Deep Learning and Machine Vision in Agriculture

A Study of Improving Neural Network Prediction Accuracy through Image Color Correction

Mark Mavis
Computer Science and Software
Engineering Department
The University of Washington
Bothell, WA, USA
mmavis@uw.edu

ABSTRACT

The agriculture industry is of central importance to U.S. economic stability due of its interconnected markets that function at global, regional, national, and local levels [1]. The U.S. agriculture industry represents 11% of all U.S. employment, contributes over 1.2 trillion dollars annually representing 5.7% of U.S. GDP production, and supplies nearly 25% of all grains to the global market [4], making it critical to the worlds food supply. But as the world's population continues to grow to a project 10 billion by the end of the century [2], we will see increased pressure on economic stability at a very fundamental level due to global resource scarcity [3], requiring humanity to produce more with less. Additionally, dense urban populations coupled with efficient modes of transportation make cities venerable to pandemics as we have seen from the coronavirus and its rapid spread around the world, leading to subsequent shutdowns of parts of the economy and unprecedented supply and demand shock to the food system [5] through food production and labor shortages. Both scenarios present great danger to the supply chain and global security and need to be addressed to guarantee future global security.

Modern technologies will be needed to fill these gaps to allow for increased food production efficiencies through data driven solutions to keep pace. After significant progress over the last decade, machine learning models and computer vision might be the solution that can provide both efficiency and resiliency against threats like the ones previously discussed. This report details how computer vision, a technology that can detect objects, can be used with machine learning, specifically neural networks, a technology that can be used to predict and categorize classes, is an extremely promising combination that can have a broad set of applications in the agriculture sector and provide increase in efficiency and resiliency. I will show how Keras, a high-level neural network library can be trained to accurately recognize an image from a group of fruit classes with a high efficiency as a proof of concept to make inferences about its future applications in agriculture. I will also show that neural networks such as Keras are susceptible to negative effects due to color temperature differences in images between training testing datasets and result in a decline in accuracy of the model, making color balancing, a process used to adjust the colors of an image to make them as accurate as possible, an important factor to be considered when collecting and processing the data that will be introduced into the ML model.

CCS CONCEPTS

• Computing Methodologies • Machine Learning • Machine Learning Approaches • Neural Networks

KEYWORDS

Machine Learning, Computer Vision, Agriculture, Deep Learning, Imagining Color Balance, Neural Networks, TensorFlow, Keras

1 Introduction

Agriculture can be considered one of the most important pillars of the world's economy. More than 45% of the world's population depends on agriculture for survival [6] and is a significant source of employment equaling 1.3 billion, making it the second greatest source of employment worldwide. In 2020, 19.7 million full and part time jobs were related to the agricultural and food sectors, making up 10.3% of U.S. employment [7]. Whether someone works on the farm, for a food manufacturing company, in a restaurant, or in a clothing or lumber store, that person's job depends on American agriculture. Agriculture naturally has its challenges also which can make it difficult to achieve high level efficiencies and prevent farmers from reaching their true potential with the resources they have available. Imagine having multiple essential processes to keep track of, excel at and monitor at the same time managing a work area often measured is hundreds of acres. When you consider this, it is easy to see why computer vision and machine learning are a perfect mix with agricultural practices in terms of crop automation and data collection. Gaining insight into how weather, seasonal sunlight, migratory patterners of animals, birds, insects, use of specialized fertilizers, insecticides by crop, planting cycles and irrigation cycles, all of which affect yield, is a perfect problem for machine learning. This research paper focuses in on how to improve computer vision and machine learning practices by showing how lighting conditions during different periods of the day can affect a machine learning model and by accounting for the lighting conditions, can increase the accuracy quite substantially.

2 Goals and Objectives of Research

The goal of this research is to develop a proof of concept by achieving three main objectives:

- Show that machine learning models are capable of being highly accurate in their ability to correctly identify different scenarios and can greatly benefit agricultural practices through a wide variety of applications.
- Provide evidence that machine learning models are susceptible to a decline in accuracy from color imbalances in training and testing imaging sets, thus showing that color correction is an important part of any machine learning model process and needs to be carefully considered when building and training a machine learning model
- Conclude which lighting combination scenarios create the highest model accuracy and which generate the lowest accuracy as to understand how to mitigate these issues in a real-world scenario.

