Leveraging Convolutional Neural Networks to Predict COVID Outbreaks of Urban US Areas Final Report

Computer Vision

Kevin Borst
Department of Computer Science
Stanford University
kborst2@stanford.edu

Nicholas Midler
Department of Biomedical Informatics
Stanford University
midler@stanford.edu

Abstract

Covid-19 has wreaked havoc on the US, causing mass loss of life. Predicting outbreaks early in their emergence offers potential to mitigate the spread of the disease. Satellite images offer a way to measure human activity, potentially offering a method to find areas where increased human activity could give rise to Covid outbreaks. We implemented an image masking-based strategy that attempts to seek and eliminate all non-human activity in satellite images, allowing convolutional neural networks models to focus on hallmarks of activity to better gauge social interactions and predict Covid outbreaks.

1 Introduction

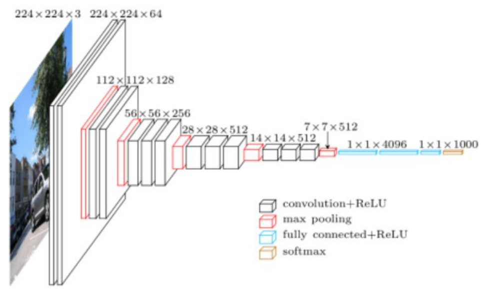
Covid-19 has caused in excess of one million deaths in the US, and a key to managing the disease and future pandemics is being able to detect outbreaks and rapidly implement public health interventions. The highly communicable nature of Covid-19 makes fast intervention paramount to stopping the spread of the virus. Satellite images of urban areas offer a potential method to predict Covid-19 outbreaks. In fact, high resolution satellite images can detect changes in human activity that might be informative for changes in COVID cases. This project proposes to use satellite images of urban areas to classify urban scenes into outbreak or non-outbreak categories. The use of satellite imagery to predict Covid-19 outbreaks could provide an early warning sign for Covid-19 outbreaks across the country, catching blind spots missed by shortfalls in testing supplies, or even pre-empting testing data as symptoms can take up to a week to incubate. [1] The rationale for the project is that Covid-19 outbreaks will occur in response to increased human activity. This same activity should register on satellite images as increased traffic, signs of movement, and other identifiable hallmarks of human activity. The problem represents an interesting subset of image detection given the sparsity of human activity in many satellite image of urban areas. Satellite images are dominated by permanent structures that remain unchanged during the pandemic. To capture this sparsity data processing steps were added to isolate the changing features of satellite images, and the CNN architecture was modified to account for pixel sparsity. These steps are discussed in the following sections.

2 Related Work

While prior research uses pollution data obtained from satellites, or human activity monitoring to predict Covid-19 case levels, no approach has used satellite images trained on human activity. [2] To help us come up with an idea for a model, we read up on the papers proposing ResNet and VGG-16, which are two model architectures that do image classification and thus relate to the basis of our project. For image classification, ResNet reformulates its layers as learning residual functions that reference its layer inputs, instead of just learning functions that have not been referenced before.

Preprint. Under review.

The layers of this Net are thus easier to optimize on and will increase in accuracy from significantly increased depth. [3] This was helpful to read about as it directly influenced our choices to try and have a more complex and deep model for our image classification. VGG-16 similarly works off of convolutional network depth being helpful in network accuracy for large-scale image recognition. It is based around increased depth using very small (3 by 3) convolution filters showing that a large improvement can be seen in the task. [4] VGG-16's model architecture, which we used as our baseline, can be seen below.



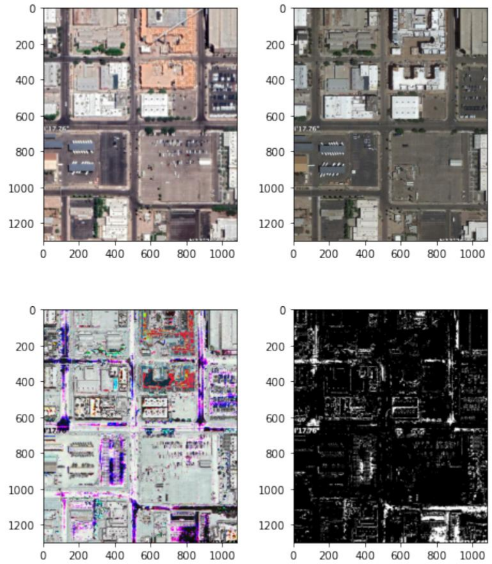
The combining of small kernels with very deep networks was further helpful to us, as we decided to use similarly small filters. Prior work has also used CNN to attempt and identify areas of change for focus. [5] While promising, we believe our unsupervised approach may offer better results over satellite images taken over short time periods (weeks and months instead of years), since the changes will be less pronounced.

