

## MATLAB Edge Detection

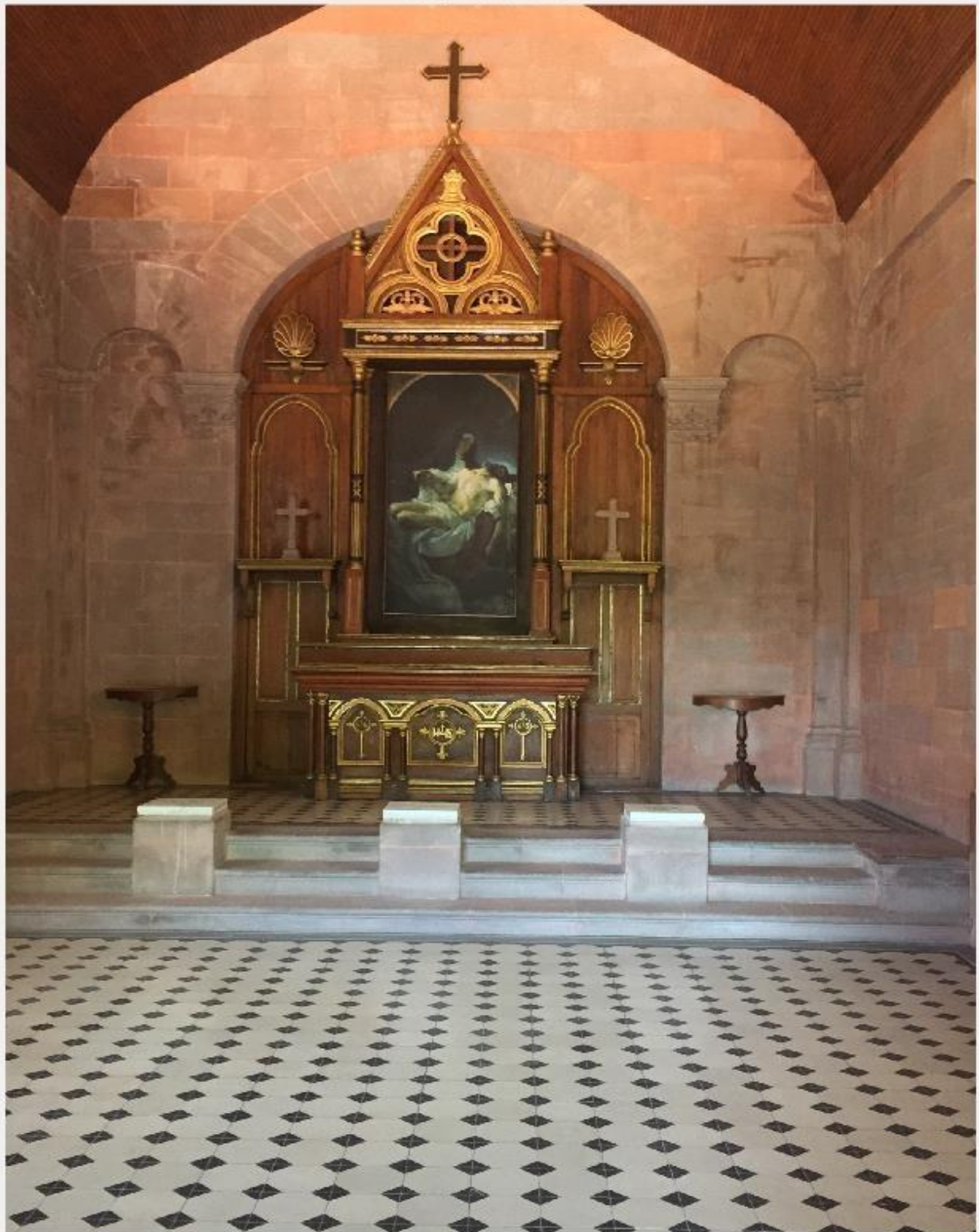
```
Capilla.m x +
1 % Jose Luis Madrigal, Paulo Ogando, Cesar Palome, Christian Parrish, Carlos
2 % Ceballos y Ricardo Antonio Cervantes
3 % carga la imagen a color '04. capilla.jpg' en la variable capilla_color
4 capilla_color = imread('05. Capilla Maximiliano, Miramon y Mejia.jpg');
5 % produce, a partir de capilla_color, la imagen a tonos de grises capilla_gs
6 capilla_gs = rgb2gray(capilla_color);
7 % produce una imagen a color suavizada con un filtro gaussiano de desviación %
8 % estándar igual a 5 unidades.
9 capilla_gaussian_sd_05 = imgaussfilt(capilla_color,5);
10 % produce una imagen en tono de grises a partir de la imagen a color
11 % suavizada con filtro gaussiano de desviación estándar igual a cinco
12 % unidades.
13 capilla_gaussian_sd_05_gs = rgb2gray(capilla_gaussian_sd_05);
14 % detección de bordes con el algoritmo de Prewitt en la imagen en tonos
15 % de grises capilla_gs
16 capilla_gs_prewitt = edge(capilla_gs,'Prewitt');
```

