Flower Classification: Analysis of Data Augmentation Strategies

Diego Acevedo, Samantha Colbert-Neal, Pallav Kamojjhala

Department of Engineering, University of San Diego

AAI 501: Introduction to Artificial Intelligence

Prof. Friesen

12 August 2024

Flower Classification: Analysis of Data Augmentation Strategies

Our project focuses on creating an AI tool that can recognize and classify different kinds of flowers. The main goal of our project is to build a strong and reliable neural network that can accurately spot various flower types in images. We're aiming to develop an AI agent that works well across a range of real-world situations, using some of the latest deep learning techniques.

To make this happen, we’re using a dataset from Kaggle called "Flowers," which has images of 16 different flower species. This dataset is challenging because the photos vary in how they’re taken—different angles, lighting, and backgrounds, which makes it a tough but perfect training ground for our model. The variety in this dataset is like what we’d expect in real-life situations, so it’s a great fit for our project.

The project involves several key steps: collecting and preparing the data, picking and training the model, and then testing how well it works. We’re focusing on convolutional neural networks (CNNs) because they’ve proven to be good at tasks like this (Li, Zhu, & Li, 2023). Our goal is to experiment with different data augmentations to find the most accurate classification model.

This report will walk you through everything we did—from the methods we used to the challenges we faced. We hope that by sharing our process, others can learn from our experience in developing this flower detection AI.

Approach

This work is using TensorFlow and Keras to create and evaluate many convolutional neural networks (CNNs) for flower categorization. The first stage of data preparation was loading a 15,740 collection of pictures from 16 different floral categories. From this dataset, sets for training (80%) and validation (20%). Keras's Sequential API was used for building the models; the Adam optimizer was coupled with the Sparse Categorical Cross entropy loss function for data validation and data training loss.

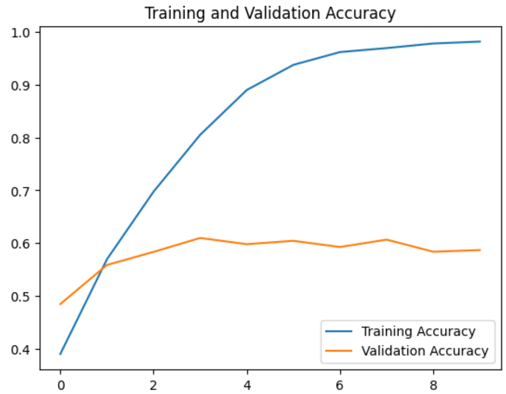
The initial model, was designed with three Conv2D layers, each followed by Max Pooling2D layers, then flattened, and connected to a dense layer, finally an output layer for classification. Next, additional models were built utilizing several data augmentation techniques to find how each of the other models—Random Flip, Random Crop, Random Translation, Random Rotation, Random Zoom, Random Contrast, and Random Brightness—would affect model performance (Daffodil Insights, 2024). By combining all these augmentation techniques, another model was constructed to evaluate the total influence on performance.

Each of these models trained with ten iterations; the training and validation accuracy were noted and plotted to show their respective performance.

Key Findings

Basic Model

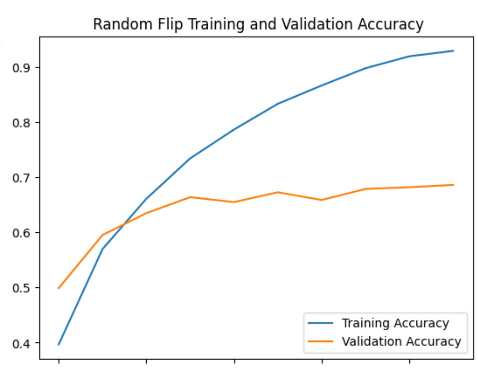
Starting strong, this model began to overfit during the fourth epoch. It was performing somewhat poorly on fresh, unseen data but well on the training set. It produced a validation accuracy of 58.70%.



Random Flip

This layer will randomly flip the inputted image in horizontal and/or vertical directions. The parameter passed for this layer is ‘mode’ and we chose to set it to ‘horizontal’.

