**Lab 7: Chihuahua or Muffin with CNN**

**Reflective Journal: Chihuahua or Muffin with CNN Workshop**

By Hisham D Macaraya

**ITAI 1378 Computer Vision**

Spring 2024

Professor Patricia McManus

October 10, 2024

**Reflective Journal: Chihuahua or Muffin with CNN Workshop**

**CNN Architecture**

Convolutional Neural Networks (CNNs) are a specialized type of neural network primarily used for image data processing. In this lab, I used a CNN to classify images into either "Chihuahua" or "Muffin." The CNN architecture consists of multiple layers, including convolutional layers, pooling layers, and fully connected layers. The convolutional layers are responsible for feature extraction, using filters to identify edges, textures, and more complex features as we move deeper into the network. This approach differs from the traditional neural network used in the previous workshop, which consisted solely of fully connected layers, making it less effective for image data.

Compared to traditional neural networks, CNNs are far more efficient at capturing spatial relationships in images due to their convolutional and pooling operations. Traditional neural networks treat image data as one-dimensional, disregarding spatial information, whereas CNNs maintain this structure, leading to better accuracy and more effective learning from image data (Goodfellow, Bengio, & Courville, 2016).

**Model Performance**

The model performance was evaluated using metrics such as training and validation accuracy, as well as loss. During experimentation, I observed that the CNN achieved higher accuracy compared to the traditional neural network model. In particular, adding an additional convolutional layer improved the model’s ability to learn complex features, thereby increasing validation accuracy by about 5% compared to the original architecture.

Interestingly, I noticed patterns in the misclassifications. For instance, the model occasionally confused muffins with Chihuahuas that had similar textures or colors. This highlights the importance of augmenting the dataset with varied backgrounds and lighting conditions to help the model generalize better. Misclassifications typically occurred for images with overlapping features, indicating a need for a more complex architecture or better-quality data for training.

Below is a screenshot of the model's output accuracy and loss curves, which illustrate the training and validation progress.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

The image below shows the output with a Validation Accuracy of 1.

A collage of a dog and a muffin

Description automatically generated

**Comparison**

When comparing the CNN to the traditional neural network, the CNN significantly outperformed in both accuracy and generalization. The traditional neural network struggled to achieve more than 60% accuracy, while the CNN surpassed 80%. In terms of training time, CNNs required more computational power due to their depth and the number of parameters. However, the increased training time was justified by the notable improvement in performance and ability to generalize to unseen data.

The fully connected architecture used in the previous workshop lacked the capability to effectively capture spatial hierarchies, resulting in a model that quickly overfitted and did not generalize well to validation data. On the other hand, CNNs, due to their hierarchical feature extraction process, could better differentiate between subtle variations within the images (Simonyan & Zisserman, 2015).

**Challenges and Solutions**

One of the challenges I faced during the lab was setting the correct hyperparameters, such as the learning rate and number of epochs. Initially, the learning rate was too high, causing the model's loss to oscillate instead of converging. To address this, I decreased the learning rate, which stabilized the training process and improved accuracy. Another challenge was the ModuleNotFoundError for certain Python packages, such as torchsummary, which I resolved by installing the missing dependencies using pip.

I also experimented with the model by adjusting hyperparameters and modifying the architecture. Specifically, I tried different configurations, such as varying the number of epochs, changing the learning rate, and adding an extra convolutional layer. These experiments helped me better understand how each parameter influences model performance. For example, increasing the number of epochs improved accuracy, but also required more training time. Adding an extra convolutional layer helped the model capture more complex features, which ultimately improved validation accuracy.

Moreover, I encountered overfitting during one of the experiments. To mitigate this, I added data augmentation techniques like random rotations and horizontal flips, which increased the model's robustness to variations in the training images. These augmentations helped reduce overfitting and improved validation performance.

**Real-World Applications**

CNNs have a wide range of real-world applications in image classification tasks. The model used in this lab could be adapted for various purposes, such as distinguishing between different dog breeds, identifying plant diseases, or recognizing medical images. In industries like healthcare, similar CNN-based models are used for detecting anomalies in X-rays or MRIs, helping doctors make more informed decisions (Esteva et al., 2017).

In addition, applications like autonomous vehicles rely heavily on CNNs for object detection and classification to identify pedestrians, road signs, and other vehicles, ensuring safe navigation. Image classification using CNNs is also integral to facial recognition technology, widely employed for security and user authentication purposes.

**Ethical Considerations**

When developing and deploying CNN-based image classification models, there are several ethical considerations. Bias in the dataset is a critical issue. If the training data is not diverse enough, the model may exhibit biased behavior, leading to inaccurate or unfair results when deployed in real-world settings. For instance, in facial recognition systems, an imbalanced dataset could lead to lower accuracy for certain demographics, which raises ethical concerns about discrimination (Buolamwini & Gebru, 2018).

Another ethical concern is privacy. Image classification models used for surveillance or facial recognition may inadvertently violate individuals' privacy rights. It is crucial to ensure that these models are deployed with proper consent and adhere to legal regulations regarding data usage and privacy. Developers must be aware of these implications and strive to create unbiased, fair, and transparent models that respect user privacy.

**References**

Buolamwini, J., & Gebru, T. (2018). Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. *Proceedings of Machine Learning Research*, 81, 1-15.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), 115-118.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations (ICLR)*. Retrieved from <https://arxiv.org/abs/1409.1556>