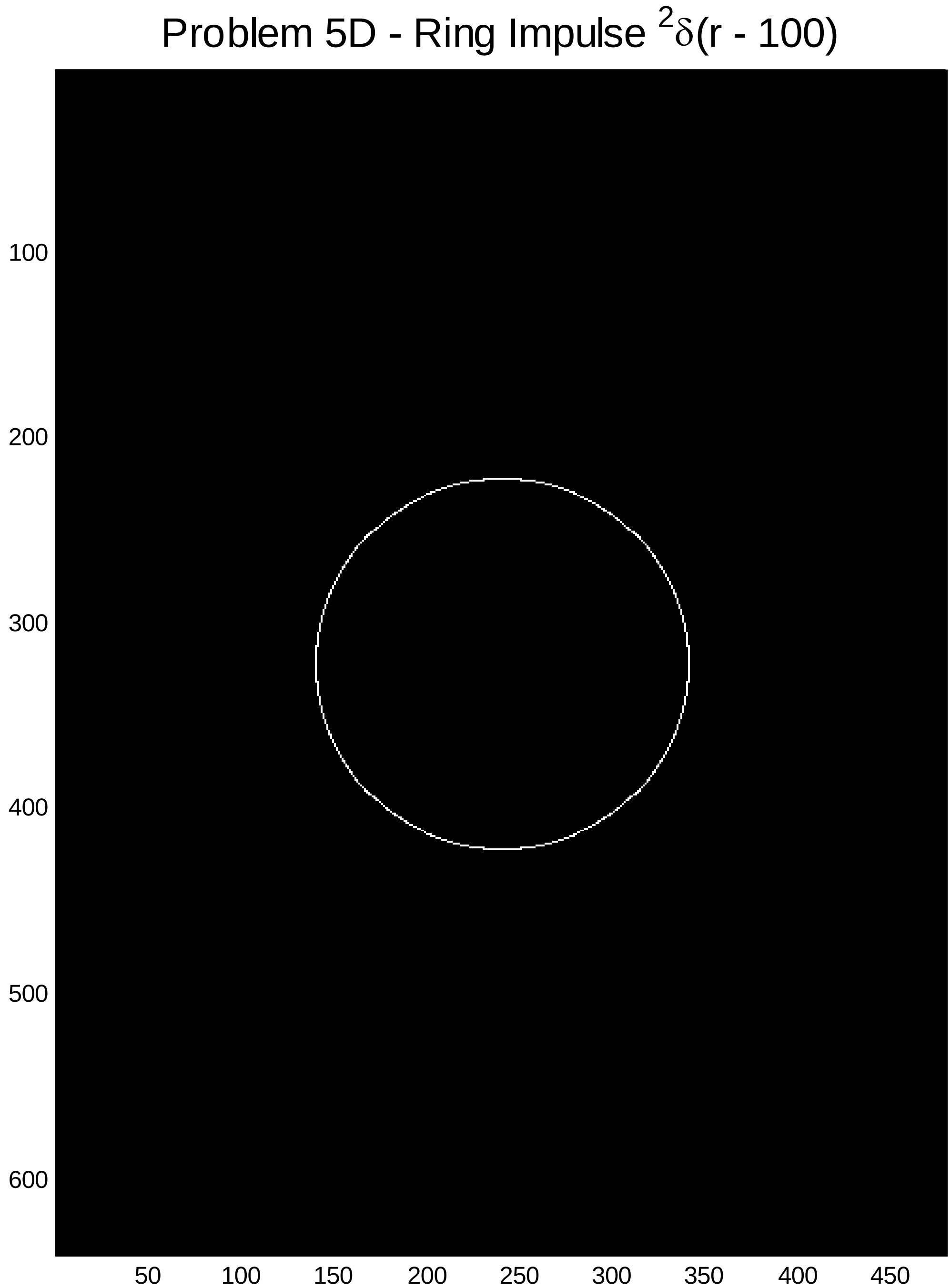
Accumulating image samples across this line, we evaluate the sampling as the sum:

$$\sum_{x_i=y_i} f(x_i,\, y_i) = 69{,}694$$

Finally, if we sample with the circularly symmetric ring impulse $^2\delta(r-100)$, then we must evaluate and accumulate image pixel values at all points that reside a radial distance of 100 away from the designated image origin. However, because our image is a matrix supported by a rectangular grid, the discretized pixels will not always fall exactly 100 units away from the origin; in fact, those points would not consummate a full circle, since the points along any continuous circle intersect grid points perfectly only at a few integral right triangle coordinate pairs like (±60, ±80) or (±80, ±60); elsewhere, the discretized grid points are only approximately 100 units away, so we must accommodate for these nearly-100 values in order to complete the circle. Thus, points that should but fail to be 100 units away from the circle will be a small fraction (< 0.5) away from the perfect integer radius 100, so we also include those points in our quasi-circular sampling ring, which we must fit to our rectangular sampling grid.

Remark that we cannot inflate this tolerance too much about 0.5, since we want our ring impulse to be at most one pixel thick all around the circle. If we make the tolerance too stringent, then the points that fall far off the smooth (continuous) circle due to discretization will be omitted despite their belonging to the circle; meanwhile, loosening the tolerance will admit points farther away from the circle, hence thickening the ring impulse beyond unity and possibly inflating the resulting sampling sum.

Our ring impulse, restricted with the germane 0.5 tolerance, bears the following appearance, ascertaining our consummatory choice:



Accruing our image pixel values at all points approximately 100 units away from the prescribed origin, we compute our ring-sampled image to be

$$\sum_{\left|\sqrt{x_i^2+y_i^2}-100\right|<\frac{1}{2}} f(x_i,\ y_i) = 77{,}281$$