3 Method

3.1 Project Objectives

- In our test cases, we will be treating the original dataset as the controlled group and consider it the “color corrected dataset”. Because the images color representation is true to real life, this will be a great bases for future iterations.
- Create duplicate dataset from the color corrected images and introduce a consistent color cast to each image in the set.
- There will be eight test cases performed throughout the research with two main subcategories, one titled “Presplit Dataset” which has the images already separated into training and testing sets, and another category title “Combined dataset” which will have all like images combined to together in their own class folder and with be randomized on each model training run to increase accuracy. Each subcategory will involve 4 individual experiments; 2 “pure” dataset passes where the model will be trained and tested on the same color type (i.e. trained on color corrected training images and tested on the color corrected testing images), and 2 cross comparison passes where the model will be trained on either the color corrected or color cast set, and then tested with the opposite datasets test group.
- The research format is the same for every iteration of the tests, where we will train one set of data from either the color corrected set or color cast training set, and then test on either the color corrected or color cast testing set.

3.2 Dataset Selection

Initially it was decided that the dataset being used in the research was going to be created from scratch by purchasing a series of different types of fruits and then creating an image catalog to control quality and consistency. Additionally, the flexibility to add more dataset to the set would be available given that the model began to exhibit low accuracy predictions. Once the dataset begun being created, it quickly became apparent that too much time was going to be spent on producing that dataset and would detract from the purpose of the study given the time frame. To mitigate weak correlations and assumptions created by the ML model due low number of data points available, a publicly available dataset that maintain a large number of images was selected consisting of over ninety thousand images. The dataset that was chosen titled “Fruits-360” came from Kaggle, a well-established community of data scientists and machine learners from all over the world. This dataset contains an image series of 131 different types of fruits that had been pre-color balanced to accurately represent true life color. The image set had also been reduced to 100px-by-100px resolution which eliminated a series of preprocessing steps, making it an ideal starting point for the data to be used in the study. The dataset was pre-split by its original creators into separate folder paths consisting of 75% images reserved for training the neural network and 25% used in the testing and validation of the model. This pre-split structure was initially used to get the neural network model up and running but it will be discussed later that this was altered to a structure were all available images of each class of fruit into their own class folders and then randomized before each iteration of training and testing to increase model accuracy.

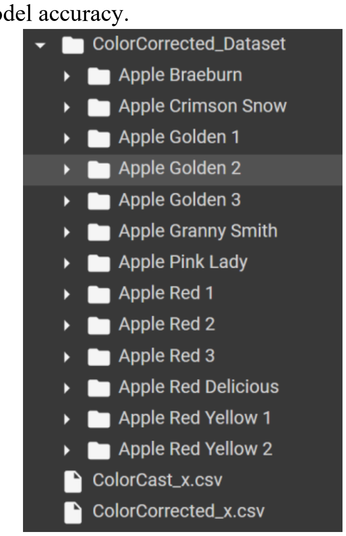


Figure 1: Dataset Storage Structure

In the interest of minimizing computational time in between the multiple iterations that will be run, an initial smaller subset of images was selected for testing purposes while building the neural network that will create the model as a proof of concept before the entire image set is feed into the model. This was because there was an assumption that unforeseen issues during the building of the machine learning system would happen considerably expanding the project timeline and should be mitigated by minimizing system training and testing times.

3.3 Dataset Creation

Once the dataset was decided upon and downloaded, the next critical step was to create an identical copy of that original dataset and to introduce a color cast, which is a shift in color balance in the image to simulate a time of the day when the natural light is much warmer and more reddish look to it as seen in figure 2.