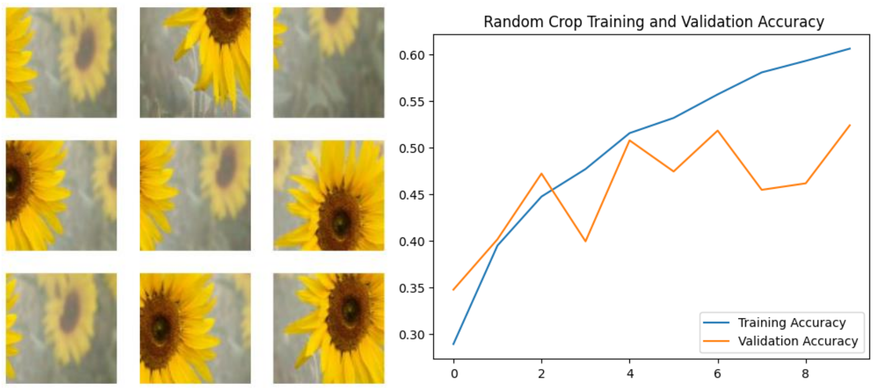
Including the random flip augmentation turned out to make the more accurate. This model developed to a validation accuracy of 68.58%, a good increase from the fundamental model.



Random Crop

The Random Crop layer crops the image to the targeted image size. To get the targeted image size we passed the ‘height=8’ and ‘width=80’ parameters to get the cropped image size. The ‘seed’ parameter that is passed controls the randomness of the image cropping.

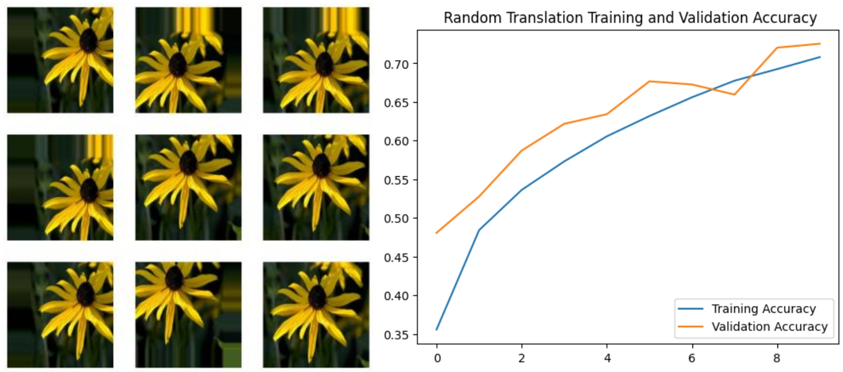
Not every enhancement worked out. With a validation accuracy of 52.35%, this one performed truly poor. Cropping seems to have severed vital portions of the blossoms, producing less than ideal outcomes.



Random Translation

The Random Translation shifts the image by height and/or width direction. The parameters passed are ‘height_factor=(-0.2, 0.3)’ and ‘width_factor=(-0.2, 0.3)’ which represent the fraction of the total image by height and width that the image is shifted.

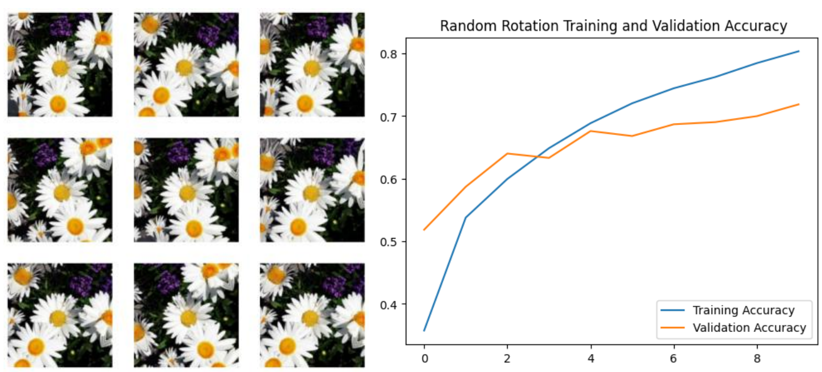
With a validation accuracy of 72.52%, the random translation model was proven to have the higest validation accuracy. Shifting the photos around seems to have made the model more flexible since it enabled it to learn to identify flowers wherever in the picture.



Random Rotation

This layer will randomly rotate an image within a certain range. We set our range of rotation to '.1' degrees.

