Group 1

Kyle Wright, Dillon Marquard, Kirankumar Ashokkumar, Ahmaree Sanders, Chloe Engel

Introduction

- Problem
- Image Data
- Footprint Algorithm
- Filters
- Generating Training and Testing Data
- Spatial Analysis
- Training Methodology
- Models
- Moving Forward

Problem

Image Data - Difference Image

- Difference images provided from Zwicky Transient Facility (ZTF)
- Difference image: image showing the changes occured over set amount of exposure to a capture
- Synthetic asteroids injected into image
- 10 npz files with 100 images, 20 asteroids in each image

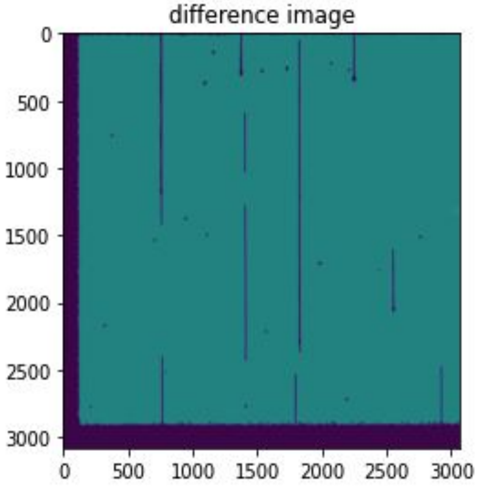


Image Data - Single Asteroid

- Each asteroid is approximately 5x5px
- Images provide: magnitude, length, angle, and location of each asteroid
- Location is represented by coordinates
- 100x100px box is created surrounding the asteroid located at the center

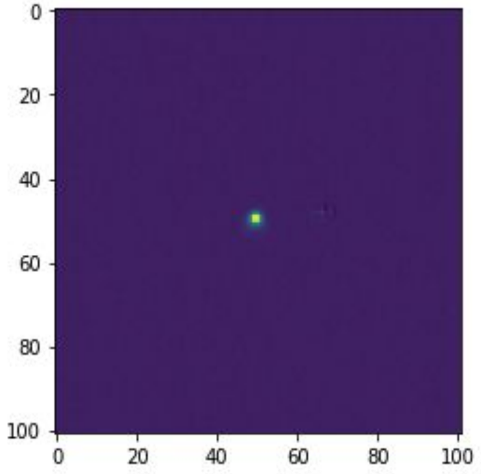


Image Data

- Question: How do the asteroid pixel values compare to the entire image?
- Outcome: asteroids could be distinguished by pixel value, classified to have a greater value than other image pixels

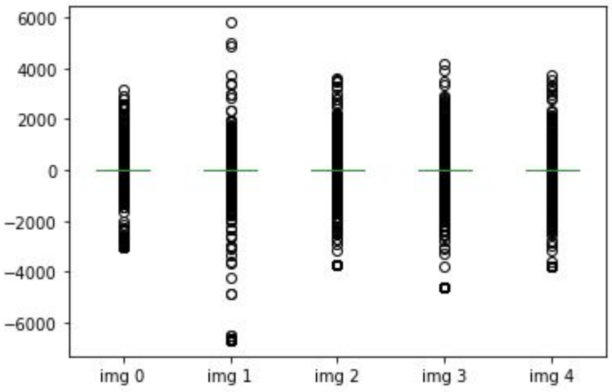


Figure a: entire image pixels

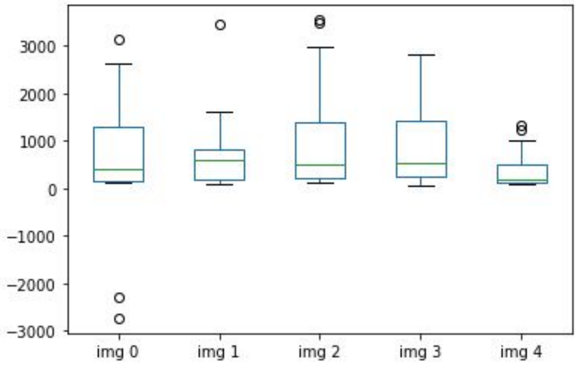
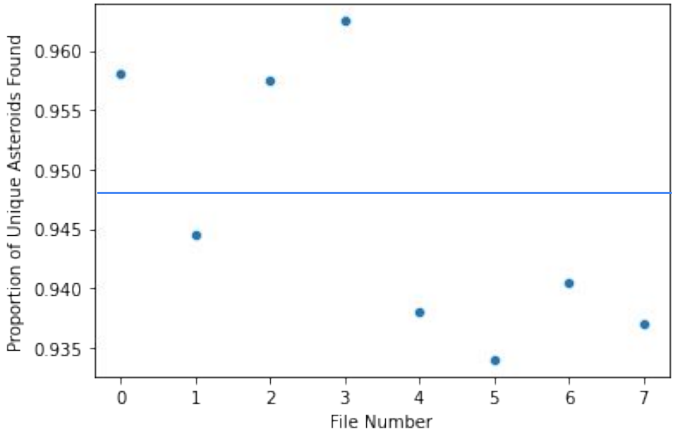
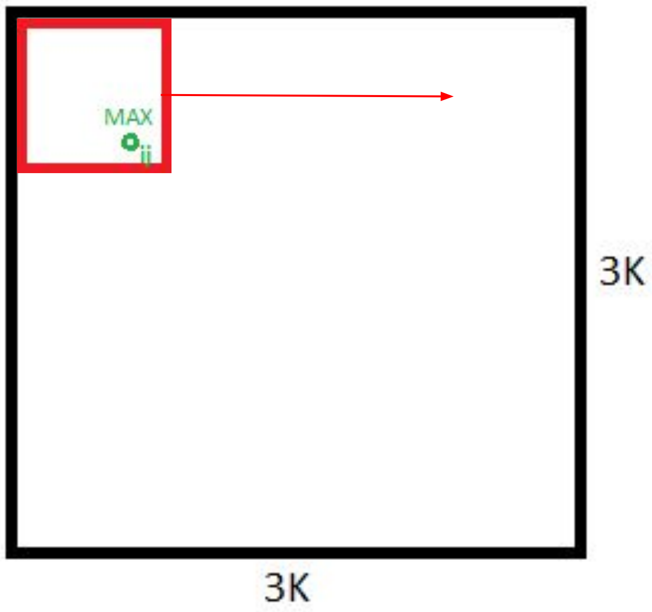


