Aerial Land-Use Image Classification
Using Convolutional Neural Networks and Transfer Learning

Alexander Archer, Linda Li, Brody Vogel & Jinghao Yan
Georgetown University

Abstract

This project explores applications of Convolutional Neural Network (CNN) models in classifying aerial land-use images. Models were trained with and without transfer learning to classify images into one of 21 categories. Overall, the methods detailed here achieved relatively good results, with some models attaining over 90% accuracy on the validation set.

1 Introduction

The ability to distinguish between various land-uses using images could be valuable in many fields. Accurate drone imagery classification would have applications ranging anywhere from damage estimation in insurance to national defense measures. Here, the goal is to do just that – classify images into one of 21 land-use categories using deep learning methods. The goals are two-fold: (1) To achieve the best accuracy possible, and (2) To understand what image features each layer and each activation in the models are picking up in the data.

2 Related Work

The first work on this dataset comes from the same group that extracted the images - Yang and Newsam. Working before the popularity of deep learning, Yang and Newsam used complicated Spatial Co-Occurrence Kernels (SCKs) to achieve accuracy scores around 78%. Since then, many other classification methods have been applied to this dataset. In 2017, Scott et al. were able to achieve 97.6% to 98.5% accuracy using deep CNNs and transfer learning with CaffeNet, GoogLeNet, and ResNet.

3 Data

The dataset consists of 2,100 images belonging to one of 21 classes. The images were manually extracted by the University of California, Merced, from the USGS National Map Urban Area Imagery collection. A few sample images are below:

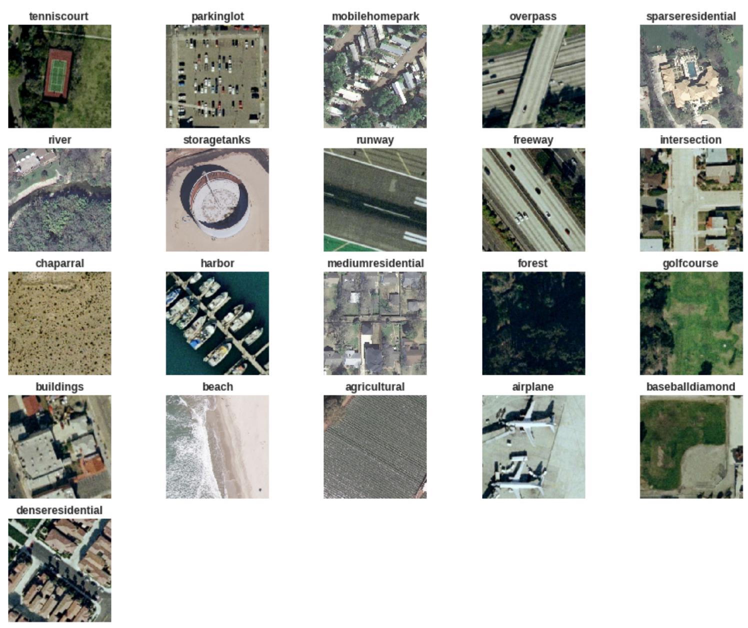

Figure 1: Land-Use Categories

Each RGB image measures 256x256 pixels, with the resolution corresponding to one square foot.

4 Methods

4.1 Training Data

The land-use dataset was divided into a training set with 80% of the images and a validation set with 20% of the images. Real-time data augmentation was used to generate batches of 100 images as the models were being trained. The training images were normalized so the values fell between 0 and 1. They were randomly