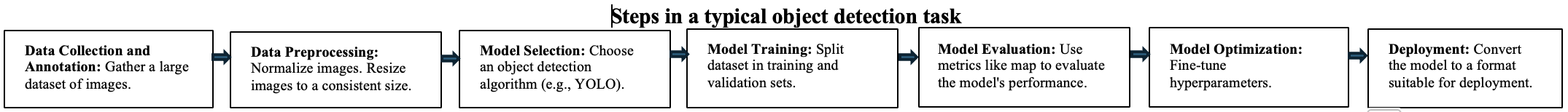
**Cheat Sheet for object detection model and Tasks**

A close-up of a text

Description automatically generatedA white paper with black text

Description automatically generatedA table of text with a few words

Description automatically generated with medium confidence



A diagram of a person

Description automatically generated A diagram of a person in a mask

Description automatically generated A diagram of a multiscale object detection

Description automatically generated A diagram of a network

Description automatically generated

A diagram of a dog and a bicycle

Description automatically generated A page of a computer program

Description automatically generated A white paper with black text

Description automatically generated A screenshot of a computer program

Description automatically generated

**Installation process for TensorFlow–**

First install tensor flow by writing it in a command – pip install tensorflow.

Now, we can start following the steps in the link given below.

<https://wandb.ai/wandb/common-ml-errors/reports/How-to-Correctly-Install-TensorFlow-Object-Detection-API--VmlldzozNTM5ODg#:~:text=%5Cmodels%5Cresearch%5C-,Open%20command%20prompt%20and%20cd%20to%20tensorflow%5Cmodels%5Cresearch.,Object%2DDetection%2DAPI%2D%2DVmlldzozNTM5ODg>

**Links for official documentation –**

<https://github.com/tensorflow/models/tree/master/research/object_detection>

<https://www.tensorflow.org/>

**Installation process for Keras–**

<https://machinelearningmastery.com/how-to-train-an-object-detection-model-with-keras/>

**Links for official documentation –**

<https://keras.io/>

**Installation process for OpenCV–**

<https://docs.opencv.org/4.x/d5/da3/tutorial_ios_install.html>

**Links for official documentation –**

<https://docs.opencv.org/4.x/>

References –

1. Allen, John. "OpenCV Object Detection Tutorial." \*John Allen's Blog\*, https://johnallen.github.io/opencv-object-detection-tutorial/.
2. Patel, Vidit. "R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms." \*Towards Data Science\*, <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e>. (Images from here)
3. "Object Detection Algorithms and Libraries." \*Neptune Blog\*, <https://neptune.ai/blog/object-detection-algorithms-and-libraries>.
4. Brownlee, Jason. "How to Train an Object Detection Model with Keras." \*Machine Learning Mastery\*, <https://machinelearningmastery.com/how-to-train-an-object-detection-model-with-keras/>.
5. "Weights & Biases Team. "How to Correctly Install TensorFlow Object Detection API." \*Weights & Biases Reports\*, <https://wandb.ai/wandb/common-ml-errors/reports/How-to-Correctly-Install-TensorFlow-Object-Detection-API--VmlldzozNTM5ODg>.
6. "Object Detection Guide." \*Hitech Digital Blog\*, <https://www.hitechdigital.com/blog/object-detection-guide>.

**Reflection**

While working on this assignment, we really got the understanding of the main libraries and tools for object detection. Still there are more tools but here we looked at only three tools. We learned about TensorFlow, Keras, and OpenCV, and now we feel confident starting to use them. Object detection is a key task in computer vision, helping systems find and identify objects in images or videos.

From this assignment, we gained a lot of comprehensive knowledge about different object detection techniques. The TensorFlow Object Detection API, figuring out how to use its pre-trained models and detailed docs to build, train, and deploy object detection models. Keras also looks relatively easy to use, especially with libraries like Keras-RetinaNet. When we use Keras with TensorFlow, it makes it perfect for quick experiments and rapid prototyping. OpenCV impressed us with its real-time computer vision capabilities, blending classical methods like Haar Cascades with deep learning models like YOLO and SSD. In the previous assignments when we were working on TensorFlow with PyTorch, our knowledge was not that clear. But after working on this assignment, it is starting to become clear. The cheat sheet we put together during this assignment is going to be very helpful for any future object detection tasks. It has got everything we need for a quick reference, like installation steps, links to official docs, and key concepts for each library, which will save us a ton of time when starting new projects. We’ve also got an overview of different models and how to customize them, which will help us pick the right one for each project. Overall, understanding the strengths and limitations of TensorFlow, Keras, and OpenCV means we can choose the best tool for the job, optimizing performance and accuracy.