CPSC81 Final Paper: Facial Expression Recognition Using CNNs

Luis Ceballos LCEBALL1@SWARTHMORE.EDU
Swarthmore College, 500 College Ave., Swarthmore, PA 19081 USA

Sarah Wallace SWALLAC2@SWARTHMORE.EDU
Swarthmore College, 500 College Ave., Swarthmore, PA 19081 USA

Abstract

Humans are able to perceive facial expressions instantly. This paper looks at how difficult of a task this would be for a computational algorithm. In order to perform this task, we use a Convolutional Neural Network, one of the best models for analyzing visual imagery today. We use this network to classify 7 facial expressions in total: angry, disgust, fear, happy, sad, surprise, and neutral. The goal of the Convolutional Neural Network is to get the highest accuracy in correctly classifying the images in the test set. We perform two experiments in which we compare the performance of the network against itself, and then we compare the performance of the network against peoples' skills in completing the same classification task. We find that it is easier for the network to classify two opposite emotions (happy vs. sad) in a much more simplified version of the problem than it is to classify all 7 emotions. We also find that our network performs better than people when classifying facial expressions, implying that identifying facial expressions is a difficult task.

1. Introduction

Facial recognition is becoming a larger and larger area of research and experimentation today. Recognizing faces is a task that humans do not need to think twice about. We can identify someone we know by their appearance instantly. However, in the context of computer vision, facial recognition is a very difficult task for computers to perform.

There are many practical applications that computers can use facial recognition for. Many large companies such as Google and Facebook make use of facial recognition in intelligent ways. In Google Photos, Google uses facial recognition in a semi-supervised learning way to allow users to search for someone's name, and every image with that person in it will show up. Facebook uses facial recognition to intelligently tag people in photos without having the user specify who the person is. There are many applications of facial recognition beyond social media, but these are just a few ways in which facial recognition is used today.

In our research, we hope to gain a better understanding of what makes a facial recognition problem difficult in a variety of contexts.

1.1. Previous Research

In 2006, Ralph Adolfs presented a paper on how humans perceive emotions. He discovered that there are many areas of the brain that are active when identifying facial expressions, including the amygdala, temporal cortex, and superior colliculus. These different brain structures interact at various points in time and often as a function of context and individual differences. He also discovered that fear and disgust in particular are processed differently in the brain than other facial expressions. In identifying these emotions, the amygdala is disproportionately important for processing facial expressions of fear and disgust (4). Since identifying facial expressions is such an active and involved process for the human brain, it will be interesting to see how a computer deals with performing this task.

In 2000, Guodong Guo, Stan Z. Li, and Kapluk Chan propose a way to perform face recognition tasks using Support Vector Machines (SVMs). At the time, SVMs were recently proposed as a new technique for general purpose pattern recognition. Given a set of points belonging to two classes, a SVM finds the hyperplane that separates the largest possible fraction of points of the same class on the

CPSC 81 Adaptive Robotics Final Report, Swarthmore College, Spring 2017.

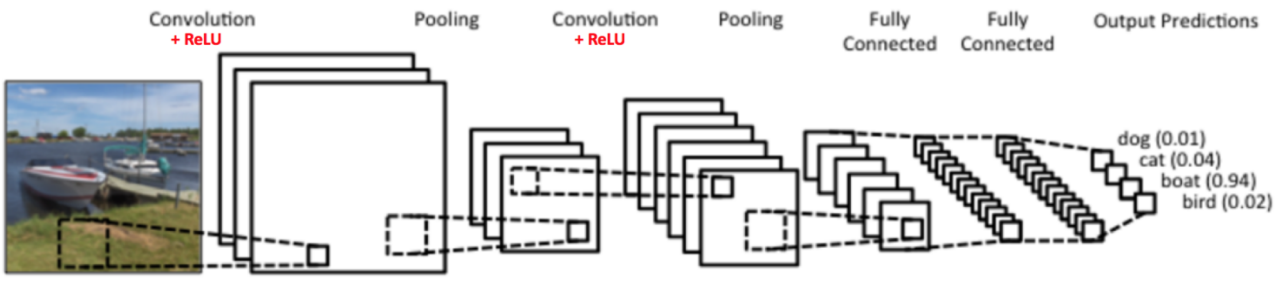

Figure 1. A simple Convolutional Neural Network.

same side, while maximizing the distance from either class to the hyperplane. This hyperplane minimizes the risk of misclassifying not only the examples in the training set, but also the unseen examples of the test set (3). They show that SVMs are an effective way to perform classification tasks, yet they serve a more general purpose rather than specializing on images.