Figure b: asteroid center pixels

Footprint Algorithm

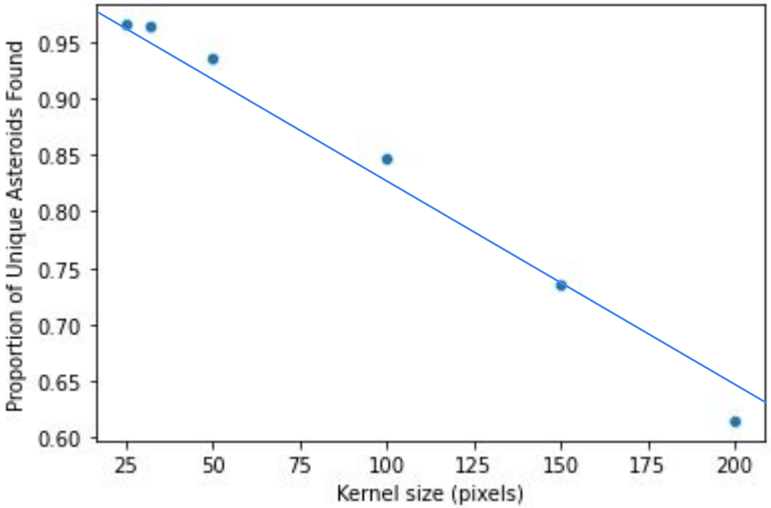
- Break the image into 50x50px subimages
- Take the max pixel coordinate of each subimage
- Finds ~95% of the asteroids





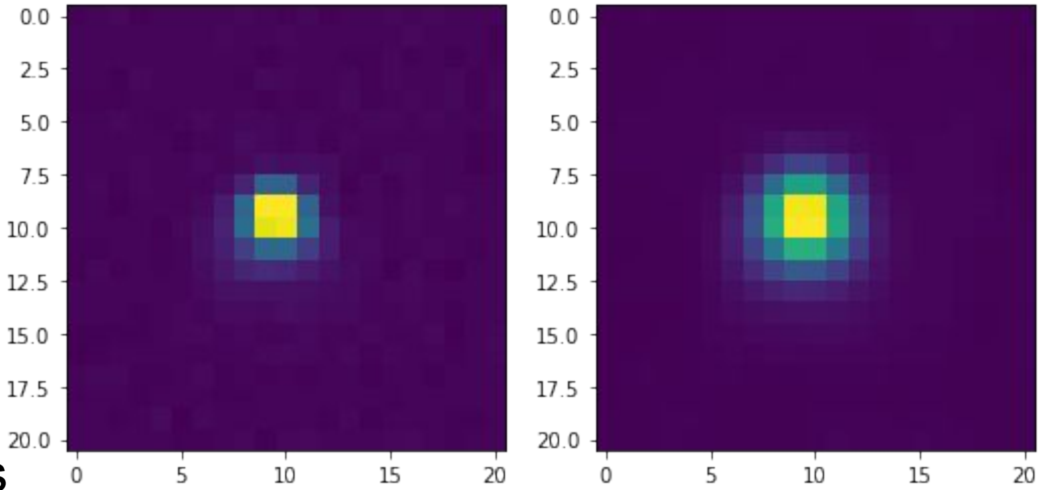
Footprint Algorithm cont.

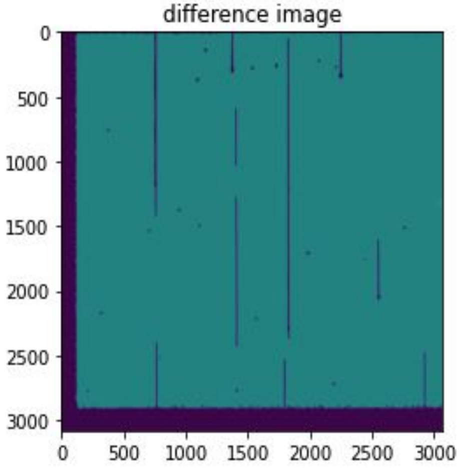
- Smaller kernel finds more asteroids
- Smaller kernel takes longer
- 10.47 images/second with 50x50px
- If asteroids are too close/blending then this method is very ineffective



Filters

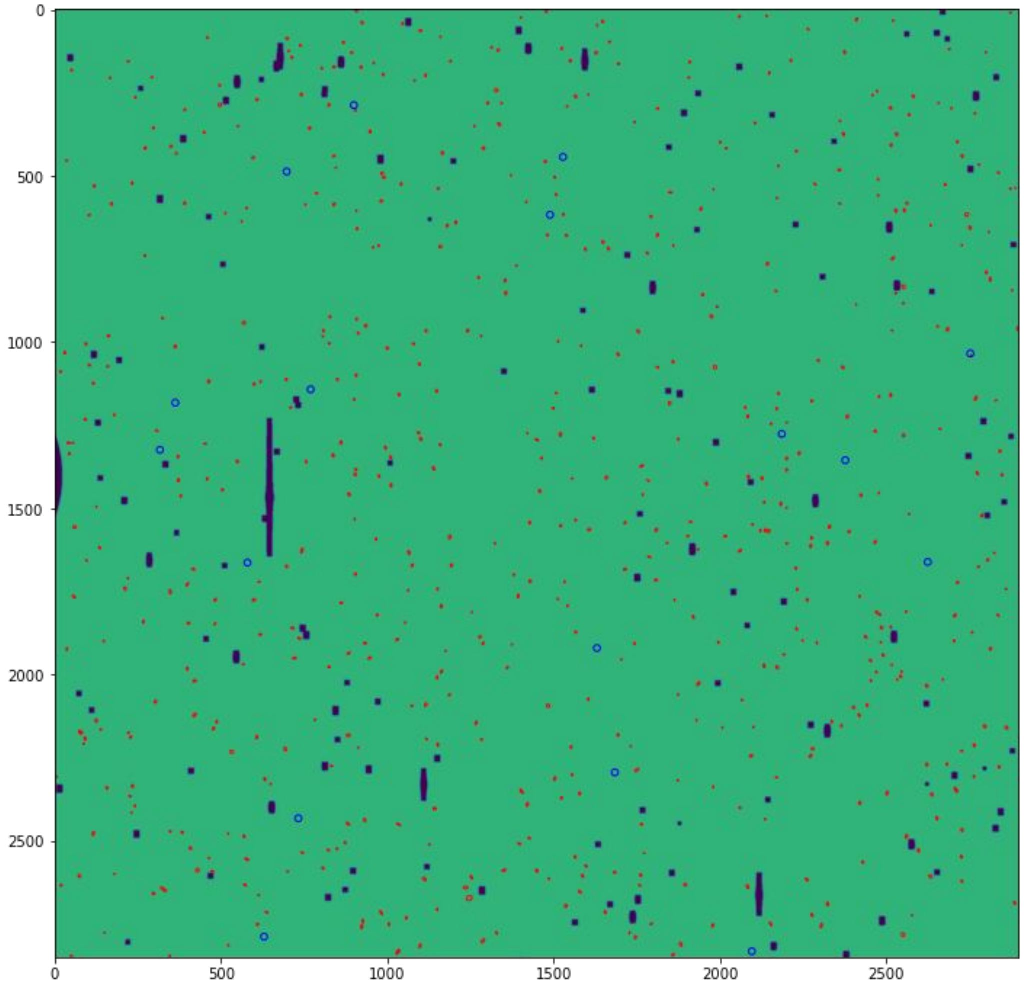
- Gaussian Filter, σ=1
- 2.91 images/second
- Less noise after filtering
- Bad for high magnitude stars
- Other potential filters
  - Source Extractor ~88% of asteroids
  - The background subtraction was particularly ineffective because of the bottom and left edges





