

Deep Learning for Conservation: Classifying Drone Images from Protected Lands

Jenica Andersen



Image from: <https://www.kaggle.com/datasets/irvingvasquez/cactus-aerial-photos>

Mexico seizes ship carrying 68,000 tonnes of illegal iron

⌚ 1 May 2014

“Police confiscated an iron ore mining operation belonging to a drugs cartel near Lazaro Cardenas...”



Mexican crime gangs branching into illegal logging, researchers warn

“..Lack of enforcement due to scant resources and corruption.”

VIGIA project

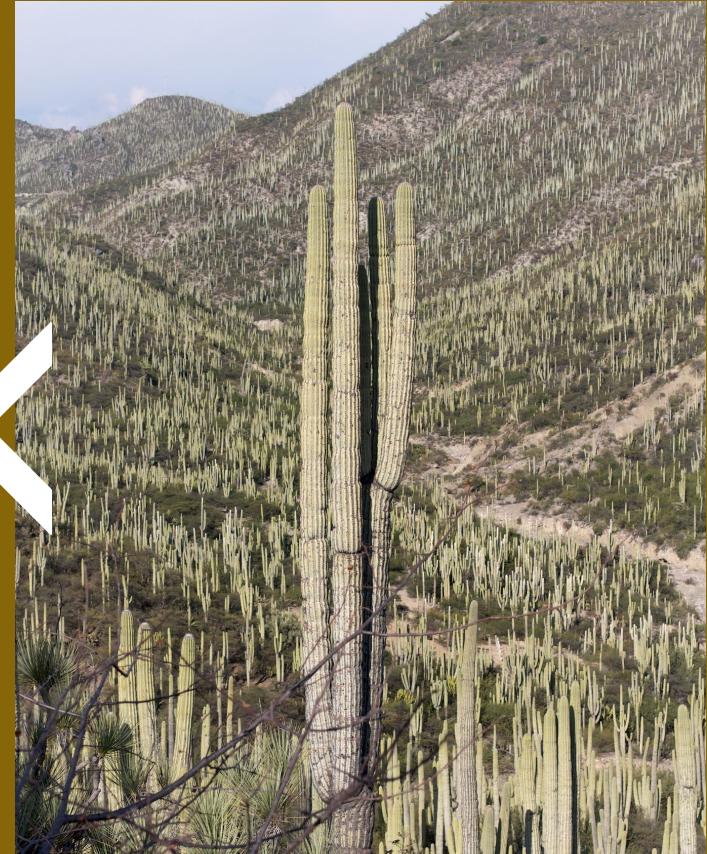
Mexico is rich in biodiversity; however, human activity is degrading these natural areas.

Government efforts to declare natural areas “protected” and surveil them have, so far, been insufficient.

Human monitoring is expensive, limited, and in some areas prone to corruption

Can we develop a system for automatic surveillance of natural areas using artificial intelligence?

–<https://jivg.org/research-projects/vigia/>



<https://benjaminblonder.org/2015/01/08/guess-the-cactus/>



Needed Surveillance:

- Detect smoke and fire
- Disrupt illegal hunting
- Prevent illegal logging or mining
- Identify cars/other human activity in protected areas

Research Question:

- Can deep learning techniques, applied to small-sized, low-resolution surveillance/drone images, offer reliable image classification for automated monitoring?

