3CB107 - AI Report

Luke Munday

School of Science, Technology and Health

Faculty of Computer Science

York St John University

Module Supervisor Mike O'Dea

4 May, 2021

Table of Contents

Chapter 1

Introduction

1.1 Problem Statement

Over the past decade, the amount of content that is uploaded to online video platforms and streaming sites has exploded. From June 2007 to May 2019, the amount of hours of video uploaded to YouTube every minute has skyrocketed from 6 to 500 hours, according to Statista, n.d.

This exponential increase in content has also lead to an increase in the amount of people who post content of either themselves or others committing illegal acts. One such event, was the 2017 Chicago torture incident, where 'the victim was kidnapped and physically, verbally, and racially abused,' as documented by Wikipedia, n.d. The entire crime was live streamed to Facebook, marking it as a 'live streaming crime.'

1.2 Project Overview

In these live streaming crimes, the perpetrator will typically record from their own perspective, meaning although there is a limited view of their person, there is a high probability they will show their hands on camera. Although hands can differ a great deal, from person to person, there are certain key characteristics that appear in people of the same gender.

Being able to identify and track such persons, from limited information and imagery, is essential to determine who these individuals are. Therefore, this project hopes to classify the gender of a person using these identifiable characteristics, from an image of either the palm or dorsal view of their hand.

Chapter 2

Design

2.1 Convolutional Neural Networks

Levi and Hassncer, 2015, used a CNN (Convolutional neural network), a form of Deep Neural Network commonly used to analyse imagery, to demonstrate gender classification from facial features. Gender classification is a binary problem, with the output being male or female.

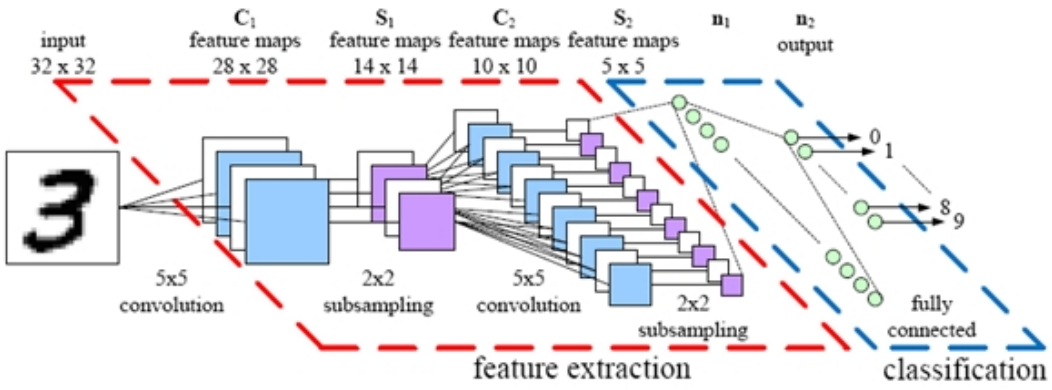

Figure 2.1: An example of CNN architecture.

This particular type of Deep Learning model can extract features from images and identify them based on those features. Whereas Neural Networks are typically constructed with hidden fully-connected layers, CNNs use multiple convolutional layers and at least one fully connected layer. Figure 2.1 shows an example of CNN architecture.

2.2 Current CNN Models

This project would make use of pre-trained models that already exist, by fine tuning them. The Visual Geometry Group (VGG) network architecture, suggested by K. Simonyan and A. Zisserman for Large-Scale Image Recognition, is one such model. This particular model 'achieved a 92.7% top-5 test accuracy in ImageNet,' which has a dataset of over 14 million images.

However, there are some disadvantages to this model, the most notable of which is that it takes a long time to train and that the architecture weights are very high.

Another model that will be investigated in this project is ResNet. Overfitting is a common problem caused by too much depth in the mode, but ResNet utilises residual blocks, which allows for a high amount of depth without sacrificing the performance.

The final model that will be used is SqueezeNet, which was first used in 2016. This network focuses on reducing the number of parameters, while maintaining a high level of precision. All of the 3x3 convolutional filters are replaced with 1x1 filters and the number of input channels is reduced.

2.3 11k Hands

The project will make use of the 11k Hands dataset, provided by Afifi, n.d., which is available for academic use. The 11k dataset is 'a collection of 11,076 hand images (1600 x 1200 pixels) of 190 subjects, of varying ages between 18 - 75 years old.'