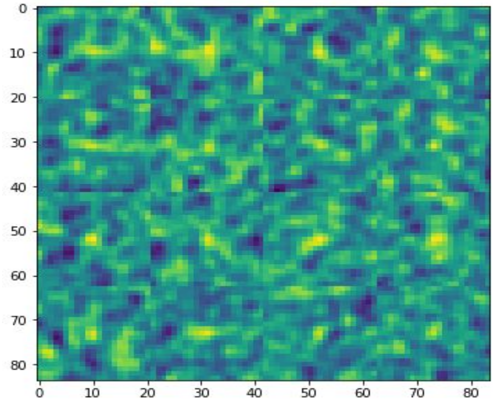
Filters cont.

- A preliminary test of Source Extractor makes it seem worth further investigating for better parameters suited for this problem
- Cutting off the bottom and left regions improved performance

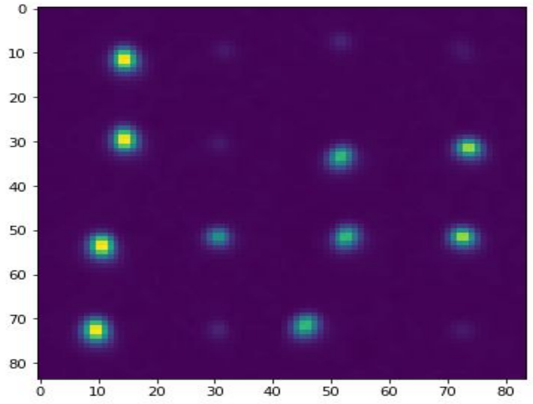


Data Visualization

- Positive class grid that the asteroids vary in magnitude and the dimmer asteroids are hard to discern from background noise.
- We can see from the Negative class grid that the noise is fairly different from an asteroid.
- In that regard, it is justified to attempt to differentiate these classes with a CNN.



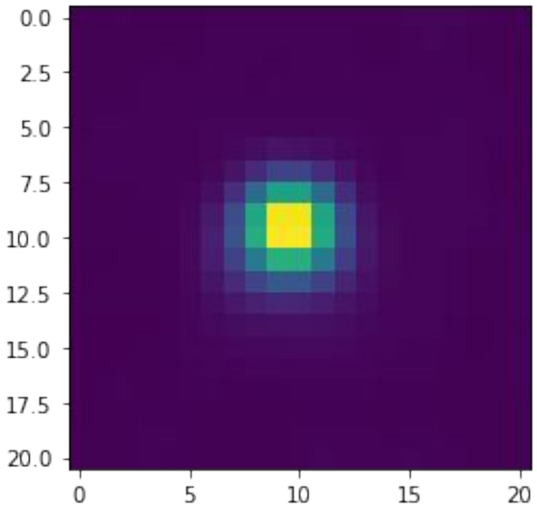
(a) Negative Class Grid



(b) Positive Class Grid

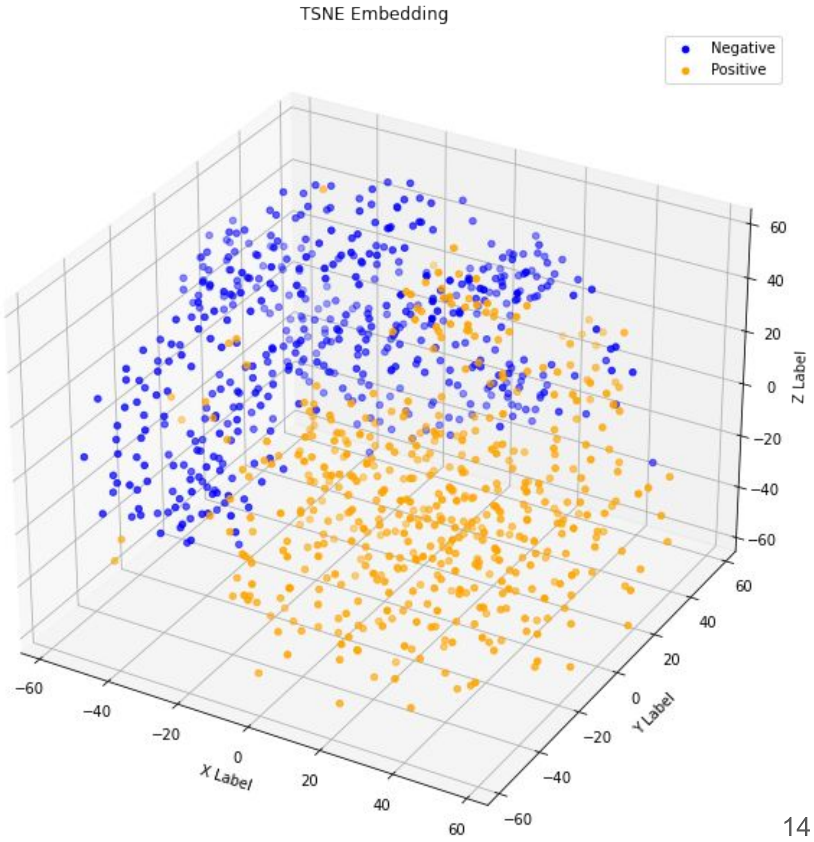
Training Testing Data

- Subimage cutout rate 1.49 images/second
- The entire image is filtered before cutting
- Cutouts are 21x21px centered around the coordinates found by our footprint algorithm
  - Smaller worked better for galstar problem
- Downsides:
  - Model won’t be as robust when its not centered relatively close to the center



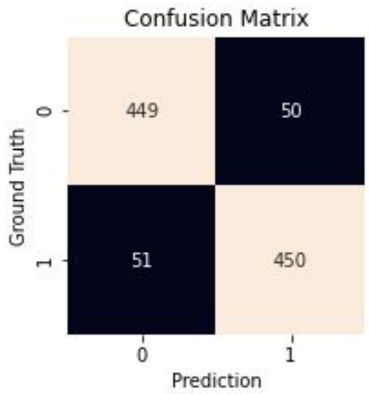
TSNE (3-components)