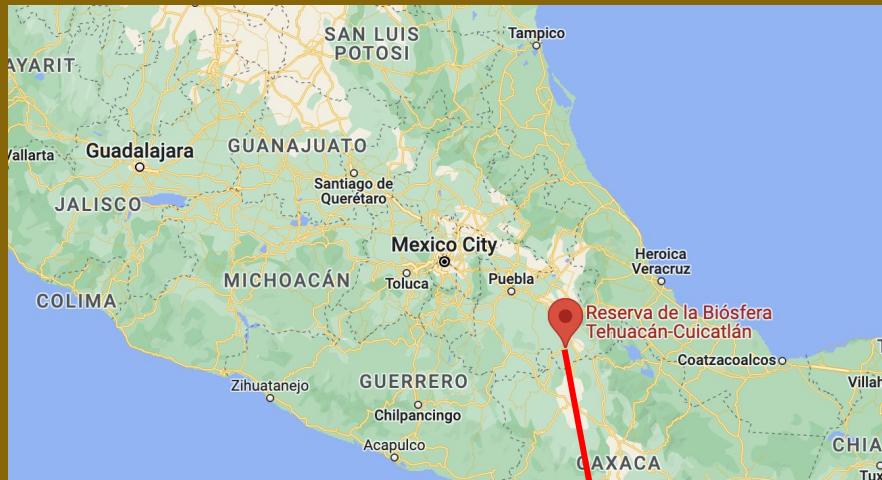


Image from: <https://www.kaggle.com/datasets/irvingvasquez/cactus-aerial-photos>

The Data:

Kaggle dataset

- Tehuacan-Cuicatlan Biosphere Reserve
- Drone aerial images



The Data:

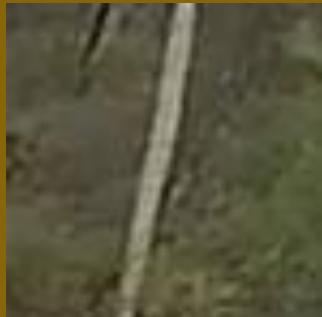
Cactus Aerial Photos

Columnar Cactus (*Neobuxbaumia tetetzo*)

- 16,138 Cactus images
- 5,366 Non-cactus images
- 80% training, 20% test data

Start with Transfer Learning

MobileNet (trained on ImageNet)

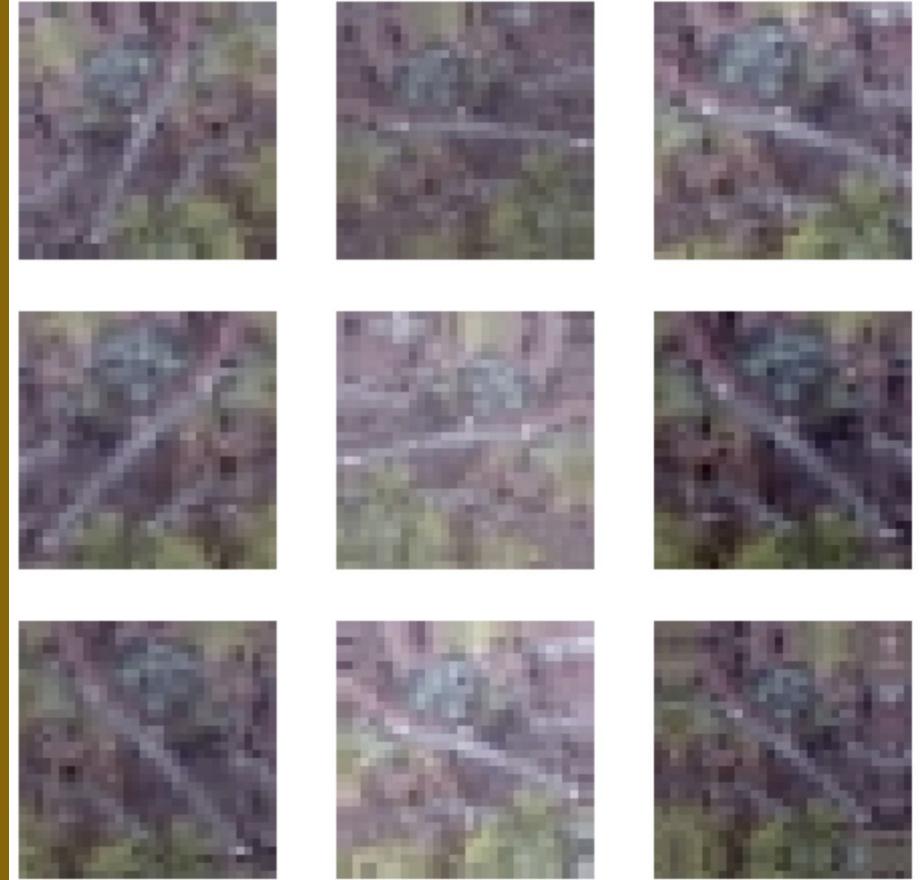


- window_screen: 0.743
- shower_curtain: 0.032
- prison: 0.021
- window_shade: 0.020
- crossword_puzzle: 0.018

