Vision-Based Approaches to Air Quality Prediction
(Application Project - Computer Vision)

Robert Schmidt
rschm@stanford.edu

Jake Taylor
jakee417@stanford.edu

1 Motivation

The Air Quality Index (AQI) is used by the U.S. Environmental Protection Agency (EPA) to report air health by aggregating measurements from five major air pollutants. These measurements are taken from ground-based sensors that are at fixed locations within an area. Spatial interpolation is then used to predict the AQI at unobserved locations that are between the sensor sites [1]. Many factors are associated with AQI at the observed sites, such as pollution from industrial areas, nearby wildfires, and local weather patterns. By modeling the observed AQI values with nearby satellite imagery that captures these factors, we hope to improve AQI predictions at the unobserved locations that lack the proximity of a measurement-producing sensor.

2 Method

2.1 Datasets

- **EPA Historic AQI and pollutant data**[1] collected at monitors across the US. The final reported AQI is the maximum of each contributing particle's AQI (in our study area, the maximum pollutant is usually ozone or PM2.5). Data was collected on each pollutant's AQI from 2017 to present, and aggregated to get the daily AQI reading at the sensor level.
- **Planet Satellite Imagery**[2] taken from the PlanetScope 3-band [2] (PSScene3Band) multispectral item type. We selected the "visual" asset type from each PSScene3Band items which give a Red, Blue, Green (RGB) band GeoTIFF raster in 3 meter resolution projected in the UTM projection (WGS84 datum) [Figure 2].

Figure 1: EPA AQI data sites (red squares). An example of an area to interpolate is shown in blue.

[1] EPA pollution data: https://www.epa.gov/outdoor-air-quality-data
[2] PSScene3Band satellite imagery: https://developers.planet.com/docs/data/psscene3band/

2.2 Methodology

- *Study area*: a set of 55 spatially contiguous sensor locations from the greater Los Angeles metropolitan area were chosen to include a variety of topographical backgrounds. Each location was given a square 6 x 6 km buffer centered on the position of the site. To avoid repeated measurements, locations that had overlapping buffers were thinned from the dataset, retaining only unique locations with the best data availability. Interpolation would then occur in areas adjacent to the selected sites [Figure 1].
- *Image acquisition*: Planet's Application Programming Interface (API) was used to find matching satellite imagery strips that strictly contained the buffer of interest [Figure 1]. A query was ran for each site over all matching strips from 2017 to present, producing roughly 7,600 square 6 x 6 kilometer clipped GeoTIFF rasters. To create the final images, these rasters were converted to JPEG and resized to 224 x 224 RGB images [Figure 2]. Filtering for images with a cloud cover factor less than 0.3 and matching to available AQI readings yielded 5,555 final images. This dataset was split into 60% train / 20% validation / 20% test sets for modeling.

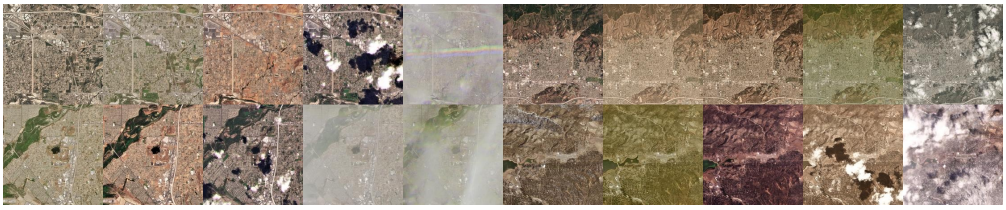

Figure 2: Satellite imagery taken at different sites with varying degrees of image quality.

3 First Modeling Approaches

We observe that it is quite difficult for humans to judge non-extreme AQI from satellite imagery; features that one might pick up on are essentially color-driven cues. Hence, for our first foray into modeling, we turned to approaches at the pixel level driven by machine learning algorithms. Given the breadth of resources available on classification for image tasks, we employed a paradigm in which we classify AQI by its level of health concern[3]. Our relatively small dataset necessitated a reduction of the number of AQI classes to three: good $[0, 50]$, moderate $[51, 100]$, and unhealthy $(100+)$. For each training example $i \in \{1, \ldots, n\}$, we encode the response as $y^{(i)} \in \{0, 1, 2\}$ for good, moderate, and unhealthy days respectively. Lastly, we note that the dataset is class-imbalanced: 55% good, 30% moderate, and 15% unhealthy. To remedy the class imbalance, we employ class-weighted losses wherever possible with the weights defined as follows:

$$w_{y^{(i)}} = \frac{\textit{\# samples in largest class}}{\textit{\# samples in class } y^{(i)}}$$

With this class imbalance, an absolute baseline classifier could classify all days as "good" and still achieve roughly 55% accuracy. Of further note is the question of "goal performance" for our task. Projects that achieved high predictive performance for AQI regression tasks incorporated time-series RNN architectures on large datsets [3] [4]. Due to the sampling constraints of our dataset, we instead focus on "real-time" prediction using only imagery. We compare our task to that addressed by Albert et al. [5], who leveraged transfer learning to achieve 70-80% performance in classifying urban zones with satellite imagery. Accordingly, we consider $70\%+$ accuracy as our goal. Code for all of our models can be found on our project repository: `https://github.com/rschmidt347/vision_AQI_prediction`.