- 3D embedding
- Reduce the feature dimensionality
- Clear boundary between classes
- Computationally expensive as we increase image size



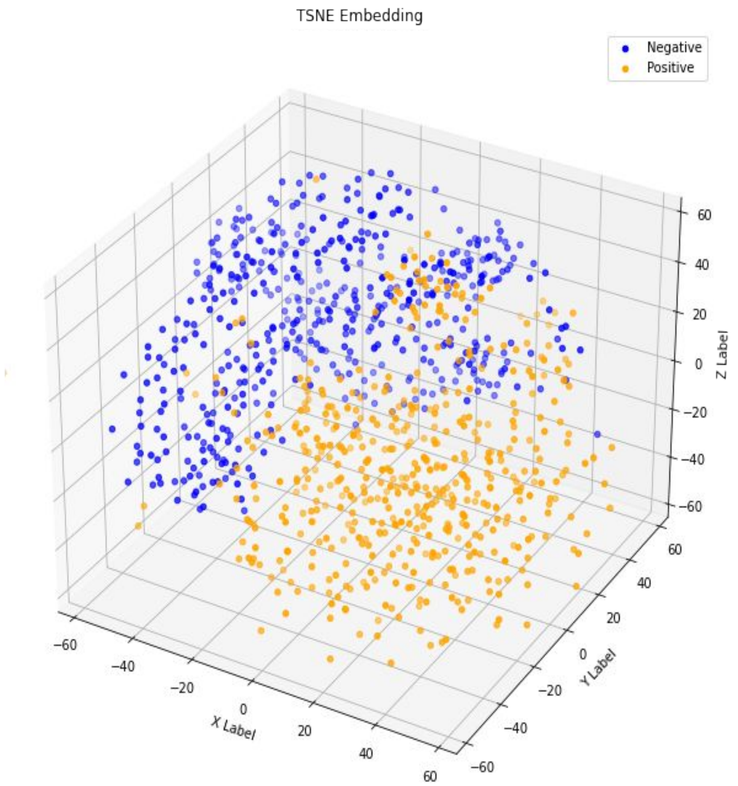
k-Means (2-clusters)

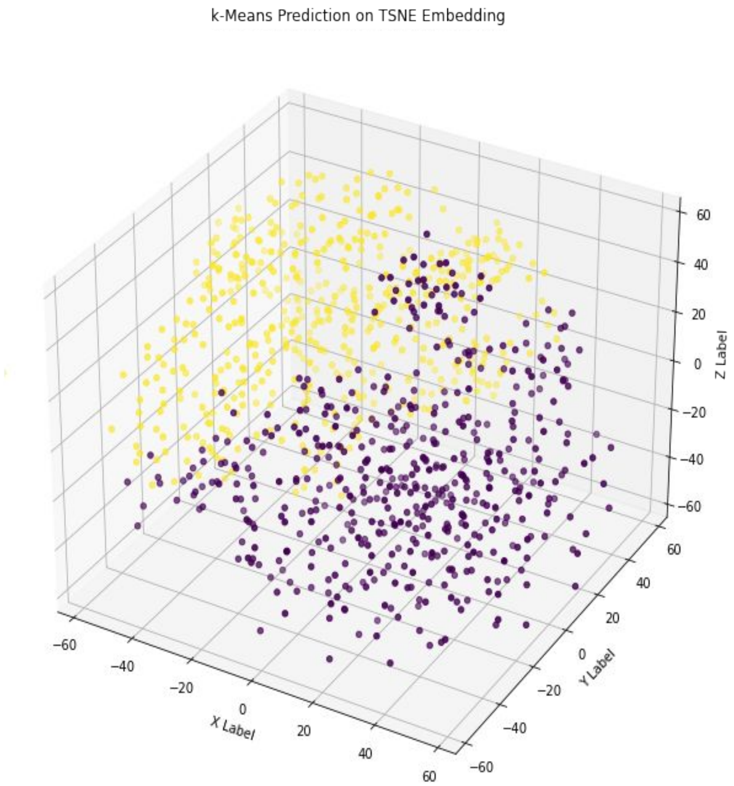
- How well can we predict on the 3D embedding?
- Spatial significance to the data
- The more diffuse the bright spot the more likely to not be an asteroid



	precision	recall	f1-score	support
0	0.898	0.900	0.899	499
1	0.900	0.898	0.899	501
accuracy			0.899	1000
macro avg	0.899	0.899	0.899	1000
weighted avg	0.899	0.899	0.899	1000

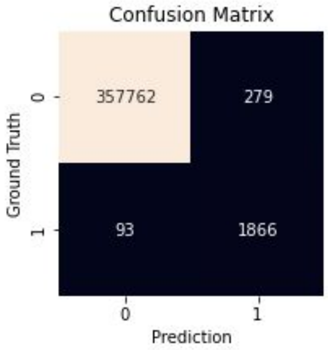
k-Means cont.


TSNE Embedding
Negative
Positive
X Label
Y Label
Z Label


k-Means Prediction on TSNE Embedding
X Label
Y Label
Z Label

Random Forest

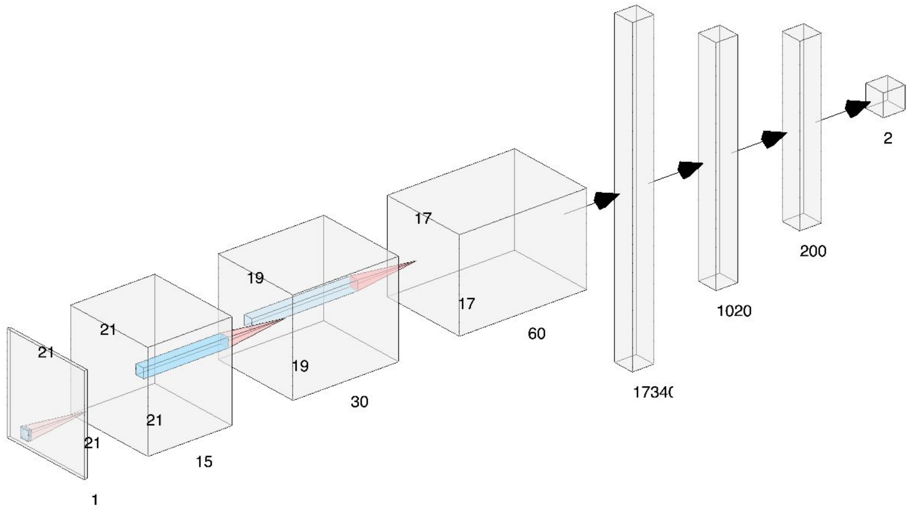
- Performed surprisingly well without any form of dimensionality reduction
- Overwhelming majority of error is False Positive
- Wasn't very competitive with CNNs



	precision	recall	f1-score	support
0.0	1.000	0.999	0.999	358041
1.0	0.870	0.953	0.909	1959
accuracy			0.999	360000
macro avg	0.935	0.976	0.954	360000
weighted avg	0.999	0.999	0.999	360000