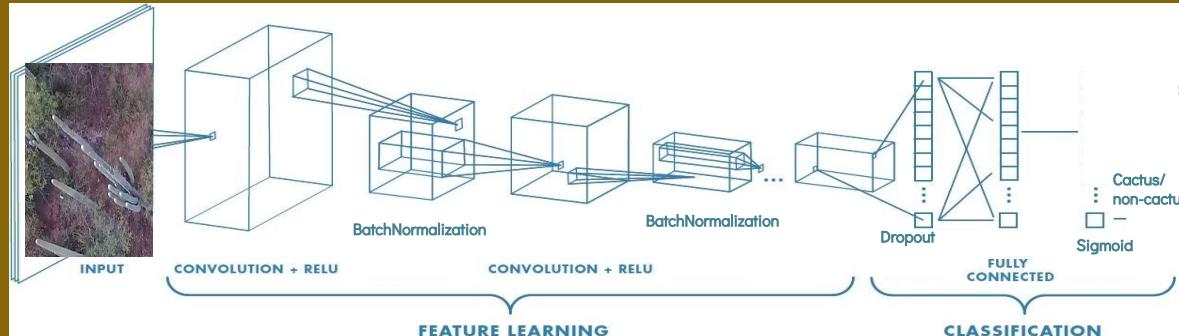
Not so helpful... proceed with model building

Data Augmentation →

- Random horizontal flip
- Random rotation 0.1
- Random Zoom -0.2-0.3
- Random Contrast 0.2
- Random Brightness 0.2