3.1 Pixel-Level Classifiers

Our first baseline models were pixel-level classifiers; here, the training examples are flattened images with associated RGB pixel values as features. Hence, for (224×224)-sized imagery, we note that each training example $x^{(i)} \in \mathbb{R}^{224 \times 224 \times 3}$. While we initially experimented with pixel-level softmax classifiers, we found the most success using support vector machines (SVMs) trained over each binary class pair. In particular, the pixel-level SVMs were implemented using sklearn's SGDClassifier, using an optimal learning rate selected by an internal 10% validation set. We performed a hyperparameter search for the regularization type (among elastic net and pure L2), regularization parameter α, and the SVM kernel; in particular, we focused on the hyperparamter search for the linear SVM given its standout performance. These results are summarized in Table 1.

[3] AQI overview: `https://www.airnow.gov/aqi/aqi-basics/`

Model	Kernel	Penalty	Regularization	Test accuracy
Softmax regression		L2	1.0000	0.6148
SVM	RBF	L2	1.0000	0.6450
SVM	Linear	Elastic Net	0.0001	0.6481
Tuned SVM	Linear	L2	1.0000	0.6602

Table 1: Results for pixel-level models.

3.2 Greyscale Frequency Model

Based upon the hypothesis that AQI primarily interacts with each image through pixel intensity, we attempted a reduction of the feature space of each image to just the frequency at which a greyscale value occurs in an image (in this setting, $x^{(i)} \in \mathbb{R}^{255}$). Although we see a promising separation in average pixel intensity within each class, this model performed more poorly than the pixel-level models; the best softmax model on this feature space only achieved a test performance of 0.522, indiciating that the dramatic reduction in feature space was at the cost of an increased bias [Figure 3]. It appears that taking out the color of the pixels and only considering the frequency of pixel intensity doesn't carry enough signal to correctly classify the images.

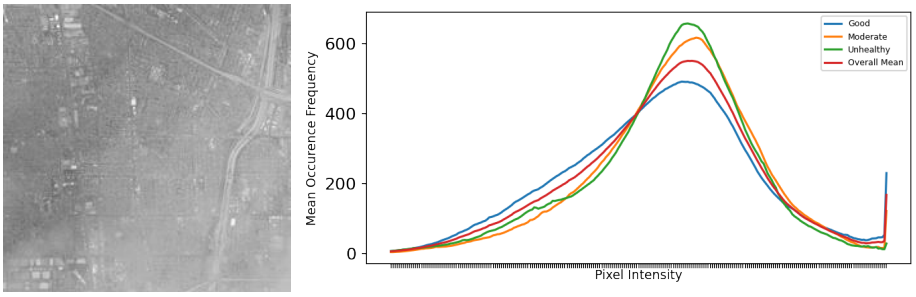

Figure 3: (Left) Grayscale image with varying pixel intensity. (Right) Occurrence Frequency of average pixel intensity by AQI class across all training images.

3.3 Bias-Variance Tradeoff

Given that tactful regularization way key to improving test performance for these initial models, it is worthwhile to further analyze how the bias-variance tradeoff impacted the performance of the best model before delving into more complex approaches. First, as seen in Figure 4, the tuned model with class weights was able to properly identify less-frequent classes given the weighted loss [Figure 4]. The rightmost part of this figure details the impact of training set size on the performance of the tuned SVM. The plot indicates that the baseline SVM was highly variable, and regularizing it reduced this variance, but only up to a point: regularization factors $\alpha > 1$ tended to further degrade test performance. Hence, to boost our test performance, we either needed more data, or a more tune-able model with the ability to leverage pre-trained weights and data augmentation.