3 Dataset and Features

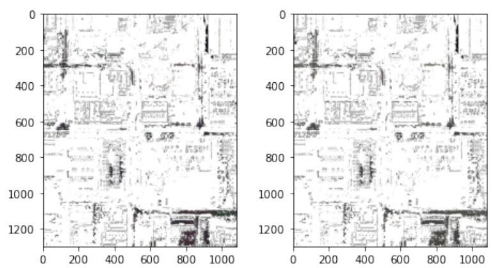
We created a dataset with images taken from Google Earth Pro. We developed a train dataset of images taken from downtown Phoenix. Phoenix was chosen as it's a large metropolitan area with low cloud cover. Each image was taken at an altitude of 1923m and covered a rectangular land area of approximately 333.36 meters by 400 meters. This generated images of approximately 1162 by 1394 pixels. Images that varied from the 1162 by 1394 pixels were resized using the cv2 library resize function. The dataset consisted of 100 locations taken at various time points for a total of 400 images. All 100 locations include pictures from March 2022, March 2021 and May 2021. Due to incomplete satellite coverage half of the scenes include an image taken on July 2020 while the other half include an image taken on December 2020. Each image was classified into either low or high Covid levels depending on the number of daily Covid cases in Maricopa county during the month the photo was taken. The 2020 census tallied Phoenix's population at 1,608,139 and Maricopa county at 4,420,568. [2] Given the large share of Maricopa counties population that resides in Phoenix, we believe that county level case counts work as an effective proxy for Phoenix infections. To discern if the hallmarks of human activity generalize beyond Phoenix, we developed a test set of 10 similarly sized locations from Manhattan, New York City. Each scene had two images taken on December 2020, a high point of the pandemic, and May 2021, a low point of the pandemic, for a total of 20 images in test set.

A visual inspection of two of the images for a random location in Phoenix shows that certain hallmarks of human activity can be discerned. The photo on the left is taken during the lowpoint of Covid-19 in Summer 2021 while the one on the right is at a height of the pandemic. Interestingly, there appears to be more human activity (busier parking lots, more street activity) in the earlier photo, suggesting human activity may be a lagging indicator of infection rates. The first two images shown on the next page are the images we refer to here.

In these photos the parking lots and roads in the picture are what we want to feed into our model as input features. In addition to a version of the dataset that simply fed in the raw images, we also had a version that focused our model's attention on these features by subtracting the various date images' pixel values. This meant we ran our model twice. The resultant image displays areas of disparity between the two pictures. While many of these areas were due to color changes between the pictures, the purple areas corresponded to the parking lots and streets in the photos where varying levels of human activity are observed. The second two images on the next page are the images we refer to.



A further binary mask was applied to the image that set to zero all values not within a specific threshold observed in the parking lots and streets. This binary mask was then applied to the original images, leading to sparse pictures that were then fed into our CNN model.



The input images are then fed into the CNN after the mask is applied. The picture on the left corresponds to the Summer 2021 image and the image on the right corresponds to the highpoint of the pandemic image.

4 Methods

Our approach employed two complementary innovations in pursuit of improved model performance. The first concerned a data processing step described above that applies a mask to satellite images that removes all stationary features to focus on human activity. This creates sparse input pixel matrices to the model. While a manual review of masked images shows that some features are preserved, they exist in a sea of zero values. The majority of pixel values are zero. We attempted to address the sparsity of the input images by using two max pooling layers to eliminate many of the zero values and focus model attention on the remaining features. We initially employed one max pool layer with a larger kernel, but achieved better performance by splitting the pooling into two relatively smaller layers.

In total, the model consisted of:

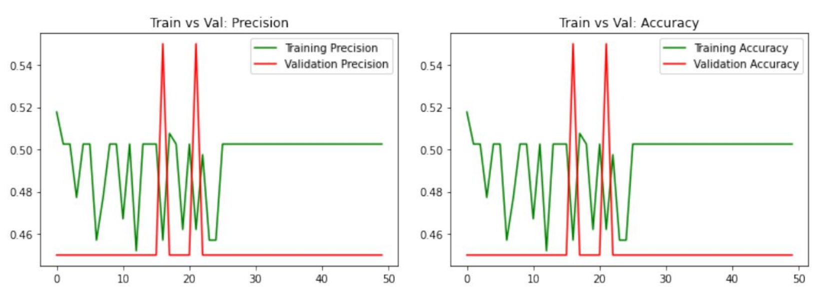
- **Max Pool** Kernel: 4x4 Stride: None
- **Max Pool** Kernel: 3x3 Stride: None

