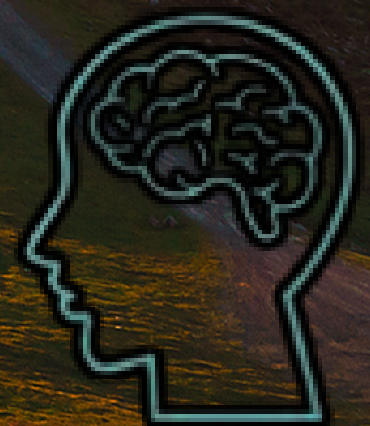


Wind Turbine Damage Recognition Model

Parker Manci



**YOUTH STEM
INITIATIVE**

Wind Turbine Damage Recognition Model

by Parker Mancini

Wind turbines are so important to the goal of going completely green - they provide a reliable electricity source that emits almost no greenhouse gases. Like any machine, however, they cannot perform at their best if they are not working properly. Damage has a huge impact on how a machine operates, so it is important to identify and fix it as fast as possible. In the case of a wind turbine, it is slightly difficult to do this because of their height, which ranges from 450-850 feet.

Drones have been used for a long time to help aid this issue, as they can take pictures of hard to reach places that can then be analyzed for damage. There is still room for improvement, however as a human is still needed to differentiate between damaged and not damaged wind turbines.

This is where image classification, a type of computer vision that categorizes images, comes in. By training one of these models off of images of wind turbines, it can learn to tell apart damaged and not damaged just as well as a human can.