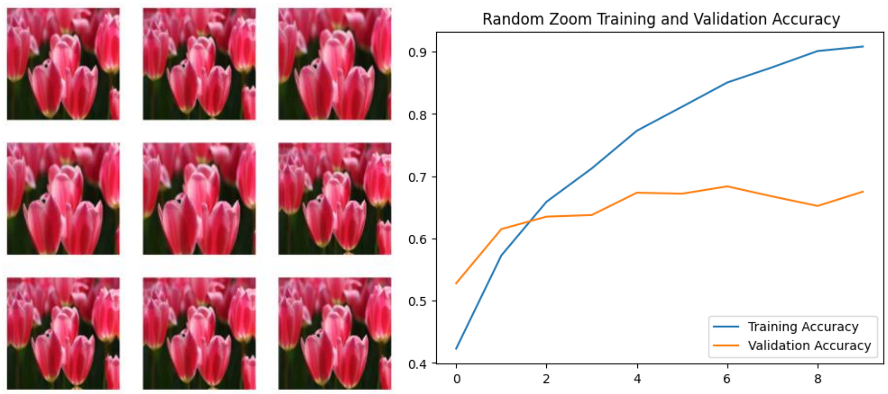
With a validation accuracy of 71.86%, Random Rotation Model likewise performed admissibly. Since flowers can show at any angle in images, rotating the pictures didn't seem to greatly confuse the model too much.



Random Zoom

This layer randomly zooms in or zooms out of the image. We passed ‘0.2’ in as our zoom value, this positive value translates to zooming in.

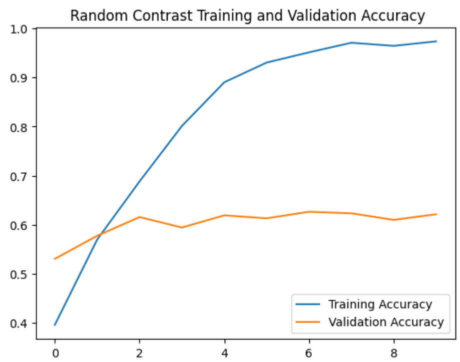
With a 67.50% validation accuracy, this model performed rather well. Random Zoom Model Though it wasn't as strong as rotation or translation, it nonetheless improved upon the fundamental idea.

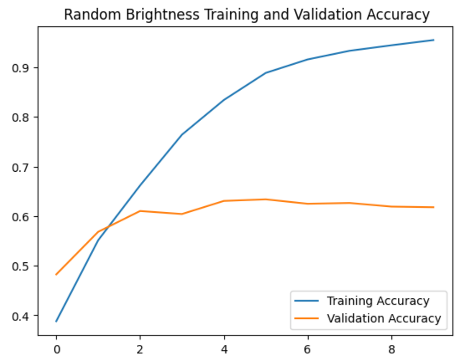


Random Contrast & Brightness

The Random Contrast last will alter the contrast of the image and the Random Brightness layer randomly alters the brightness of the image. The parameter used for the brightness layer is ‘factor’ and the value ‘0.5’ represents the percentage of a positive change in brightness

With validation accuracy of 62.13% and 61.79%, these augmentations had less influence. The performance of the model wasn't much changed by varying brightness and contrast.

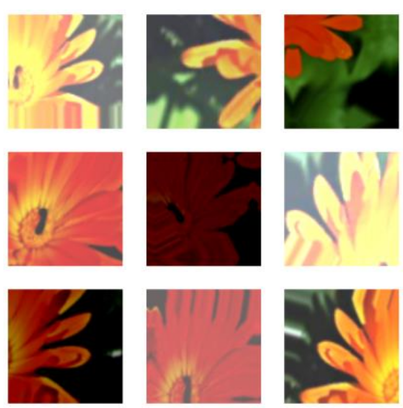


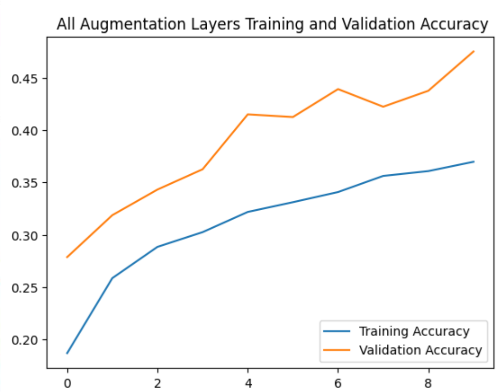


All Data Augmentation Layers

We wanted to test out what would happen if we applied each data augmentation to one model. This model is a combination of: Random Flip, Random Crop, Random Translation, Random Rotation, Random Zoom, Random Contrast, and Random Brightness