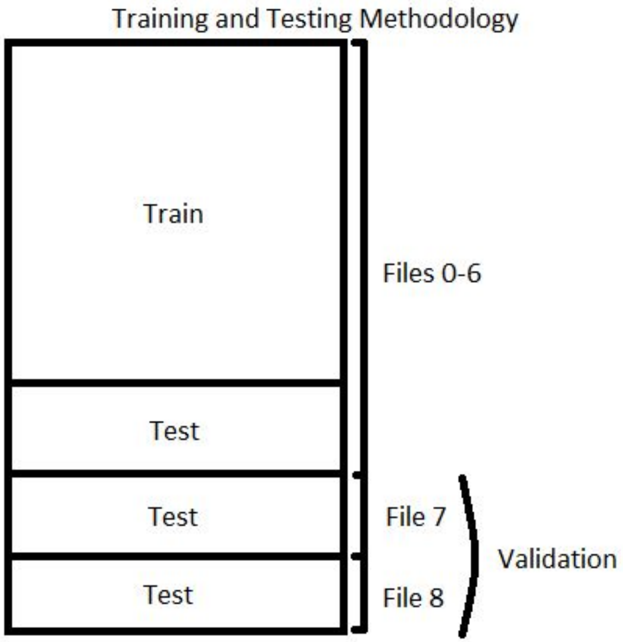
Baseline CNN architecture

- Based on the star and galaxy CNN model
- Trying to maintain as much information across convolutions
- Increasing in/out channels



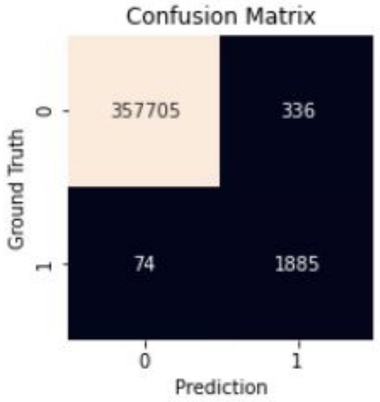
Training the CNN

- Near equal proportion of each class
- Shuffling in different negative images after each batch
- Too few negative samples in relation to population
- Requires more epochs to train
- Computationally expensive to shuffle
- Enough training data that overfitting is unlikely
    - Could increase feature set with reflection and translation
        - No more than 5 per img else we found it tended to not generalize



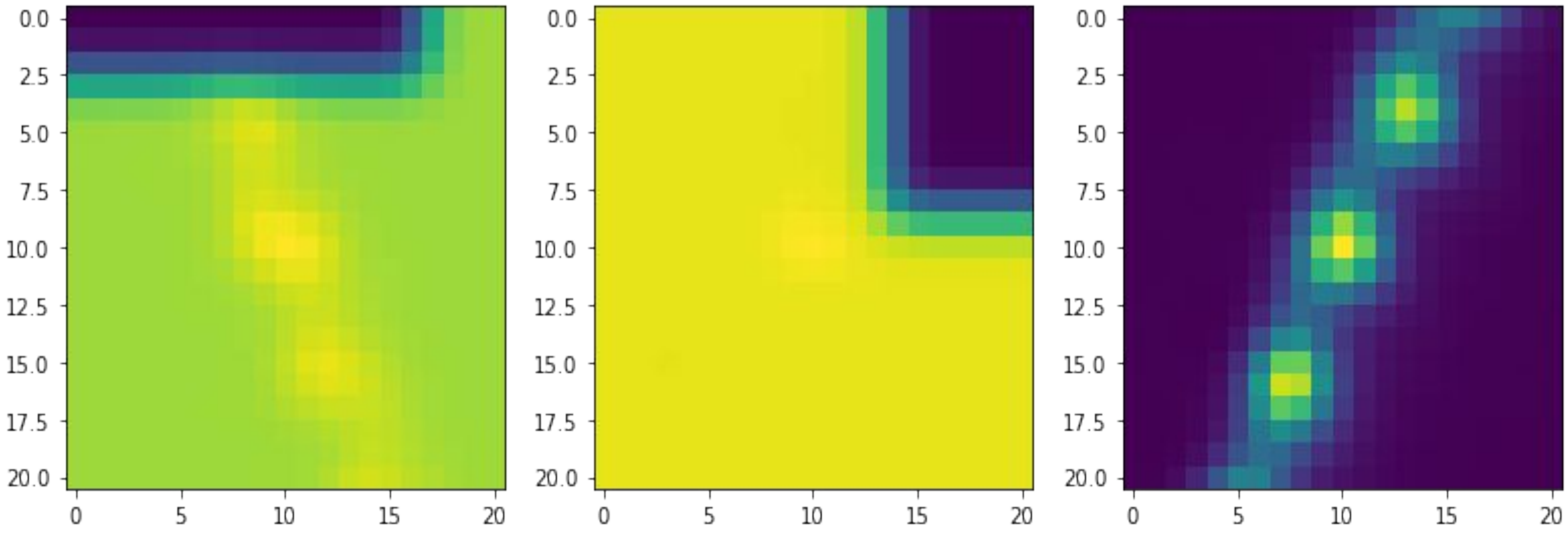
Single CNN

- Prediction rate is 10.31 images/second
- Too many False Positives
- A more complex model seems to be required to give the classifier a more robust feature set
- Weight decay to limit overfitting