Model Architecture



- Adam Optimizer
- Scaled data

02

Conv2D

11

Batch

Normalization

09

Separable
Conv2D

02

Pooling

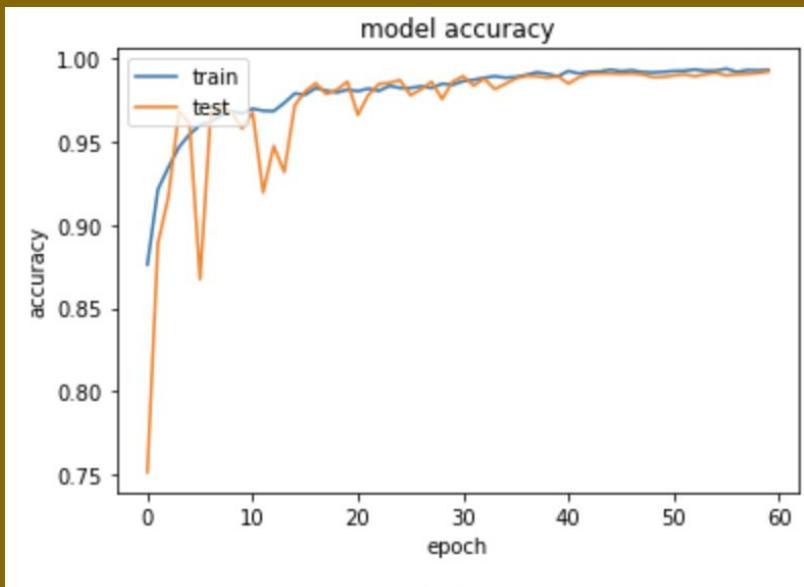
02

Dropout

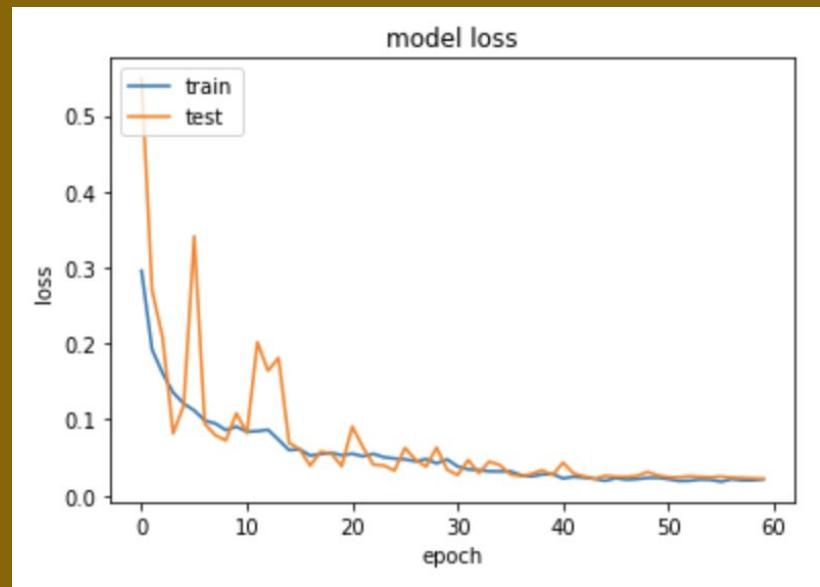
02

Dense Layer

Results: Model Performance



Training Accuracy: 0.993
Test Accuracy: 0.992



Training Loss: 0.021
Test Loss: 0.022

How does the model perform on other images?



Model: “These are NOT cactus”



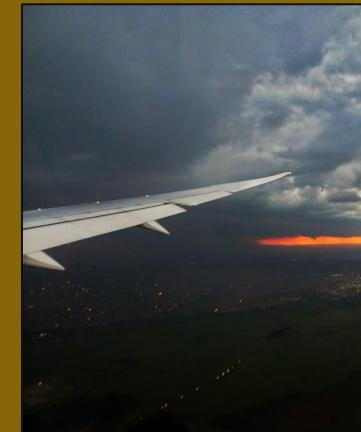
CNN: 1.13% likely a cactus
98.87% likely non-cactus

MobileNet:
75% megalith
3% alp
2% stone_wall
1% beacon
1% worm_fence



CNN: 0.34% likely a cactus
99.65% likely non-cactus

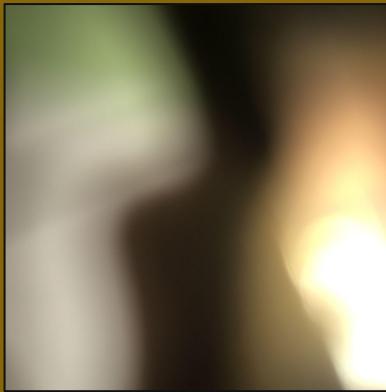
MobileNet:
22% ant
10% bee
3% sulphur_butterfly
2% fly
2% yellow_lady's_slipper



CNN: 0.00% likely a cactus
100% likely non-cactus

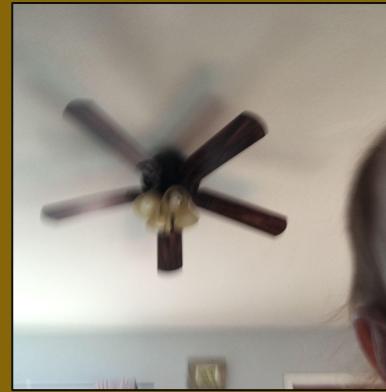
MobileNet:
76% wing
6% volcano
1% airliner
1% missile
1% aircraft carrier

Model: “*Maybe* these are cacti?”



