Please find below the steps to execute the program successfully:

* Clone or Download the tensorflow repository: <https://github.com/tensorflow/models>
* Extract the downloaded repository
* Follow the steps in this link to install all the required components to get the object detection module working: <https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/installation.md>
* After successfully installing all required components based on the instructions navigate to /models/research/object\_detection/ and paste the two python files called
* vehicle\_detector.py
* grabscreen.py
* Install the following python packages
* tensorflow or tensorflow-gpu
* jupyter
* matplotlib
* pillow
* lxml
* opencv-python (imported as cv2)
* numpy
* Open the game (GTA 5) -> Settings -> Display -> Change to windowed screen and set screen resolution as 800x600 (CPU) maybe higher if executed on GPU
* Move the game window to the top left corner of the screen
* Open vehicle\_detector.py and run the code
* Wait for the Opencv window to open (might take a while depending on system processing power)
* Project is now ready to be tested
* Please find below the steps to execute the program successfully:  
    
  1. Open Python IDLE and run the “gta\_vehicle\_without\_lanes.py” .  
    
  2. Open the game (GTA SanAndreas) in window mode and keep it open on the left –top corner.  
   Play the game for couple of minutes to generate the training data.  
    
  3. The training data will be saved on the object-detection folder as “training\_data\_XX\_.npy”  
    
  4.After generating the training dataset for balancing data run the “balance\_data\_copy.py”.  
   It will show the results how many left, forward and right moves done.  
    
   5. Install Alexnet with tflearn in windows.  
   Keep the “Alexnet.py” in the object-detection folder.  
    
  6. Run the “train\_model.py” with the numpy file which has generated earlier in the same directory.  
   The file will generate the metrics for trained values.  
    
  7. Open the game again like step 2 and run the file “after\_balancing\_keys.py”. It will show the prediction  
   values as long as the game will play by itself. Keep the running file minimized and observe the game   
   performance.  
     
  Note: In the Game first take a scooter and set it to first person view. The game should play in the daylight as it doesn’t produce accurate results if played at night. Set the screen resolution into 800\*640.