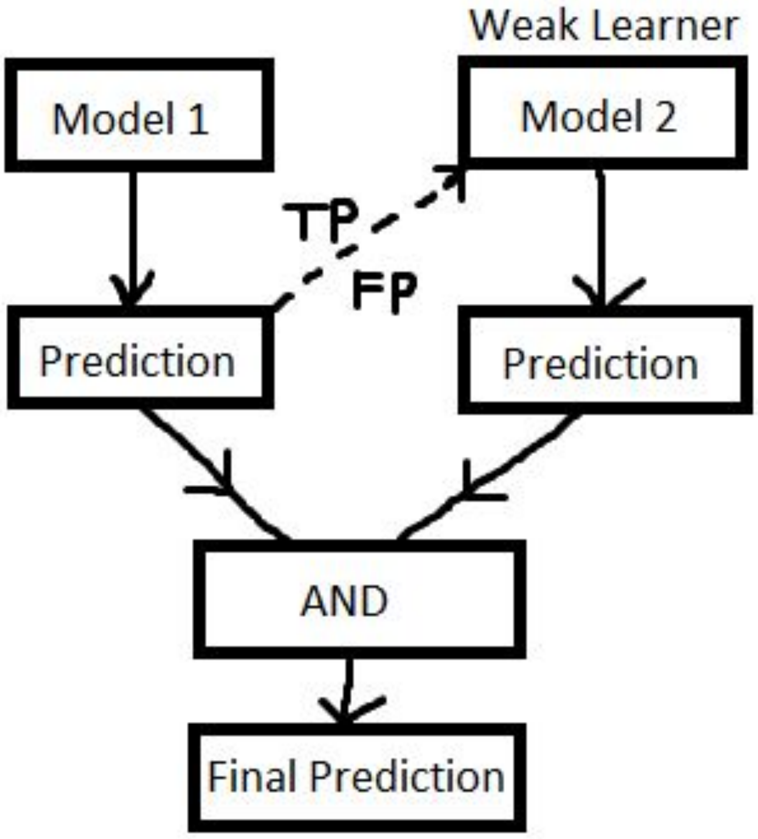
	precision	recall	f1-score	support
0.0	1.000	0.999	0.999	358041
1.0	0.849	0.962	0.902	1959
accuracy			0.999	360000
macro avg	0.924	0.981	0.951	360000
weighted avg	0.999	0.999	0.999	360000

False Positives



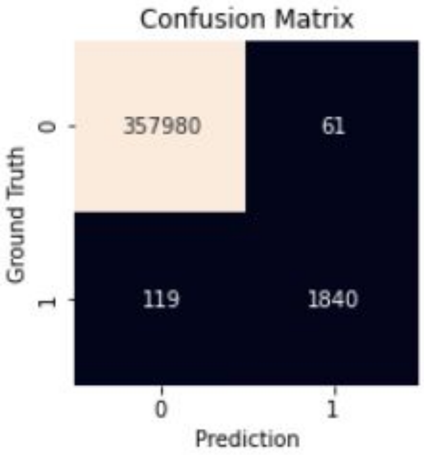
Boosting with CNN

- Tried to minimize the number of parameters
- Two separate CNN
- Feeding predictions from one model to another
- Anding the predictions from the two models to make the final predictions



Baseline Boosted CNN

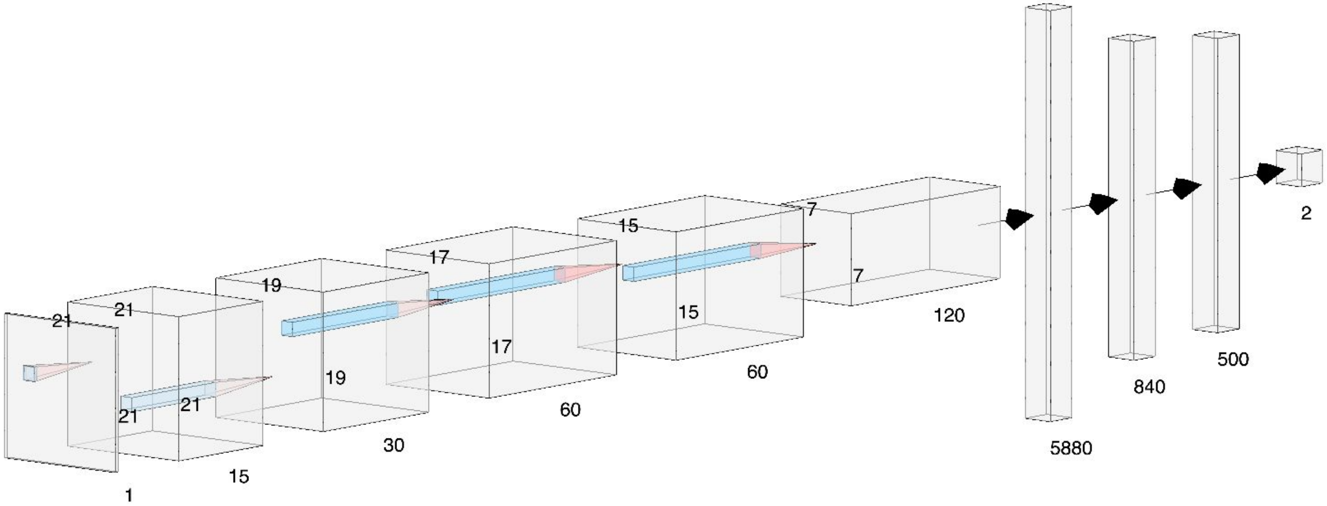
- Struggles with False Negatives
- Lacks nuance between artifacts and asteroids
- Too limited/starved of an architecture



	precision	recall	f1-score	support
0.0	1.000	1.000	1.000	358041
1.0	0.968	0.939	0.953	1959
accuracy			1.000	360000
macro avg	0.984	0.970	0.977	360000
weighted avg	0.999	1.000	0.999	360000

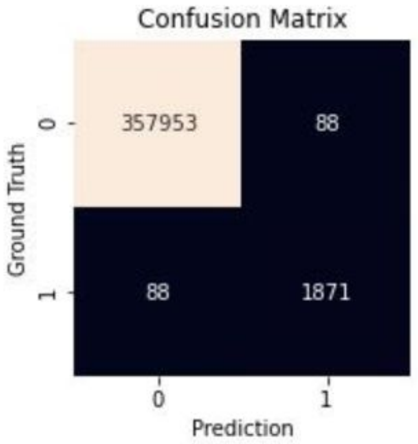
Different CNN architectures

- Wanted to see how average pooling would compare to max pooling
- Addition of a 4th convolutional layer
- Increase amount of input/output channels



Different CNN architectures cont.

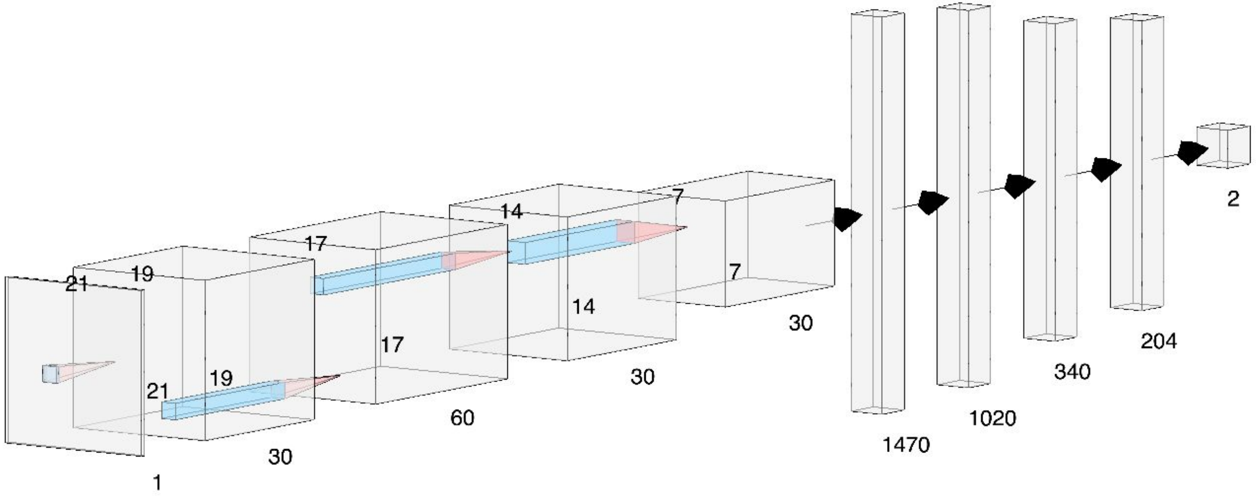
- Recall increased from baseline CNN
- Precision decreased from baseline CNN
- Average pooling smoothed out features
    - Features stand out less