CNN: 9.65% likely a cactus
90.35% likely non-cactus

MobileNet:
68% matchstick
4% spotlight
4% lighter
2% volcano
2% candle



CNN: 10.87% likely a cactus
89.13% likely non-cactus

MobileNet:
22% wall clock
10% analog clock
3% tripod
2% matchstick
2% tarantula

Horned owl



volcano



Implications for misclassification

- Flag images that should be cactus but aren't
- Human review flagged images
- False negatives = No major implications
- Misclassification of negative class as positive could reduce effectiveness of security system
- False positives = security threat

Precision is a better metric than accuracy

Next Steps:



Address
Overfitting



Domain
Adaptation



Residual
layer



Transfer
Learning

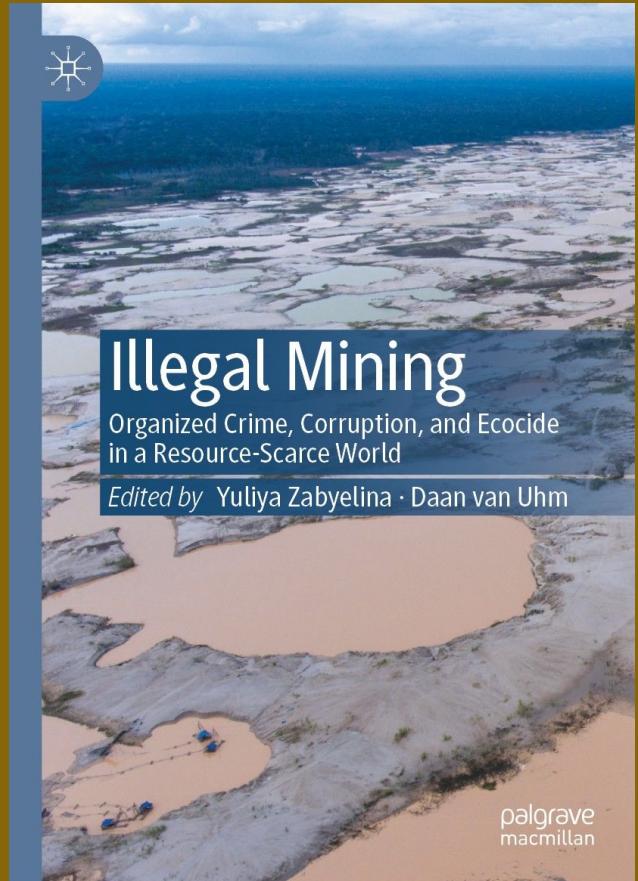
Multiclass
Model



Video Analysis

Summary

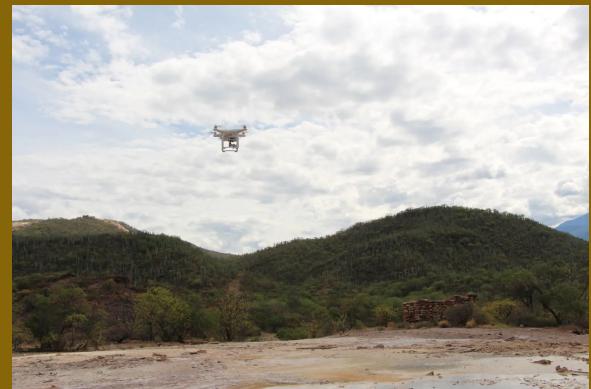
- Yes CNN can offer high performing models on low resolution images
- Transfer learning is not very helpful in this case
- Many models, trained on this data, perform well; fine-tuning improves marginally
- Need for this kind of solution growing
 - fire alert systems
 - security surveillance
 - environmental monitoring



List of references

Used for project info and help with models:

- <https://jivg.org/research-projects/vigia/>
- https://keras.io/examples/vision/image_classification_from_scratch/
- <https://www.kaggle.com/code/twhitehurst3/aerial-cactus-keras-cnn-95-accuracy>
- <https://ladvien.com/lego-deep-learning-classifier-cnn/>



<https://jivg.org/research-projects/vigia/>

Thank you!

Questions?

jenica.andersen@gmail.com



CREDITS: This presentation template was created by [Slidesgo](#), including icons by [Flaticon](#), infographics & images by [Freepik](#)