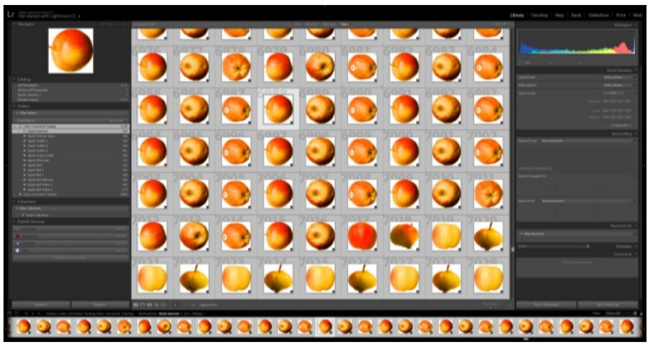


Figure 2: illustration of the image color balance shift processing in Adobe Lightroom

Adobe Lightroom was used to generate a replicated dataset consisting of images that have the same image manipulation. All images' temperatures were shifted exactly the same amount to establish a consistent look so that color variability wasn't a factor in accuracy fluctuations in the ML model training research results.

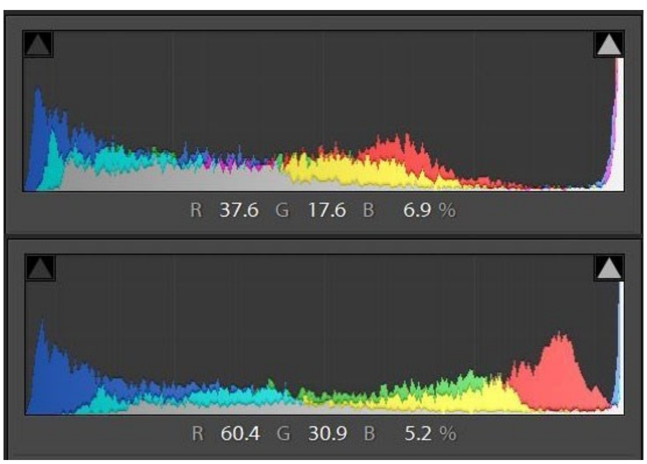


Figure 3: The image color distribution of an image before and after the color shift

Looking at the color histogram in figure 3, we see a distribution of color values throughout any given image and in the second picture of figure 3, you can see a significant shift towards the right side of the graph of the red and orange values, meaning that the image contains much more pixels that contain a reddish hue to them, essentially simulating a early morning sunrise or later evening sunset when the color in the sky is drastically different that the middle of the day.

3.4 Data Preprocessing and Model Creation Flow

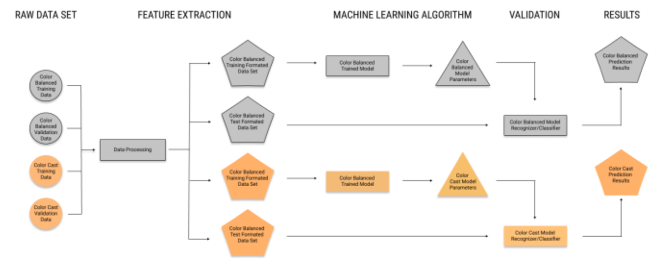


Figure 4: Diagram of the modeling framework and flow

All images in the dataset were preprocessed by stripping the individual RGB values of each pixel into a delimited csv file in the form of 9-tuple values separated by comas using Python's open-source computer vision library cv2. Each row in the csv files represent a flattened array of all 10,000-pixel values throughout each individual image. The number of rows in the files matches the number of images in the dataset. Additionally, a second csv file is used to store a single object code list that corresponds to each row in the pixel value csv. These are the labels that identify the class that the image belongs to that the model will learn to predict.

3.5 ML Modeling Platform Choice

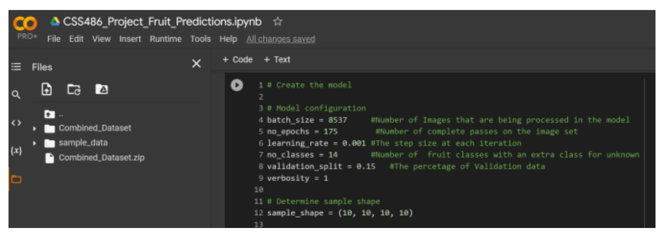


Figure 5: Example of the work environment this project was completed in

The project environment was carefully chosen to fit with the needs of the project and its ability to provide a model implementation that allows the user to choose different training and testing paths on the fly. Google Colab's usage of JupyterLab, a web based interactive development environment that is flexible by allowing users to configure and arrange workflows in a variety of ways of execution, giving the user a greater degree of control.

