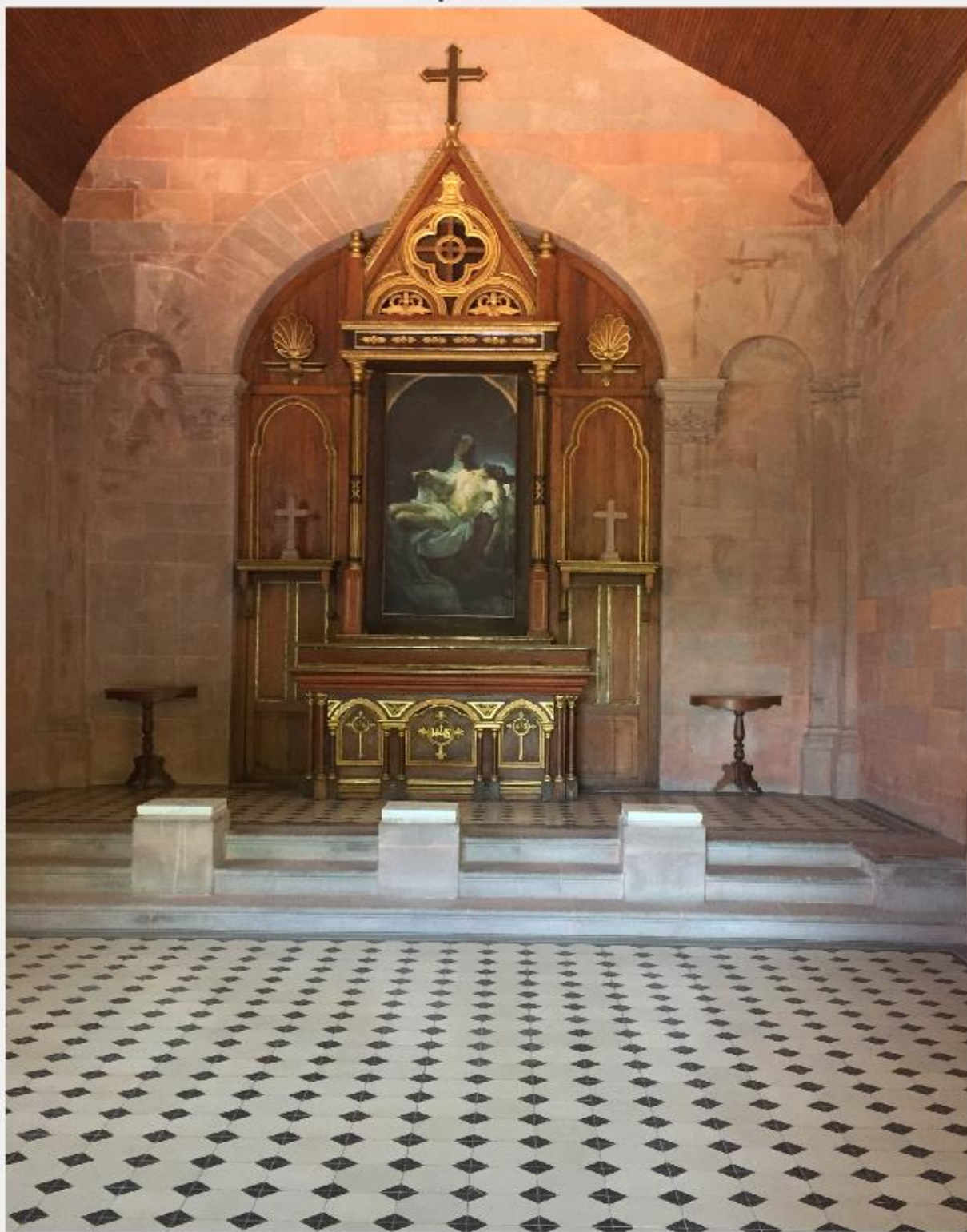José Luis Madrigal

MATLAB Edge Detection

```matlab
1   % carga la imagen a color ''04. capilla.jpg' en la variable capilla_color
2   capilla_color = imread('05. Capilla Maximiliano, Miramon y Mejia.jpg');
3   % produce, a partir de capilla_color, la imagen a tonos de grises capilla_gs
4   capilla_gs = rgb2gray(capilla_color);
5   % produce una imagen  a color suavizada con un filtro gausiano de desviación %
6   % estándar igual a 5 unidades.
7   capilla_gaussian_sd_05 = imgaussfilt(capilla_color,5);
8   % produce una imagen en tono de grises a partir de la imagen a color
9   % suavizada con filtro gausiano de desviación estándar igual a cinco
10  % unidades.
11  capilla_gaussian_sd_05_gs = rgb2gray(capilla_gaussian_sd_05);
12  % detección de bordes con el algoritmo de Prewitt en la imagen en tonos
13  % de grises  capilla_gs
14  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.

José Luis Madrigal

capilla color

José Luis Madrigal

José Luis Madrigal

José Luis Madrigal

capilla GrayScale Sobel

José Luis Madrigal

capilla GrayScale Canny

José Luis Madrigal

capilla GrayScale Roberts

José Luis Madrigal

capilla GrayScale Gauss 5 sd Prewitt

José Luis Madrigal

capilla GrayScale Gauss 5 sd Sobel

José Luis Madrigal

capilla GrayScale Gauss 5 sd Canny

José Luis Madrigal