


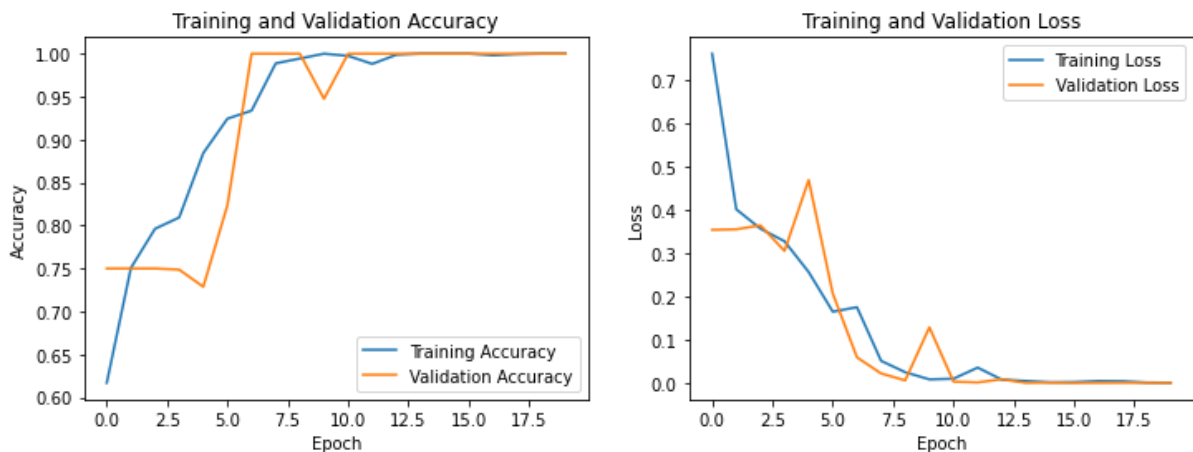
Semester (Term, Year)	Fall, 2023
Course Code	AER850
Course Section	1
Course Title	Introduction to Machine Learning
Course Instructor	Reza Faieghi
Submission	Project
Submission No.	1
Submission Due Date	26/11/2023
Title	Using Image Classification in a DCNN
Submission Date	27/11/2023

Authors/Contributors:	Student Number (XXXX12345):	Signature*:
Shaurya Ganguly	XXXX15243	

By signing the above you attest that you have contributed to this submission and confirm that all work you contributed to this submission is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, and "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Academic Integrity Policy 60, which can be found at www.torontomu.ca/senate/policies/

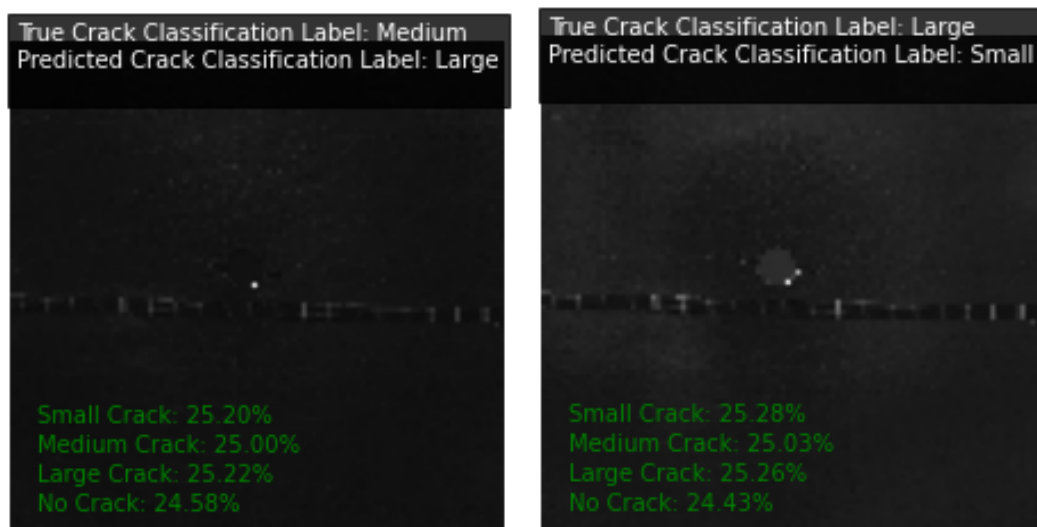
Model Performance:

These two graphs assess the loss and accuracy performance of the model. They behave as expected as the loss and accuracy curves are inverse to each other indicating a successful model. However, the model and training accuracy seems way too high from what is expected, and the validation loss appears to spike at abnormal locations as well. This could be because the hyperparameters were not properly optimised for the best results.

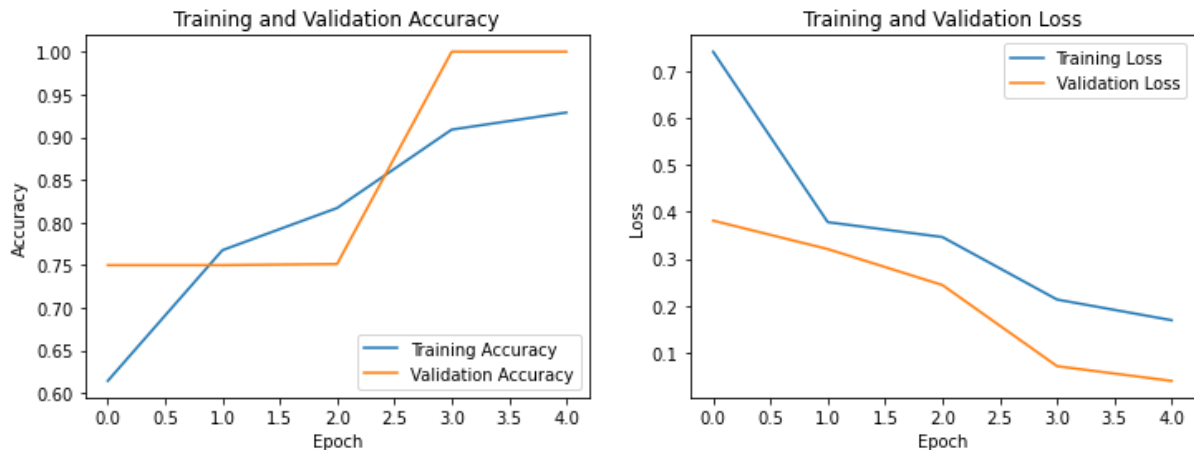


Model Testing:

The model was not able to successfully identify which image was the medium crack, and which was the large crack. It seems that the model is underfitted since it has an equal probability of all possible classes hovering around 25%.



Alternative Model:



A model with only a few epochs and poorly chosen hyperparameters can perform poorly for several reasons. With too few epochs, the model may not have enough opportunities to learn the underlying patterns in the data. Deep learning models, especially neural networks, require sufficient exposure to the training data to adjust their parameters and capture complex relationships. As a result, this may lead to underfitting, where the model fails to capture the complexities present in the data. Underfitting occurs when the model is too simplistic to represent the underlying patterns, resulting in poor performance on both the training and validation sets. Hyperparameters, such as learning rate, batch size, and architecture-related parameters, play a crucial role in the training process. If these hyperparameters are not appropriately tuned, the model may struggle to converge to a good solution. For example, a learning rate that is too high may cause the model to oscillate or diverge, while a learning rate that is too low may result in slow convergence. These reasons are possible explanations for why the model performed poorly.

Github Link:

<https://github.com/ShauryaSG/Project-2>