

Deep learning is used more and more for vision applications like **image recognition** and **object detection**. With neural networks we are able to extract more and better features from images. Many of these neural networks require a lot of memory and computing power to extract those features. Therefore, many mobile applications exist where the neural network is executed in the cloud. In addition, by developing deep learning applications this way an internet connection is always required. Furthermore, the operations of an existing neural network are often not compatible with the **mobile environment** where the application is executed. A coding language often used for designing and training neural networks is Python. Consequently these neural networks are designed to be executed in a Python environment, while Android studio applications are executed in a Java environment. The purpose of this thesis is to research the **compatibility** between operations of an **existing neural network** in its original environment and a mobile environment. So we can implement complex neural networks on a mobile device with a minimal effect on its accuracy. We will study this implementation for recognition systems and detection systems.

Furthermore we will study the compatibility for the ResNet50, Faster-RCNN and the YOLO architecture. Initially we will study existing neural networks designed and trained in the TensorFlow and PyTorch framework. Each framework has its own method to convert a neural network to a language independent version. Thereafter we demonstrate that TensorFlow supports the most operations for an Android studio implementation. While PyTorch supports less operations, it executes the supported operations faster. PyTorch has to import its not supported operations with an alternate method in Android studio. Eventually all three architectures are executable in Android studio with a minimal effect on the accuracy.