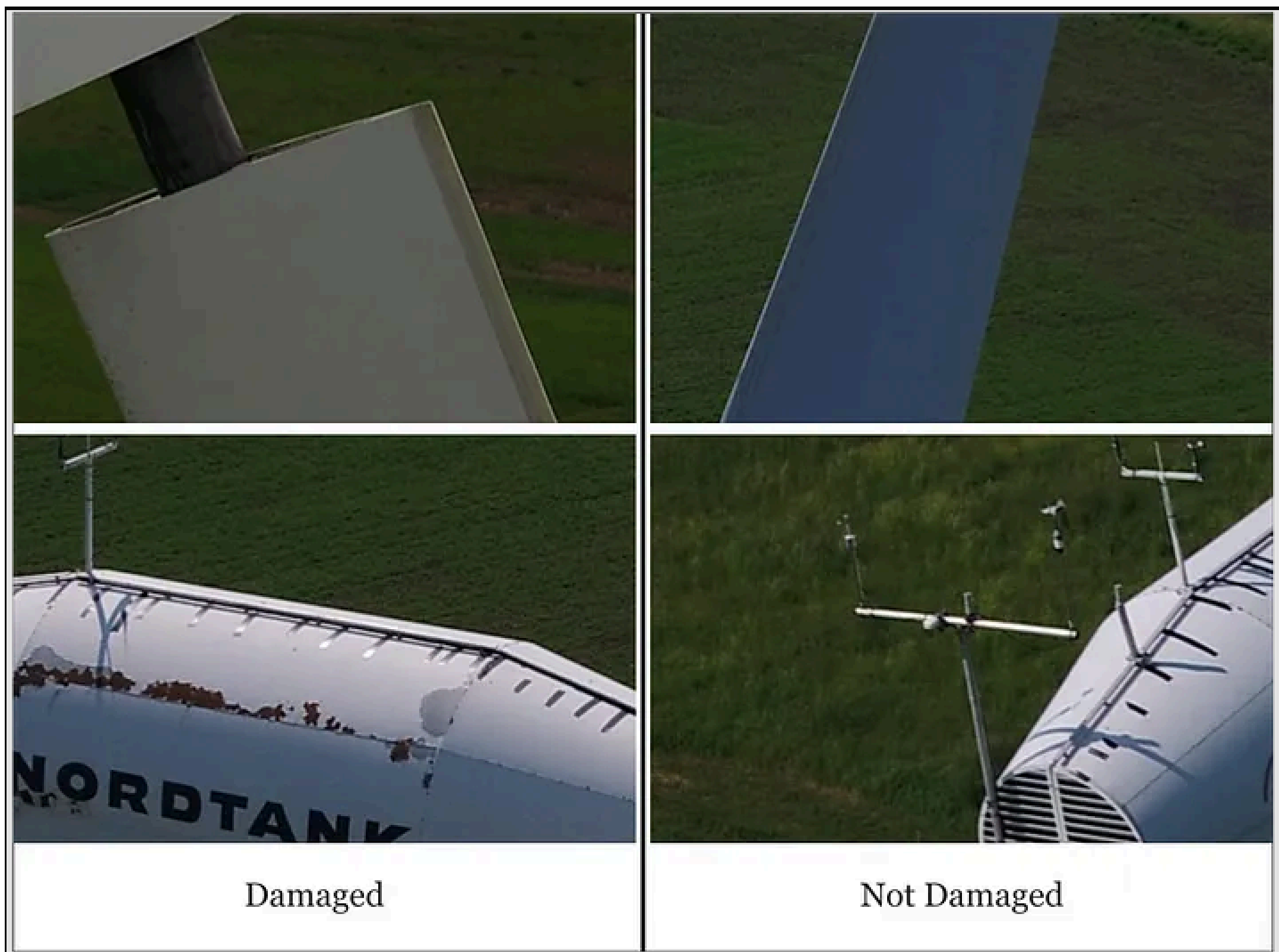
```

16 input_dir = "C:/Users/parke/Downloads/Wind_turbine"
17 categories = ['Damaged', 'Not_Damaged']
18 data = []
19 labels = []
20
21 for category_idx, category in enumerate(categories):
22     for file in os.listdir(os.path.join(input_dir + "/", category)):
23         img_path = os.path.join(input_dir, category, file)
24         img = imread(img_path)
25         img = resize(img, (15, 15))
26         data.append(img.flatten())
27         labels.append(category_idx)
28
29 data = np.asarray(data)
30 labels = np.asarray(labels)
31
32 X_train, X_test, y_train, y_test = train_test_split(data, labels, test_size = 0.2, shuffle = True, stratify = labels)
33
34 classifier = SVC()
35 parameters = [{'gamma': [0.01, 0.001, 0.0001], 'C': [1, 10, 100, 1000]}]
36 grid_search = GridSearchCV(classifier, parameters)
37 grid_search.fit(X_train, y_train)
38
39 best_estimator = grid_search.best_estimator_
40
41 y_prediction = best_estimator.predict(X_test)
42 score = accuracy_score(y_prediction, y_test)
43
44 print('{}% of samples were correctly classified'.format(str(score * 100)))

```

I built an image classification model that can make this process more efficient, as it can determine whether a wind turbine is damaged or not damaged based off of an image. I worked in PyCharm and used scikit-learn for the body, as shown in the picture above, and was able to achieve an accuracy rate of about 70%.

To summarize how it works, it takes in file of images during training that it resizes and sorts into categories. It then splits the data into a training set and testing set, and creates multiple image classifiers to find the best one for the data. Once it find this, it predicts the labels for the test set.



Here are some examples of images I used to train my model, with the left side being damaged and the right side being not damaged. While they are not perfect, I used them to train my model because I think they give a good representation of what surface damage usually looks like on a wind turbine.

While building this, one of the major challenges I faced was trying to find the right workspace to write my code in. I usually code artificial intelligence in Google Colab, so I had to pivot when I realized I could not load local files onto the platform. I ultimately worked in PyCharm, where I was able to upload the images needed to train my model.

This was my first time building an image classifier, so it was a learning experience modifying code to adapt to images instead of written data. After reading lots of articles and watching a lot of tutorials (This one in particular was super helpful), I was able to build a working model and gain many skills along the way.

For my next project, I want to use generative AI to predict cloud movements in a specific area for small period of time. I have never worked with this type of artificial intelligence before, so I know I will have to learn a lot of new skills in order to build it. I am up for the challenge however, and am very excited to see what comes next