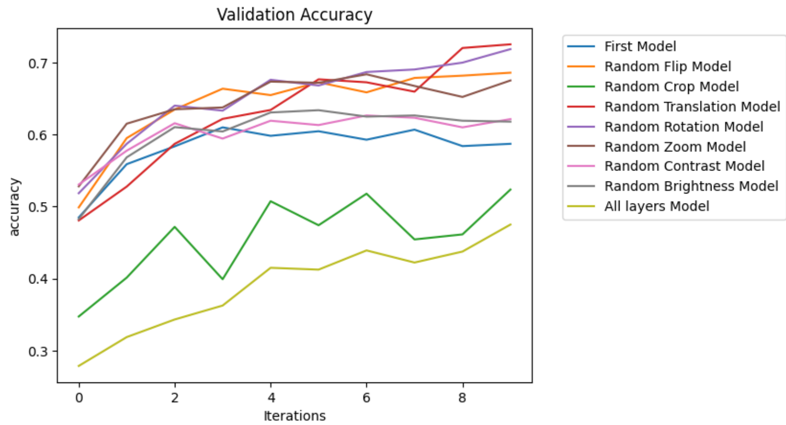
Surprising as it was, this model performed the worst—with a validation accuracy of just 47.49%. Combining too many augmentations turns out to make the work more difficult rather than beneficial.





Validation Accuracy Across Models

We can see that the Random Translation (red) and Random Rotation (purple) performed the best and were some of the most accurate models. Random Crop (green) and All layers (yellow) significantly underperformed compared to the rest of the models, while our first basic model seemed to have an average performance landing in the middle of the other models.



Conclusion

Turning out the best performance with a validation accuracy of 72.52%, the Random Translation Model. Given translation augmentation allowed the model to become more flexible in identifying flowers, independent of their placement in the image—a common circumstance in real-world images—this model most likely performed well. The model that aggregated all the augmentations failed most likely because the combined alterations made it difficult for the model to learn effectively. The Random Translation Model found the perfect equilibrium and improved performance without overloading the model with too many modifications.

References

Altamirano-Gomez, G., & Gershenson, C. (2023). A comprehensive review of convolutional neural network architectures and their applications. IEEE Access, 11, 4567-4583. https://doi.org/10.1109/ACCESS.2023.4567114Daffodil Insights. (2024). What is data augmentation in deep learning? Techniques and applications. Daffodil Software. https://insights.daffodilsw.com/blog/what-is-data-augmentation-in-deep-learning

l3llff. (n.d.). *Flowers*. Kaggle. Retrieved from []]]]]]]

Li, W., Zhu, N., & Li, M. (2023). Optical convolutional neural networks: Methodology and advances. Applied Sciences, 13(13), 7523. https://doi.org/10.3390/app13137523

Russell, S., & Norvig, P. (2021). Artificial intelligence: A modern approach (4th ed.). Pearson.

ScienceDirect. (2024). Data augmentation in deep learning: A comprehensive survey. ScienceDirect. https://www.sciencedirect.com/science/article/pii/S0957417422007763

Taye, M. M. (2023). Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions. Computation, 11(3), 52. https://doi.org/10.3390/computation11030052

Tzafestas, S. G. (2018). Synergy of IoT and AI in modern society: The robotics and automation case. Robotics & Automation Engineering Journal, 3(5), Article 555621. Retrieved from https://sandiego.instructure.com/courses/12847/files/1792425?wrap=1

Appendix

All team members agreed to and equally contributed to the following bodies of work:

- **Project Proposal**
  - Dataset Selection
- **Code**
  - Approach
  - Data Prep
  - Data Augmentation Models
- **Technical Report**
  - Introduction
  - Analysis / Interpretation
  - Conclusion
- **PowerPoint Presentation**
  - Slides + Voiceovers