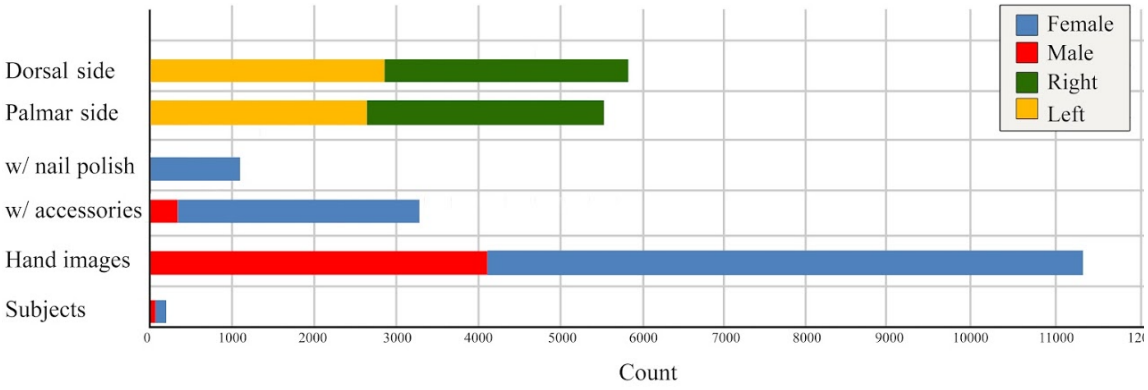

Figure 2.2: 11k Hands dataset statistics.

The dataset also provides relevant meta data, which will be used to divide the set into an 80-20 split, for training and validation respectively. The photos will be split by subject, so the model is learning to look for distinct features and traits of each gender, rather than the specific hands of a subject.

Many scholarly articles agree that the 80% training and 20% validation is one of the most commonly used splits to partition data. Korotcov et al., 2017 has demonstrated this in use with a Deep Neural Network learning model, where the dataset was randomly split, while maintaining equal proportions.

Elgallad and Ouarda, 2019 suggests the idea of using a Convolutional Wavelet Neural Network for gender classification, with 'SqueezeNet acting as a tool for unsheathing features, and Support Vector Machine (SVM) operating as discriminative classifier.'

In their report, they achieved a recognition rate for all dorsal images between 97.57% and 99.71% and a rate between 96.86% and 98.57% for all palmer images, in the 11k dataset.

2.4 Training Strategy

Mikołajczyk and Grochowski, 2018 focuses on one of the most common problems in machine learning, which is the lack of adequate training data and unequal class balance. One such approach explored in the paper is through data augmentation.

Their paper focuses on a variety of data augmentation techniques, mainly 'in the task of image classification, starting from classical image transformations like rotating, cropping, zooming.'

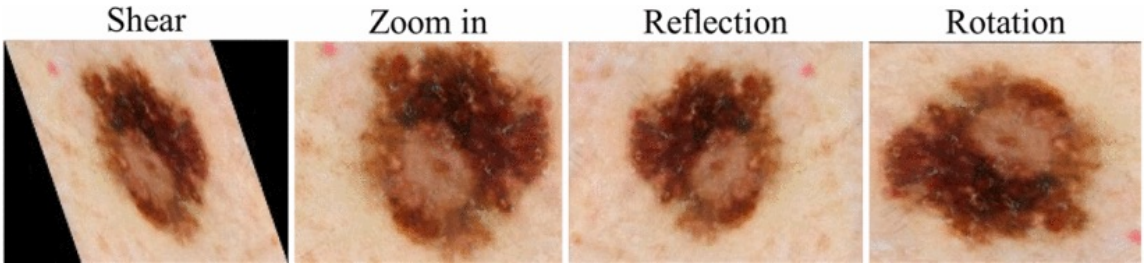

Figure 2.3: Basic image transformations.