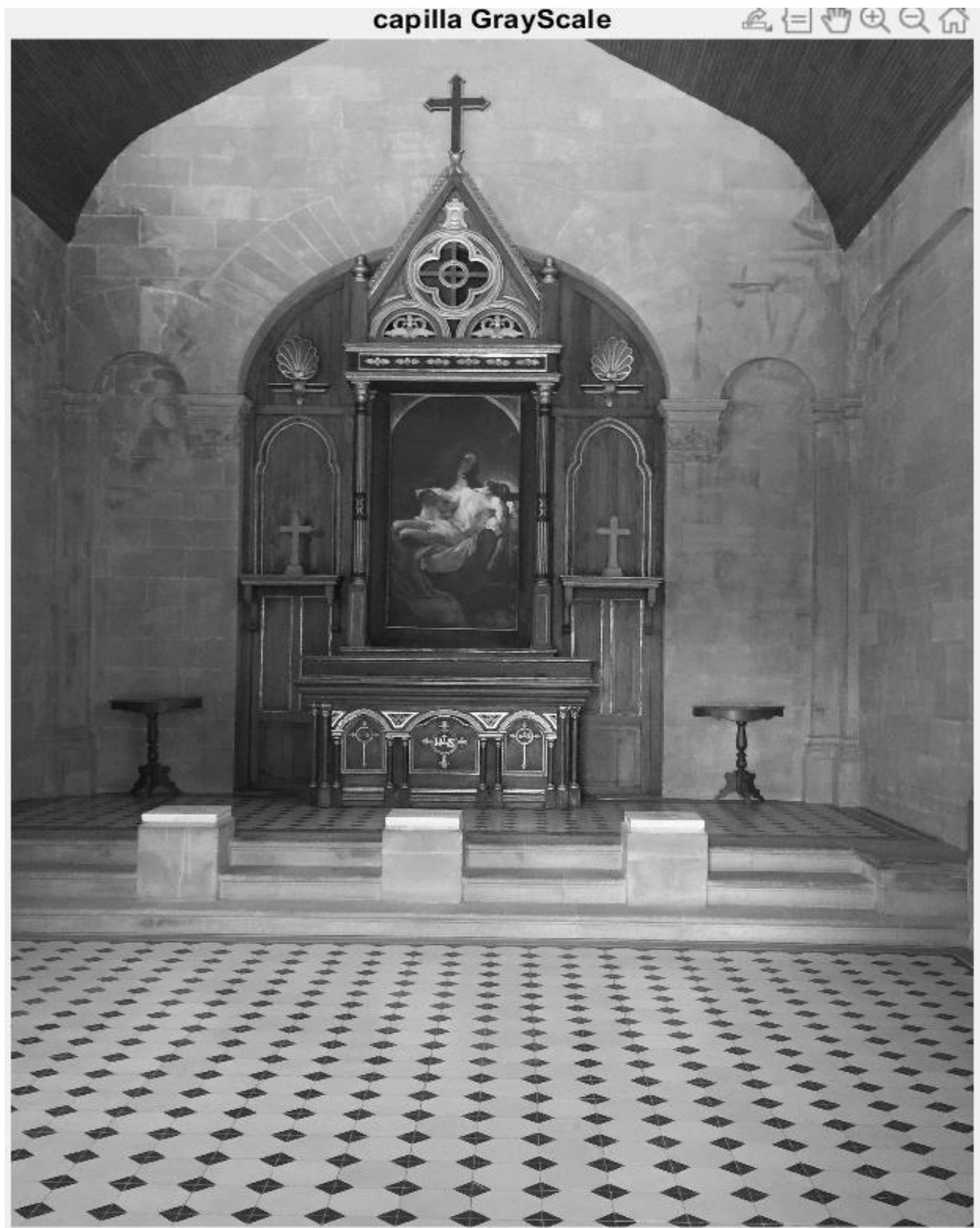
The Prewitt, Sobel, Canny and Roberts edge detection algorithms were applied, with and without a Gaussian filter.