3.6 External Data Storage

A clever system of pre-compiled files and external storage had to be developed to overcome limitations imposed by the development environment. Each time the project was shut down, all project files are removed from the cloud server to maintain available storage resources for other users so existing work has to be compressed into a zip file and exported. The benefit of this is that the pre-compiled csv files of all image information can be directly imported from external storage to avoid the step of lengthy pre-processing feature extraction each time the project is being worked on.

4 Experiment and Results

4.1 Presplit Data Results

The first series of experiments were conducted using the presplit dataset color corrected images to see what kind of accuracies could be obtained. The goal was to use this group of images as a baseline to compare all other training and testing combinations against.

4.1.1 Trained and Tested with Color Corrected data: 67%

In this iteration, the neural network was trained using just the color corrected presplit data set and then test and validated using the color corrected testing group of images. As we can see in figure 6, the performance of the accuracy on the validation data is much worse than the training on the training set and resulted in a 67% accuracy rate. We can also see from the model's cross entropy graph, which is the difference between two probability distributions

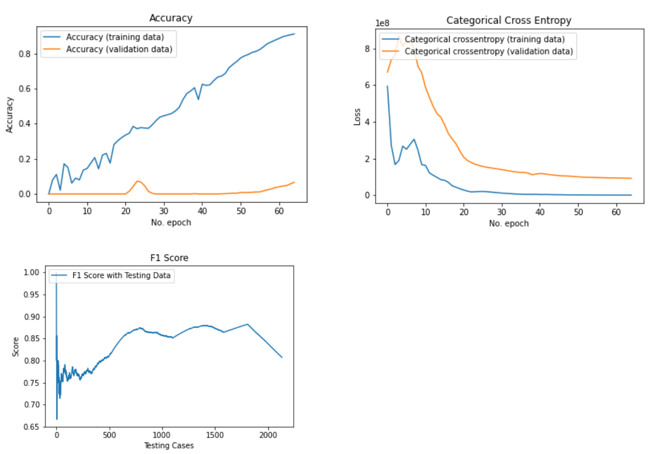


Figure 6: Direct comparison results of training with color corrected images and testing with color corrected images

4.1.2 Trained and Tested on Color Cast data

In this iteration, the model was trained using the color cast image set and tested with the color cast testing group. This iteration was designed to replicate a model that has been train on images taken in the middle of the day and to see how it performs prediction objects during that same period of the day. In figure 7, we see very similar patterns to what we saw in figure 6 with one significant difference, and that is that the highest accuracy was approximately 52.1% which was a 15% departure from the highest achievable accuracy with the color corrected dataset.

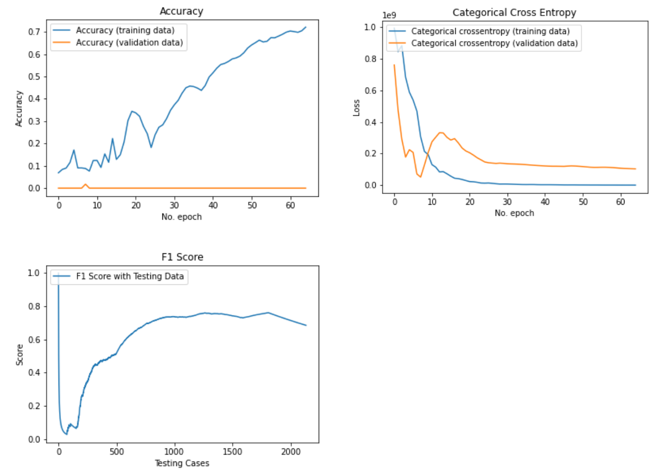


Figure 7: Direct comparison results of training with color cast images and testing with color cast images

4.1.3 Trained on Color Corrected/Tested with Color Cast

This iteration was the first of the series where the model would be trained and tested on opposite datasets. In this case, the model was trained on the color corrected dataset and then tested on the color cast testing group. The purpose of this scenario was to simulate a model that had been trained on images that were taken during the middle of the day when the natural light is balanced and to see how it would perform when predicting objects at a time of day when the sun is low in the sky and produces much warmer light which is cast of the object. In figure 8, we can see that the accuracy of the validation data is completely flat and struggles to even move. This has resulted in a dismal accuracy rate of 10.1%

