Project write up - Finding car lanes

Reflection:

My pipeline consisted of the following steps:

1. Convert image to gray-scale. I had to change this for the videos as they expect a color image.
2. Apply the gaussian blur filter with a threshold of 3
3. Detect the edges using the Canny edge function. Getting the parameters right 50, 150 was a bit of trial and error.
4. Eliminate all the unwanted regions in the image. It took a lot of trial and error to get this one right. The forum discussions were very helpful. I had done this wrong completely and was stuck in the assignment as I was expecting the error in the average/extrapolate function.
5. Apply Hough lines
   a. Averaging extrapolating the line segments. I tried three approaches and all of them had the common step of separating the segments into left and right lines based on slope. Then
      i. Just find the xmin, xmax and ymin, ymax of the line segment and draw the line. This didn't work really well.
      ii. Use np.polyfit - I just couldn't understand the logic with polyfit.
      iii. Use scipy curve fit - This was an option suggested by my mentor. Worked really well.
6. Finally blend the lines with the real image using the weighted image function.

Potential shortcomings with this pipeline:
A lot of trial and error to find the correct parameters in almost all of the steps mentioned above. Also the parameters won't work if there are curved edges or sharp turns. Perhaps you can refine the parameters for curves but you have to keep refining this process for every other change.

Suggest possible improvements:
One possible improvement can be the dynamic elimination of unwanted edges instead of a hard coded region match. I am not sure how to do this.