This was applied to 10 images, but I will only show 1 as an example, although it was true that, for each image, there was a more effective algorithm than the others, which depended on the elements and tones that were had.

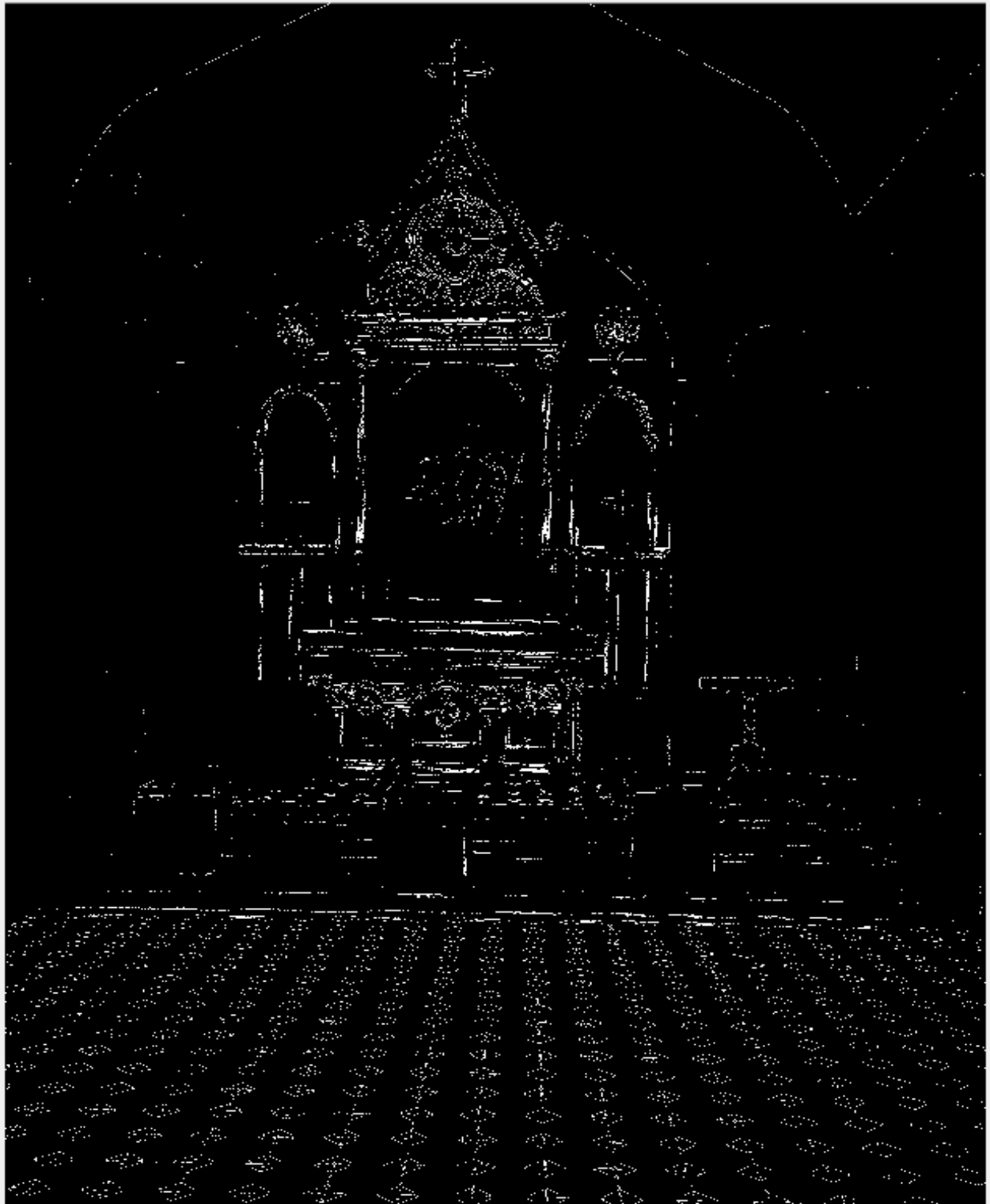
The image of the Maximiliano, Miramon and Mejia Chapel was taken by my teacher, Salvador Elías Venegas Andraca, so he is the author.

capilla color





capilla GrayScale Prewitt



capilla GrayScale Sobel