Listed below is the code for applying such transformations to the training and validation datasets.

```
train_transformation = transforms.Compose([
    transforms.RandomResizedCrop(transform_image_size),
    transforms.RandomHorizontalFlip(),
    transforms.ToTensor(),
    transforms.Normalize(
        [0.485, 0.456, 0.406],
        [0.229, 0.224, 0.225])
])
```

The line 'transforms.RandomResizedCrop(224)' is used to extract a random patch of the image, with a size of (224, 224) and the horizontal flip transfomartion, 'transforms.RandomHorizontalFlip(),' will increase the number of unique images by a factor of 2. The values in the normalise matrix were pre-computed specifically for the 11k dataset and will scale the input data accordingly.

```
valid_transformation = transforms.Compose([
        transforms.Resize(transform_image_size),
        transforms.CenterCrop(transform_image_size),
        transforms.ToTensor(),
        transforms.Normalize(
            [0.485, 0.456, 0.406],
            [0.229, 0.224, 0.225])
])
```

By inserting subtly tweaked versions of preexisting data, we will massively expand the amount of data available to train and validate the models.

Chapter 3

Implementation

3.1 Datasets

The 11k datasets provides a metadata file for each of the subject's images. It contains information such as their name, age, gender, skin tone, accessories, nail polish, hand aspect, image name and any anomalies.

The data set is then randomly sampled in an 80-20 split, with the images sorted by both a component of hand aspect and gender. The models may then be trained solely on the palmer, dorsal or both views.

To do so, each line of the file is read in and the pertinent data is allocated to a dictionary. This is then used to place a copy of the image into the relevant dictionary, as shown below.

```
def get_metadata(line):
    line_info = line.split(",")
    info_dict = {
        'subject' : line_info[0],
        'gender' : line_info[2],
        'hand_aspect' : line_info[6],
        'image_name' : line_info[7]
    }
    return info_dict
```

We may eradicate any bias invalidation by randomly sampling by subject and separating the training and validation subjects apart. As a result, the smaller dataset would have little effect on testing raw hand accuracy.

3.2 Model Training

Too et al., 2019 investigates the effects of deep CNN fine-tuning and evaluation for image-based classification. The paper used a variety of models to investigate this possibility, including VGG 16, Resnet and DenseNets.

'All the models except VGG 16 had accuracy above 90%,' according to the report. The models seemed to also display a steady gain in accuracy as the number of epochs increased, 'with no signs of overfitting and performance deterioration'.

The accuracy and loss for each phase are computed, with the average epoch accuracy and loss computed for each iteration. Below, the calculations for both operations are listed.

```
# Phase accuracy and loss
phase_accuracy = phase_accuracy + (torch.sum(prediction == labels.data))
phase_loss = phase_loss + (loss.item() * images.size(0))
```

The 'torch.sum()' function will sum all the given elements in a tensor, where phase accuracy is the number of correct classifications, but epoch accuracy is the percentage of phase accuracy out of the total amount of images.

```
# Epoch accuracy and loss
epoch_accuracy = phase_accuracy.double() / phase_count
epoch_loss = phase_loss / phase_count
```

Chapter 4

Results

4.1 Model Accuracy

All three of the models were run on both the palmer and dorsal datasets, with an image transformation size of 224 pixels, 15 epochs and a batch size of 8, with their corresponding highest epoch accuracy listed in Table 4.1.