	precision	recall	f1-score	support
0.0	1.000	1.000	1.000	358041
1.0	0.955	0.955	0.955	1959
accuracy			1.000	360000
macro avg	0.977	0.977	0.977	360000
weighted avg	1.000	1.000	1.000	360000

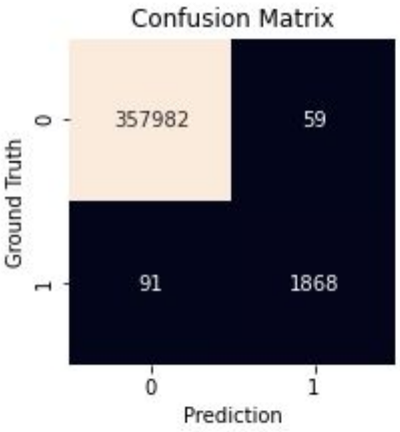
Different CNN architectures cont.

- More medium feature set and smaller low and high level features
- Similar linear layer size to baseline CNN



Different CNN architectures cont.

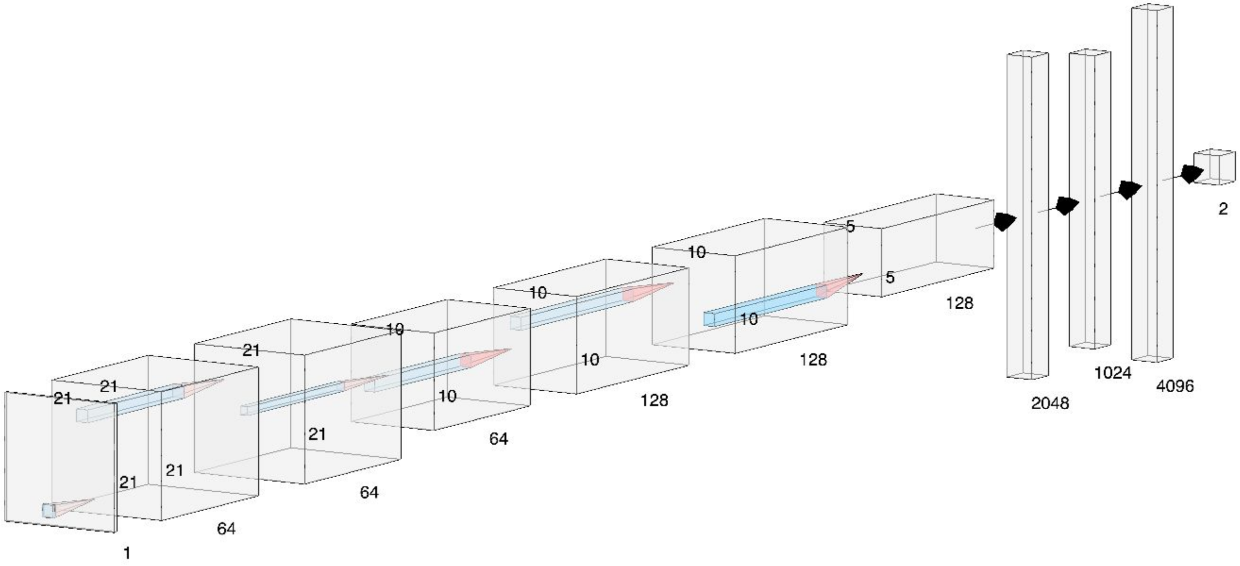
- The increase in feature size helped, but we see a change in the type of error we see to be more prevalent
- Performed better than the baseline CNN because it could capture more nuanced features



	precision	recall	f1-score	support
0.0	1.000	1.000	1.000	358041
1.0	0.969	0.954	0.961	1959
accuracy			1.000	360000
macro avg	0.985	0.977	0.981	360000
weighted avg	1.000	1.000	1.000	360000

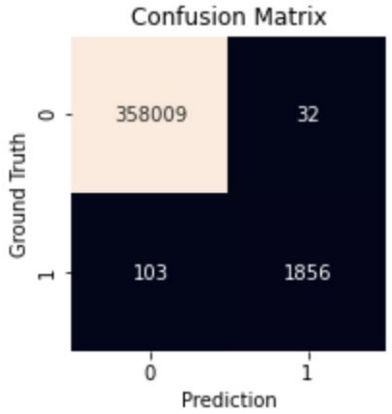
Different CNN architectures cont.

- Wanted to see how other established models worked
- VGG 16 but modified
- Implemented padding for all layers



Different CNN architectures cont.

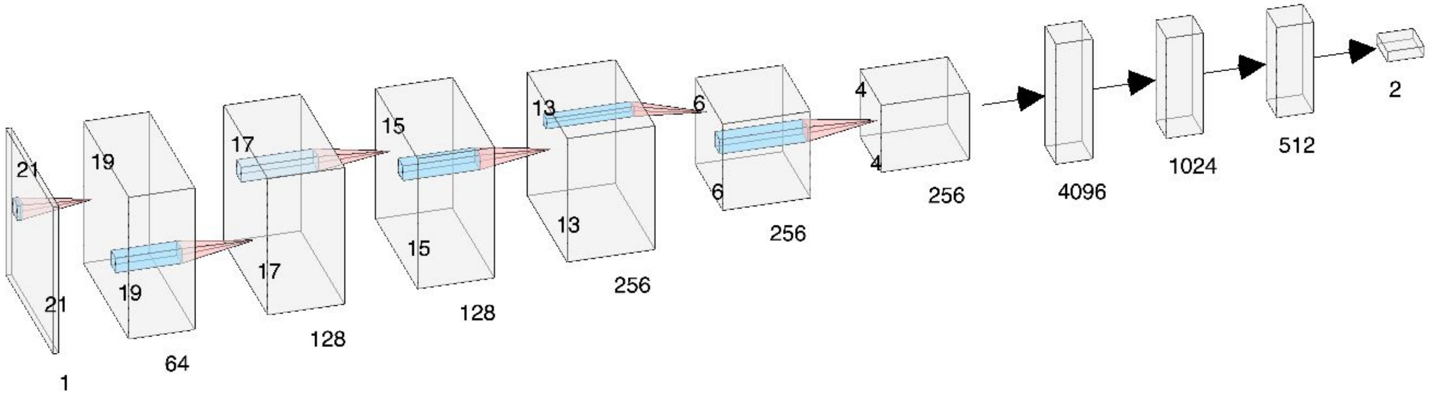
- Recall was reduced due to max pooling
    - Size of feature maps decreases
- Precision improved based on the length of the model