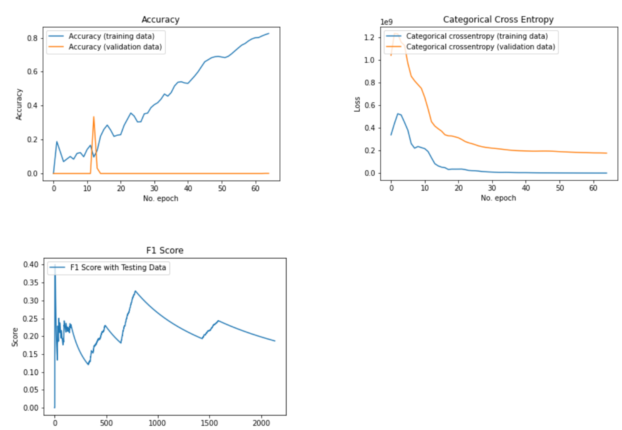


Figure 8: Cross comparison results of training with color corrected images and testing with color cast images

4.1.4 Trained with Color Cast/Tested with Color Corrected

This iteration was designed to replicate the situation where the model was trained images that were taken during sun set or sun rise periods and then used to predict objects during the middle of the day when lighting color conditions are more balanced. The highest accuracy that we were able to achieve was 29% which is not great but better than the model simulating the opposite scenario.

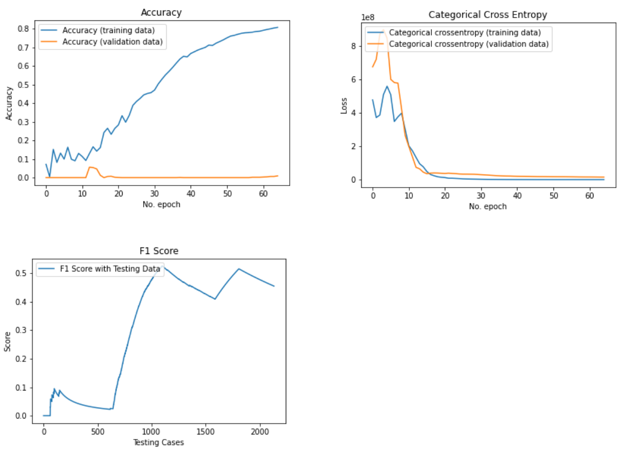


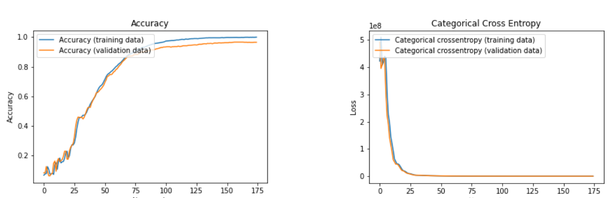
Figure 9: Cross comparison results of training with color cast images and testing with color corrected images

4.2 Combined Data Results

After the training and testing iterations from the presplit dataset were complete, the decision was made to try and pursue ways of improving overall accuracy, so the presplit color corrected and color cast datasets that had 25% of the images split off from the training set were combined into one single dataset. This way during the creation of the delimited csv file creation containing all pixel color values, the image rows could be randomized and then split into training and test sets.

4.2.1 Combined Randomized Color Corrected Run

This iteration used the newly implemented dataset randomization that allows the user to choose the percentage size of the training and testing data sets initially, and then the rows of the image csv files where then randomized and broken into those chunks. Again, this iteration is meant to replicate training a model on color balanced images and then predicting objects at that same part of the day. This time in figure 9 we can see great performance out of both the training data and the validation data. The model was able to achieve a 95.3% accuracy rate. It looks like the model learned accuracy begins to level off around 70 epochs and could be revised but this iteration shows that the randomization made a huge difference in the accuracy of the model.