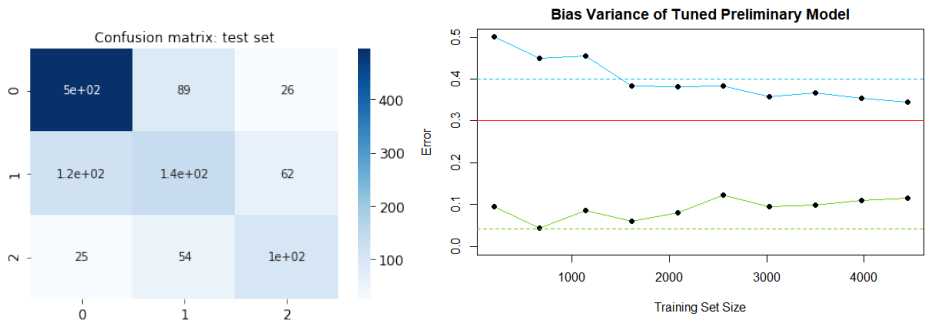

Figure 4: Results for the tuned linear SVM. (Left) Confusion matrix; (Right) Dashed blue/green lines indicate the average test/training error from the default linear SVM; solid blue/green lines show the tuned test/training error as a function of training set size.

4 Convolutional Neural Network

4.1 Transfer Learning

Convolutional neural networks are renowned for their ability to extract information at different levels of abstraction from image datasets. Furthermore, rapid advances in the computer vision have greatly facilitated transfer learning, whereby one can employ the weights from a powerful network pre-trained on a large dataset. To this end, we utilize the keras API [6] for Tensorflow to leverage networks pre-trained on ImageNet. To adapt the pre-trained networks to our task, we remove the "base" neural net's classification layers and replace them with one of the following modules [7]:

1. *Fully connected*: this standard approach stacks fully-connected (FC) layers that feed into the final softmax classification. To avoid over-fitting on such a small dataset, we also employed BatchNorm between Dense layers, and added in dropout and regularization hyperparameters to be tuned as needed. [Figure 5]
2. *Global average pool*: alternatively, we use a global average pooling (GAP) layer that takes as input the output of the base model, and feeds directly into the softmax layer.

Unlike the ImageNet corpus, our satellite imagery dataset "contains much more semantic variation than natural images...there is no 'central' concept that the image is of." [5] In instances of transfer learning with tasks not wholly like that of the base network, it is advisable to train some layers and leave other, earlier layers frozen; this approach leverages the idea that convolutional networks learn more general features in early layers, and more specific features in the later layers.

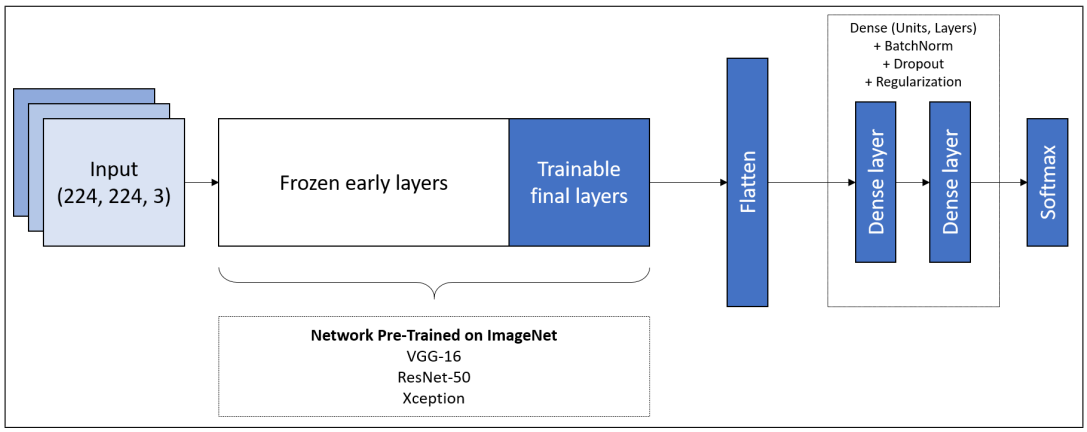

Figure 5: Fully-connected classification module for pre-trained base model.

4.2 Hyperparameter Search

There are numerous hyperparameters and architectures to consider for our transfer learning task.

1. Base network: one of ResNet-50 [8] or Xception [9]
2. Number and size of FC layers, or whether to use global average pooling
3. Dropout and L2 regularization factors
4. Training methodology:
   - Staged training: first train the classification layers only with a frozen base (higher learning rate); then, unfreeze the final layers of the base model and continue training (lower learning rate)
   - Direct training: begin training immediately with a set number of layers unfrozen (typically start with a much lower learning rate)
5. Number of final unfrozen layers in the base network
6. Optimizer: SGD or Adam [10]; exponential, power, [7], or 1-Cycle [11] learning rate scheduling

In spite of computational limitations, we were able to iterate over many combinations of these hyperparameters. Throughout the experiments, we kept a constant batch size of 16 and ran for 100 epochs subject to early stopping with a patience of 10 epochs. Furthermore, we employed data augmentation via random horizontal and vertical reflections during training to bolster our small dataset. The best experimental results are found in Table 2.