- **Conv Layer** Filter: 64, Kernel: 7x7, strides: None, padding: Valid
- **Conv Layer** Filter: 128, Kernel: 5x5, strides: None, padding: Valid
- **Max Pool** Kernel: 3 Stride: None
- **Conv Layer** Filter: 256, Kernel: 3x3, strides: None, padding: Valid
- **Conv Layer** Filter: 256, Kernel: 3x3, strides: None, padding: Valid
- **Conv Layer** Filter: 512, Kernel: 3x3, strides: None, padding: Valid
- **Conv Layer** Filter: 512, Kernel: 3x3, strides: None, padding: Valid
- **Conv Layer** Filter: 512, Kernel: 3x3, strides: None, padding: Valid
- **Conv Layer** Filter: 512, Kernel: 3x3, strides: None, padding: Valid
- **Max Pool** Kernel: 2x2 Stride: None
- **Max Pool** Kernel: 2x2 Stride: None
- **FC** Nodes: 64
- **FC** Nodes: 32
- **FC** Nodes: 2

The model was trained using batch gradient descent with Adam optimization with a learning rate of 0.001 and beta1=0.9 and beta2=0.999 using KLDivergence loss. We also developed VGG-16, a validated CNN model for image recognition to use as a benchmark.

5 Experiments/Results/Discussion

The results for VGG-16 can be seen below on the non-modified images.

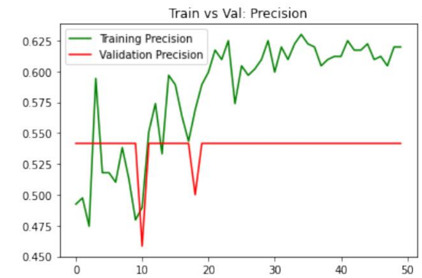


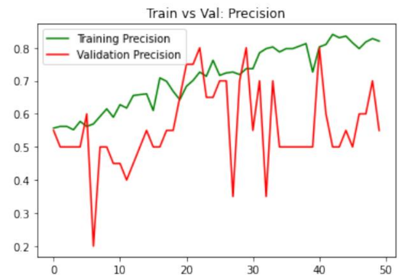
Our VGG-16 model ultimately struggled with the task we set for it. It pretty much stayed around 50% accuracy or precision (our two primary metrics) on either our training or validation sets for the entirety of us running it. But since it was a baseline, this was not a big deal to us. We were able to perform much better with our custom model, so VGG-16 was successful in serving as a benchmark for our model's performance.

Going off the metnion of our metrics, we chose to prioritize precision in our analysis since the cost for missing outbreaks is likely higher than a false positive. It is also better to warn of an outbreak when there isn't going to be one than to not warn of an outbreak when there is going to be one.

The results for our custom model can be seen on the next page.

The left figure shows the precision for the masked input images and the right figure shows the precision for the raw precision. Our model trained on the masked images performed slightly worse than the model using the raw input images. This result argues against our mask-based approach. Deep learning models are infamous for performing well with large amounts of data. Our mask approach cut against this wisdom by limiting the input data to the CNN model by masking much of the model's features.





6 Conclusion/Future Work

Ultimately, our model was rather effective in classifying images correctly with their relative level of Covid cases. Interestingly enough, the masks that we applied to the images thinking that they would highlight human activity and help the model did the opposite of helping. It turns out the raw images are able to be descriptive enough on their own and the right kind of model is able to properly analyze them. VGG-16 turned out to not be such a kind of model, but we were able to find that such a kind of model does exist through our own model, which is promising because it means real-life usage of convolutional neural networks to track potential disease outbreak is possible and should be explored. If we had more resources and chose to pursue this project further down the line, it might help to have a more professional dataset. Our request for satellite image-based data from online services like Planet never bore any fruit and hindered us. Had we gotten this Planet data, we could have done a more thorough job of feature extraction and used these features with better computational power to really test the limits of what features of an image could hint about potential Covid outbreaks.

7 Contributions

Kevin gathered half of the training dataset and the New York City test dataset. He also implemented VGG-16 for benchmark test performance and did the research on related work.

Nicholas gathered half of the training dataset and implemented the custom model with the max pooling model for sparse input matrices. He also created the masks for input images.

8 Code

A link to a Github repository containing the code for this project can be found by copy and pasting this link:
https://github.com/KWBorst/CS230_COVID

9 References

1. Elias, Christelle et al. "The incubation period of COVID-19: A meta-analysis." International journal of infectious diseases : IJID : official publication of the International Society for Infectious Diseases vol. 104 (2021): 708-710. doi:10.1016/j.ijid.2021.01.069
2. U.S. Census Bureau quickfacts: Phoenix City, Arizona. (n.d.). Retrieved June 1, 2022, from https://www.census.gov/quickfacts/phoenixcityarizona .
3. He et al. Deep Residual Learning for Image Recognition. arXiv; 2015. arXiv.1512.03385
4. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv; 2014. arXiv.1409.1556.
5. de Joung and Bosman. Unsupervised Change Detection in Satellite Images Using Convolutional Neural Networks. arXiv:1812.05815