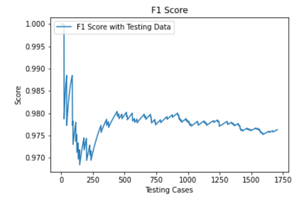


Figure 10: Combined randomized color corrected dataset results

4.2.2 Combined Randomized Color Cast Run

This iteration is using the combined randomized dataset and is simulating a model that is trained on images that were taken during periods of natural light when warmer in coolers and present and then the see what the accuracy is of predicting on images during that same period of time. We see in figure 11 that the results are very similar to the previous run but resulted in an accuracy of 86.7%, a 9-point reduction from the color corrected set.

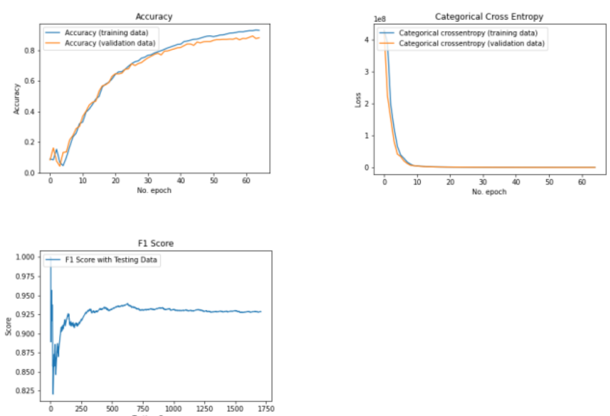
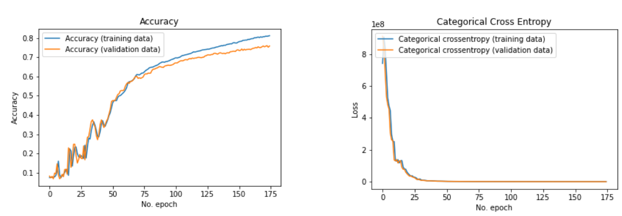


Figure 11:

4.2.3 Trained with Color Cast/Tested with Color Corrected

This iteration will use the randomized dataset and simulate a model trained with images taken during periods of high color cast and tested with color corrected images. The highest achievable accuracy rate that the model was able to achieve was 15%.



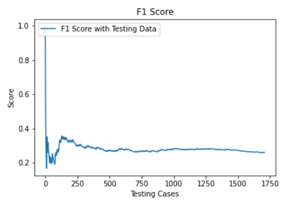


Figure 12:

4.2.4 Trained with color corrected/tested with color cast run

The final iteration uses the randomized dataset and is simulating a model trained on the color corrected images and tested with the color cast images. This model also we very close to the previous iteration in its results and resulted in a small reduction of accuracy at 13.8%.

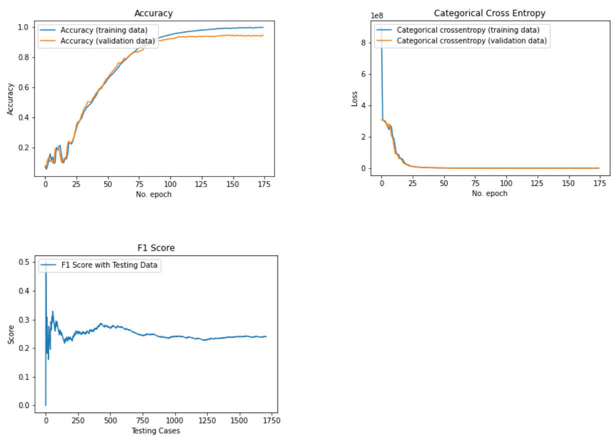


Figure 2

5 Conclusion and discussion

5.1 Presplit Dataset

After completing all 4 experiments with the presplit data, it was clear to see that the model being trained with the color corrected images and then tested on the color corrected testing set with the model being trained on the color cast images and tested with the color cast testing set was our second-best results. While this may have been assumed that this would have been the case, what is interesting is that in theory, both of these pure colors corrected and color cast runs should have been somewhat equal if the color cast wasn't a factor in the models accuracy. Considering they are both identical datasets other than the color, the fact that the color cast run couldn't get within approximately 15% of the accuracy of the color corrected run shows that the color balance of an image does have a significant impact on the model.

Both cross training and testing runs model accuracy was extremely bad but, in our testing, we did see that the model that was being trained on the color cast images and tested on the color corrected testing set was able to hit 29% accuracy versus the later only getting to 10%. So, while both of these models seem to be unable in any real-world application, there is some evidence that says one scenario affects the model more than future models could take this into account based on what scenario the model is trying to perform in.

