

## Darknet YOLOv4 using Custom Dataset

My recent project (Darknet Yolov4) is working with over 110,000 images (458 GB) related to the construction domain.

Please kindly go to my darknet Yolov4 Github: <https://github.com/thanthanswe-github/docker-darknet-yolov3-yolov4-training-test>

Additionally, I created a darknet Yolov4 docker file as well.

```
FROM nvidia/cuda:11.1.1-cudnn8-devel-ubuntu18.04

ENV DEBIAN_FRONTEND noninteractive

RUN apt-get update
RUN apt-get -y install nano sed wget git

#Installing OpenCV dependencies
RUN apt-get install -y build-essential cmake git pkg-config libgtk-3-dev \
    libavcodec-dev libavformat-dev libswscale-dev libv4l-dev \
    libxvidcore-dev libx264-dev libjpeg-dev libpng-dev libtiff-dev

#Getting OpenCV
RUN mkdir /opencv_build
WORKDIR /opencv_build
RUN git clone https://github.com/opencv/opencv.git

#Compiling OpenCV
RUN mkdir /opencv_build/opencv/build
WORKDIR /opencv_build/opencv/build
RUN cmake -D CMAKE_BUILD_TYPE=RELEASE \
    -D CMAKE_INSTALL_PREFIX=/usr/local \
    -D INSTALL_C_EXAMPLES=ON \
    -D INSTALL_PYTHON_EXAMPLES=ON \
    -D OPENCV_GENERATE_PKGCONFIG=ON \
    -D BUILD_EXAMPLES=ON ..

RUN make -j$(cat /proc/cpuinfo | grep processor | wc -l)
RUN make install

#Getting YOLOv4/YOLOv3
WORKDIR /
RUN git clone https://github.com/AlexeyAB/darknet.git
WORKDIR /darknet

RUN sed -i 's/GPU=0/GPU=1/' Makefile
RUN sed -i 's/CUDNN=1/CUDNN=1/' Makefile
RUN sed -i 's/OPENCV=0/OPENCV=1/' Makefile

#Compiling Darknet
RUN make -j$(cat /proc/cpuinfo | grep processor | wc -l)

COPY ./dataset ./data

WORKDIR /darknet
RUN wget https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v3_optimal/yolov4.conv.137
RUN wget https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v3_optimal/yolov4.weights
RUN wget https://pjreddie.com/media/files/darknet53.conv.74
RUN wget https://pjreddie.com/media/files/yolov3.weights
```

Here, I would like to display my example images as well as real-time image.

Result Image:



Fig 1: a bus (testing) image.



Fig2: A motorcycle image.



Fig 3: A bicycle (testing) image.



Fig 4: A worker on a ladder (real-time) image.