	precision	recall	f1-score	support
0.0	1.000	1.000	1.000	358041
1.0	0.983	0.947	0.965	1959
accuracy			1.000	360000
macro avg	0.991	0.974	0.982	360000
weighted avg	1.000	1.000	1.000	360000

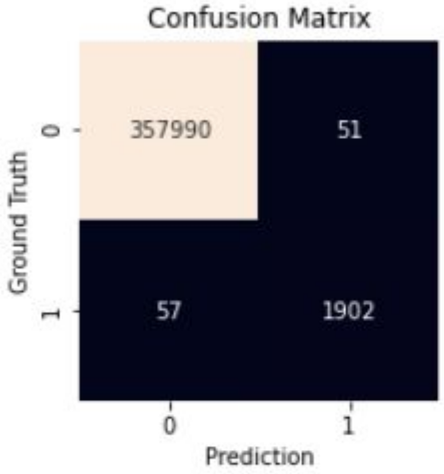
Final CNN architecture

- Increasing the feature set
- Enables more nuance features
- Increasing the number of neurons in the final layer of the model



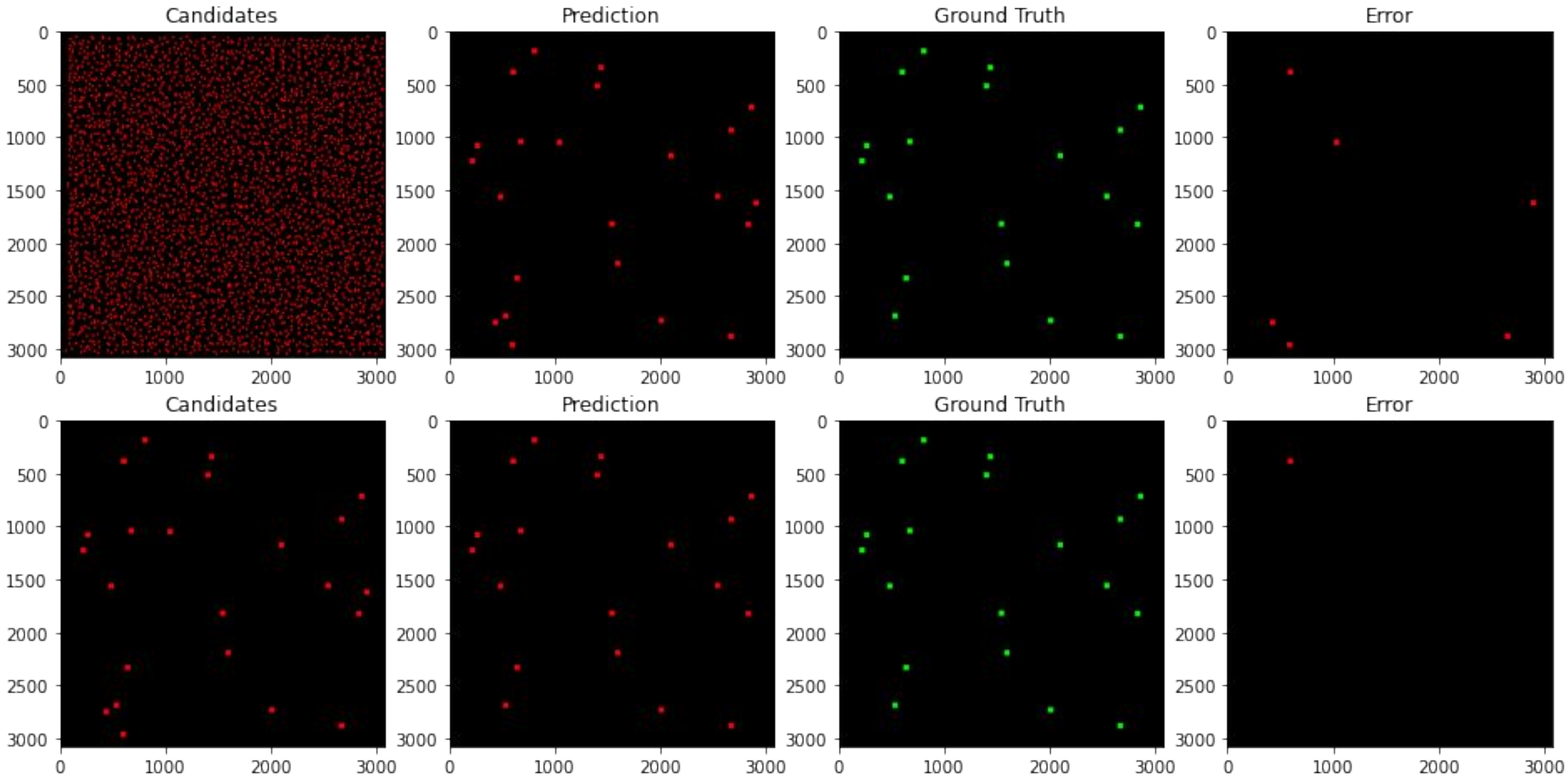
Final CNN architecture cont.

- Capacity to understand nuance between artifacts and asteroids
- Still room to explore different hyperparameters
- Error is more balanced between Type I and II
- There are missing asteroids, so recall will change (don’t know if we want it)



	precision	recall	f1-score	support
0.0	1.000	1.000	1.000	358041
1.0	0.974	0.971	0.972	1959
accuracy			1.000	360000
macro avg	0.987	0.985	0.986	360000
weighted avg	1.000	1.000	1.000	360000

Final CNN cont.



Conclusion

- The training methodology could be improved so that the model can better predict asteroids as they deviate from the center of the cutout
- Effective to generate datasets from a models error.
  - Training another model on that data which is mixed with a small sample of regular data is worth investigating further.
- Sources of error
  - Footprint Algorithm struggles to detect high magnitude asteroids
  - Using discovered stars to find undiscovered stars
  - If either of the neural networks classifies a asteroid, not a asteroid