5.2 Combined Dataset

In our series of runs on the combined dataset, we were able to increase our efficiency quite a bit throughout the implementation of a data randomization function which increase our pure runs by a significant amount. The accuracy of both the color corrected and color cast runs from the presplit data increased by over 25%. But what was interesting is that both of those increases were proportional to each other, which says something about the accuracy of the modeling itself. Again, the color corrected run outperformed the color cast run by approximately 10% which again, backs up the theory that color cast images negatively affect the model's accuracy.

Each of the cross comparison runs seem to perform very similar at around 13 percent. These results do cast a little bit of doubt on the both the cross comparisons of both datasets because in theory, both of them should operate at the same accuracy proportions that all 4 of the presplit dataset runs operated at.

5.3 Discussion

After all the testing and research was completed throughout the course of this experiment, I am confident at declaring that color cast do play a significant role in model accuracies when it comes to the use of computer vision agriculture applications. Because of this, a series of challenges that will need to be solved to build tools for farmers that will be able to effectively work due to the changing lighting conditions. It is easy to say that if a farmer is going to use computer vision and machine learning, to just use those tools during periods of the day when the sun is high in the sky so there is no color interference and low color variability during the application of the model, but this is not a luxury that a farmer will be able to afford. Farms utilize a wide range of times from sunrise to sunset and the technological automation tools will need to accurately accommodate for this at all times of the day.

Something I would like to add to the discussion of these findings is that because we can see that machine learning models can be much for accurate when objects are lite by very balanced lights (white light), the sector of agriculture that would immediately benefit the most would be the farmers who are implementing indoor controlled growing environments like vertical farmers. Because they are artificially creating the lighting conditions, they can provide an consistent quality of light which will benefit any technologies using computer vision and machine learning.

ACKNOWLEDGMENTS

I would like to acknowledge and thank Adobe for allowing their software to be a part of this research. Their photo manipulation suite was pivotal in allowing the editing and creation of our second dataset seamlessly without any problems or laborious work. Without their software's batch imaging manipulation abilities, it would have been next to impossible to create any kind of consistency across images all images that had a color cast introduced. Special regards to Google Colab and JupyterLab's for providing the computational resources to handle the workload. Their software's ability to compartmentalize code blocks that can be compiled in any order, made the research process that we were going for possible.

REFERENCES

[1] Alston, Julian M., and Philip G. Pardey. 2014. "Agriculture in the Global Economy." Journal of Economic Perspectives, 28 (1): 121-46.DOI: 10.1257/jep.28.1.121

[2] United Nations, Department of Economic and Social Affairs, Population Division (2019). World Population Prospects 2019: Highlights (ST/ESA/SER.A/423). https://population.un.org/wpp/Publications/Files/WPP2019_Highlights.pdf

[3] Alexandratos, N. and J. Bruinsma. 2012. World agriculture towards 2030/2050: the 2012 revision. ESA Working paper No. 12-03. Rome, FAO. https://www.fao.org/3/ap106e/ap106e.pdf

[4] McConnell, M. (Trans.). (2021, May 28). Feedgrains Sector at a Glance. U.S. Department of Agriculture. Retrieved December 10, 2021, from https://www.ers.usda.gov/topics/crops/corn-and-other-feedgrains/feedgrains-sector-at-a-glance/

[5] What is the impact of COVID-19 disease on agriculture? SICHE, Raúl. Scientia Agropecuaria [online]. 2020, vol.11, n.1, pp.3-6. ISSN 2077-9917. http://dx.doi.org/10.17268/sci.agropecu.2020.01.00.

[6] François Bourguignon, Maurizio Bussolo, 2013. Income Distribution in Computable General Equilibrium Modeling. Handbook of Computable General Equilibrium Modeling Volume 1, Pages 1383-1437. https://doi.org/10.1016/B978-0-444-59568-3.00021-3.

[7] Kathleen Kassel, Anikka Martin, 2021. Ag and Food Sectors and the Economy.https://www.ers.usda.gov/data-products/ag-and-food-statistics-charting-the-essentials/ag-and-food-sectors-and-the-economy/