Base model	Layers	Dropout	L2 factor	Strategy	Optimizer	Unfrozen layers	Test accuracy
Xception	$128, 64$	0.5	0.0	Staged	Adam	15	0.6256
Xception	GAP			Staged	Adam	25	0.6373
Xception	1024×2	0.5	0.05	Direct	Slow SGD	15	0.6454
Xception	1024×2	0.3	0.0	Staged	Adam	15	0.6580
Xception	1024×2	0.3	0.0	Staged	1-Cycle	32	0.6670

Table 2: Results for selected convolutional neural network experiments.

4.3 Evaluation

Models built on Xception almost always outperformed those built on ResNet-50; moreover, efforts to build a CNN from scratch on the dataset did not even match the performance of the pixel-level models. The success of Xception may be due to its depthwise separable convolution layers, which are able to search for both spatial and cross-channel patterns. Additionally, thoughtful learning rate scheduling was critical for achieving high test set performance. The best-performing Xception model employed 1-Cycle scheduling, with optimal maximum learning rate η_1 found by training the model for one epoch by starting at a low learning rate that increases by 0.5% at each iteration [7]. For our task, we found that $\eta_1 = 0.035$.

5 Discussion

5.1 Unsupervised Analysis

An unsupervised analysis was performed on imagery taken on 9/26/2020 from the blue interpolation grid shown earlier in Figure 1; MODIS [12] Fire Detections from the past 9 days are also shown in the same area. In Figure 6, we predict using the tuned SVM over this grid. The predictions appear to show that (1) although the images were used independently during training, the classifications show some degree of spatial clustering, and (2) the classifications seem to be correlated with the interaction of fire presence, prevailing winds, and topography. In contrast, a similar evaluation using the CNN yielded more conservative estimates across the prediction area: fewer areas were classified as unhealthy.

Figure 6: Unsupervised Analysis: Interpolation Grid overlayed with MODIS Fire Detections from 9/17-9/26.

5.2 Conclusion

Ultimately, the pre-trained CNN was able to produce the best-performing model with test accuracy 0.6670. However, the next-best model is the tuned linear SVM, which has test accuracy 0.6602 and is significantly faster to train. Both models also have comparable performance in correctly identifying each class in spite of the class imbalances. It is important to note the challenges in training the neural net given the somewhat low signal-to-noise ratio in the satellite imagery as well as the varying image quality among training examples.

Overall, we are thoroughly optimistic about future research in this area: with a small dataset and a detailed hyperparameter search, we were able to achieve almost 70% accuracy in real-time AQI prediction. Acquiring more images would both improve prediction and allow us to explore CNN-RNN approaches that model the time dependency of AQI. Furthermore, obtaining images with infrared or topographical bands would provide valuable insight on features not readily apparent from RGB imagery alone. In summary, we see great promise for supplementing traditional AQI prediction algorithms with vision-based approaches on satellite imagery.

Contributions

Both group members contributed equally to the project. Jake initially spearheaded data acquisition, while Robert developed the primary CNN model architecture. Each team member performed model optimization and analysis.

References

[1] D. Mintz, "Technical assistance document for the reporting of daily air quality - the air quality index (AQI)," May 2016. [Online].

[2] P. Team, "Planet application program interface: In space for life on earth," 2018–.

[3] R. Navares and J. L. Aznarte, "Predicting air quality with deep learning lstm: Towards comprehensive models," *Ecological Informatics*, vol. 55, p. 101019, 2020.

[4] Kök, M. U. Şimşek, and S. Özdemir, "A deep learning model for air quality prediction in smart cities," in *2017 IEEE International Conference on Big Data (Big Data)*, pp. 1983–1990, 2017.

[5] A. Albert, J. Kaur, and M. C. González, "Using convolutional networks and satellite imagery to identify patterns in urban environments at a large scale," *CoRR*, vol. abs/1704.02965, 2017.

[6] F. Chollet *et al.*, "Keras." `https://keras.io`, 2015.

[7] A. Geron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, Inc., 2019.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

[9] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2017.

[10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

[11] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay," 2018.

[12] G. Technology and A. Center, "Modis fire detection." `https://fsapps.nwcg.gov/googleearth.php`, 2020.

[13] A. Reff, D. Mintz, and L. Naess, "The 0_3 NowCast: U.S. EPA's method for characterizing and communicating current air quality." [Online].

[14] F. Chollet, "Building powerful image classification models using very little data."