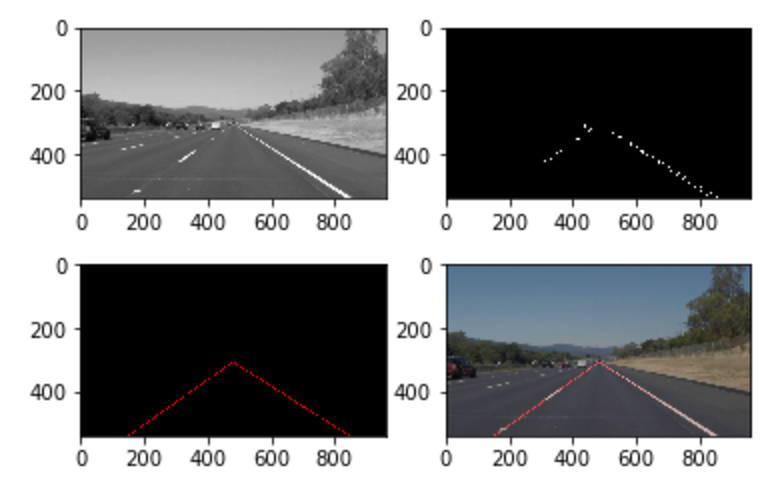
I start with importing the necessary Python libraries (matplotlib, numpy, cv2, math) that will be used later. A test image is shown as follows and its dimension is printed out (540\*960\*3).

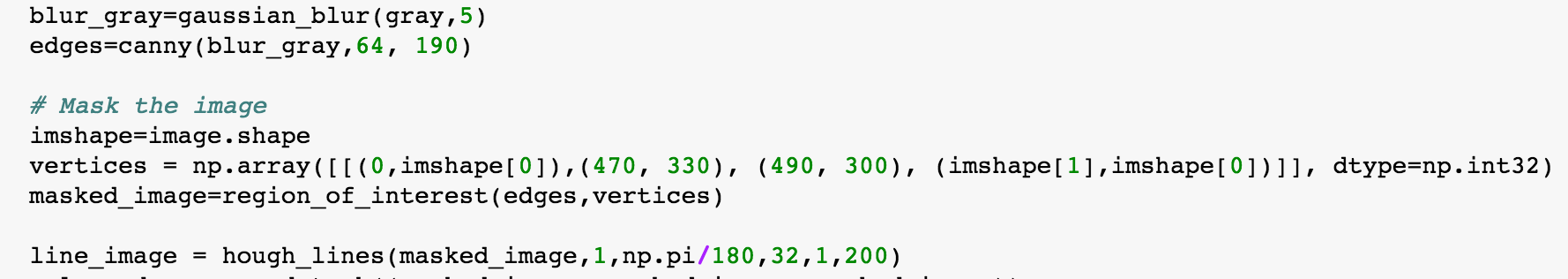


I then define the following functions that help to identify the lane lines.

* Grayscale: returns an image to grayscale.
* Canny: applies canny algorithm with low/high threshold parameters.
* Gaussian\_blur: applies Gaussian smoothing with kernel size.
* Region\_of\_interest: applies an image mask that removes the parts outside the region of interest.
* Draw\_linear\_regression\_line/Draw\_lines: Extrapolate the line segments to map out the full extent of the lane. The left/right lanes are separated by calculating the slope.
* Hough\_lines: applies hough transformation.
* Weighted\_img: combines the original image with the lane lines drawn by the model.



The process\_image function combines the above functions and will be used in the video pipeline. I have tried some combinations of the parameters and the followings appear to give the best result.



The video output is attached as a separate document.