

# MASK-RCNN for CUSTOM CLASSES

For this project, we are going to detect the wagons on a goods train. The Video that we are going to use is available on youtube (<https://www.youtube.com/watch?v=E6b6zHiD6G8>). Copy the video link and go to online video cropper website (<https://ytcutter.com/>). Paste the video link on the website and set the start time as 120 and end time as 230 and download the video. You can directly download the video from the repository as well but this online cropper can be used when you wish to use some other video for your project and I would personally suggest you all try with SIMILAR video footage.

Ensure that you have Mask-RCNN installed and working properly, the methods to install mask-rcnn is well explained in the README.md file in the GitHub repository. (<https://github.com/vedanshdwivedi/Mask-RCNN> ). Also, I would suggest you work on Ubuntu Virtual Machine.

Now, I hope that you have the following things with you

- Mask-RCNN installed
- Video to work upon
- Basic knowledge of Mask-RCNN (The architecture of Mask-RCNN)
- Ubuntu Virtual Machine (optional)
- Knowledge about different classes present in Mask-RCNN by default

Once you are done with all the mentioned points, we can start with the project.

First of all, move the video in the same directory as your jupyter notebook and also create two folders named 'frames' and 'select' in the directory. Now, we need to break the video into frames and store them into the /frames directory. For this, you can use the following piece of code :

```

# Breaking Video to Frames
import cv2
cam = cv2.VideoCapture("coal.mp4")
currentframe = 0
count = 0
select = 0
while(True):
    currentframe+=1
    ret,frame = cam.read()
    if ret:
        name = 'frame' + str(count) + '.jpg'
        d = currentframe%5
        if d==0 :
            cv2.imwrite("frames/"+name, frame)
            count+=1
        d = currentframe%44
        if d==0 :
            name = 'frame' + str(select) + '.jpg'
            cv2.imwrite("select/"+name, frame)
            select+=1
    else:
        break

# Release all space and windows once done
cam.release()
cv2.destroyAllWindows()

```

This piece of code will store every 6th frame in the video. We are storing every 5th frame in order to reduce redundancy as every immediate frame would be very similar. Also, this code saves every 44th image to a separate folder named 'select'. These images we would be using to train our model. Once you run the above piece of code, you'll end up with somewhere around 100 images in the 'select' folder. We have to further divide those images into 'test' and 'train' folders(create 'test' and 'train' folders in 'select' directory) in the ratio 1:3 (Copy every 4th image to the 'test' directory and the remaining images has to be copied to 'train' folder).

Once we have divided the images into test and train directory, we have to annotate the images. For annotating the images, we would use the VGG Image Annotator (<http://www.robots.ox.ac.uk/~vgg/software/via/>) . We can download the latest version of 'Image Annotator' (Version 2). It's a simple HTML file to be opened by any browser. You can also download the annotator from the GitHub Repository.