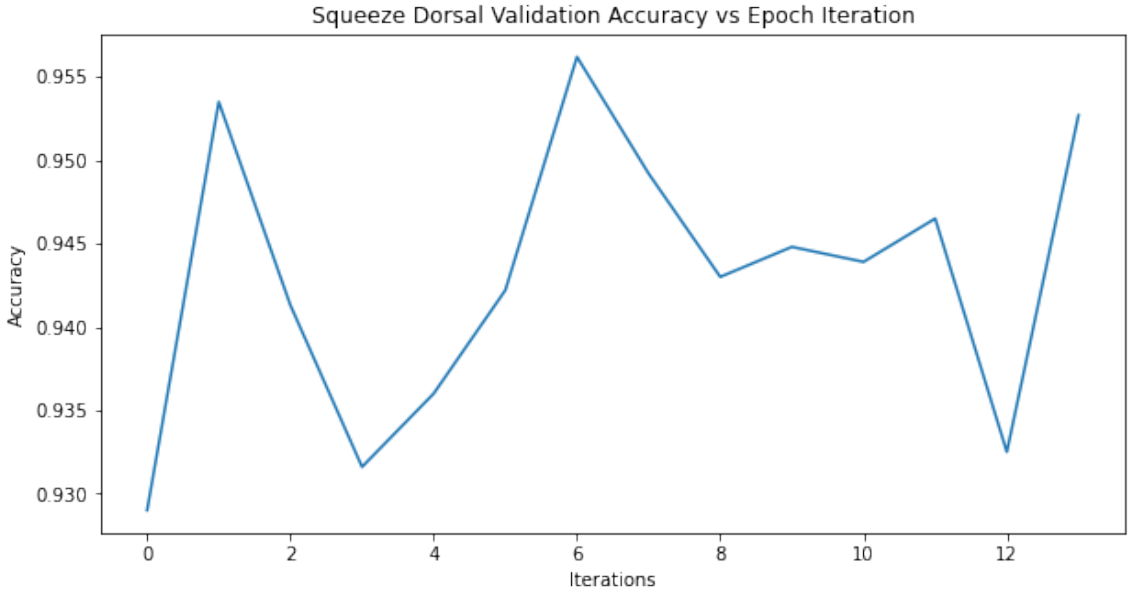

Figure 4.1: A graph of Squeezenet dorsal validation accuracy.

Table 4.1: A table showing each model's highest epoch validation accuracy for each hand type.

Model	Hand Type	Accuracy
Squeezenet	Palm	88.6%
Squeezenet	Dorsal	95.6%
Resnet18	Palm	90.6%
Resnet18	Dorsal	91.0%
VGG11 bn	Palm	84.2%
VGG11 bn	Dorsal	86.8%

4.2 Evaluation

Table 4.1, shows Squeezenet outperformed Resnet18 and VGG11 with dorsal images, while Resnet18 outperformed the others with palmer images. Overall, the experiment was a success, with high accuracy ratings on both hand types. However, it is important to keep in mind that the model can only be reliable for people who have young hands, and will be marginally biassed towards women due to the dataset's minor imbalance in subject gender.

Chapter 5

Conclusion

5.1 Future Considerations

In terms of the problem statement, the project achieved its goal by using a CNN to determine hand gender by picture. This may be taken a step forward by modifying the algorithm to evaluate the dataset with more characteristics, like age. This could help us narrow down the person we're looking to identify from the picture, but that would require a much bigger dataset of images from each of the age classifications.

Chapter 6

Links

GitHub Project:

https://github.com/LMunday98/3CB107-Assessment-Ai_CNN

Installation Guide:

https://github.com/LMunday98/3CB107-Assessment-Ai_CNN/blob/main/README.md

11k Hands Website:

https://sites.google.com/view/11khands

11k Hands Dataset:

https://drive.google.com/file/d/1KcMYcNJgtK1zZvfl_9sTqnyBUTri2aP2/view

References

Afifi, Mahmoud (n.d.). *11k Hands*. URL: https://sites.google.com/view/11khands (cit. on p. 4).

Elgallad, Elaraby and Wael Ouarda (Jan. 2019). 'CWNN-Net: A New Convolution Wavelet Neural Network for Gender Classification using Palm Print'. In: *International Journal of Advanced Computer Science and Applications* 10. DOI: 10.14569/IJACSA.2019.0100516 (cit. on p. 5).

Korotcov, Alexandru et al. (2017). 'Comparison of Deep Learning With Multiple Machine Learning Methods and Metrics Using Diverse Drug Discovery Data Sets'. In: *Molecular Pharmaceutics* 14.12. PMID: 29096442, pp. 4462–4475. DOI: 10.1021/acs.molpharmaceut.7b00578 (cit. on p. 5).

Levi, Gil and Tal Hassncer (2015). 'Age and gender classification using convolutional neural networks'. In: pp. 34–42. DOI: 10.1109/CVPRW.2015.7301352 (cit. on p. 3).

Mikołajczyk, Agnieszka and Michał Grochowski (2018). 'Data augmentation for improving deep learning in image classification problem'. In: *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pp. 117–122. DOI: 10.1109/IIPHDW.2018.8388338 (cit. on p. 5).

Statista (n.d.). *Hours of video uploaded to YouTube every minute as of May 2019*. URL: https://www.statista.com/statistics/259477/hours-of-video-uploaded-to-youtube-every-minute/ (cit. on p. 1).

Too, Edna Chebet et al. (2019). 'A comparative study of fine-tuning deep learning models for plant disease identification'. In: *Computers and Electronics in Agriculture* 161. BigData and DSS in Agriculture, pp. 272–279. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2018.03.032 (cit. on p. 8).

Wikipedia (n.d.). *2017 Chicago torture incident*. URL: https://en.wikipedia.org/wiki/2017_Chicago_torture_incident (cit. on p. 1).