Steve Lawrence, C. Lee Giles, Ah Chung Tsoi, and Andrew D. Back propose another way to perform facial recognition tasks using Convolutional Neural Networks (CNNs). Using their CNN, they were able to minimize their error by training on just five images per person to 3.8%. One of the main benefits of using a CNN is that the images require little preprocessing given that CNNs are meant to analyze images. They are also able to extract high level and low level features of each face, allowing their model to generalize well to their test data (5). Overall, they found that CNNs are a very successful method for performing the facial recognition task.

1.2. Our Experiments

For our experiments, we want to explore the facial recognition problem as well, but in a much simpler setting. Instead of attempting facial recognition for specific people, the main goal of our experiments is to run a classification task to identify different facial expressions. There are 7 total facial expressions that we are hoping to classify: angry, disgust, fear, happy, sad, surprise, and neutral.

There are two main experiments that we want to run. The first experiment is comparing the results between a model that classifies all 7 facial expressions and a model that classifies just 2 of these facial expressions: happy vs. sad. By comparing the accuracies of the models in this experiment, we will be able to see if it is an easier task to classify just two expressions instead of all 7, and why this might be.

The second experiment is comparing the results of our model that classifies all 7 facial expressions to the results of having humans classify the same images. This experiment will be looking more directly at how difficult of a task it is to classify facial expressions. People do this everyday as they interact with a plethora of individuals, yet it remains a difficult task. If this is a difficult task for humans to perform, then how can we expect our model to be able to perform this same task just as effectively? By comparing these results, we will get a better understanding of the difficulty and realistic expectations for our model to perform this classification task.

In order to perform these experiments, we will be implementing a CNN. In the next section we will explain why this is the best choice.

1.3. What is a Convolutional Neural Network (CNN)?

As the main method for our classification task for facial expression recognition, we are using a CNN. A CNN is a feed-forward artificial neural network that is known to be successful at at analyzing images.

There are many advantages to using a CNN for image analysis. The main reason is that CNNs are designed for analyzing images. CNNs are also very easy to train and are able to generalize well once they are trained. CNNs also make use of local connections, shared weights, pooling, and many layers in order to classify images.

The key idea behind neural networks is that they learn non-linear decision boundaries using hidden layers and correct their errors through back propagation. Neural networks in general receive an input (a single vector) and transform it through a series of hidden layers. Each layer is made up of a set of neurons, where each neuron is fully connected to all neurons in the previous layer, and where neurons in a single layer function completely independently and do not share any connections. The connections between nodes contain weights that are derived from the data set used for training. As the network is trained, the weights are updated to match the training input. The last fully-connected layer is called the "output layer." In classification tasks, this layer provides the classification result. Regular neural networks dont scale well to full images be-

cause each layer is fully connected to the next, meaning that the total amount of weights quickly grows with each layer added to the network. Therefore, this full connectivity can be very wasteful.

CNNs, specifically, take advantage of the fact that the input typically consists of images, so they constrain the architecture to match the conditions of analyzing an image. The main aspect that is different from a regular neural network is that the layers of a CNN have neurons that are arranged in 3 dimensions: width, height, and depth. The neurons in a layer will be only connected to a small region of the layer preceding it instead of all the neurons, as would be the case in a regular neural network. The final output layer will have a significantly smaller amount of dimensions because the architecture of the CNN will reduce the full image into a single vector of scores, arranged along the depth dimension.

As previously described, a CNN is a sequence of layers, and every layer transforms one set of activations to another through a differentiable function. There are three main types of layers: Convolutional Layer, Pooling Layer, and Fully-Connected Layer. Here are descriptions of each type of layer in more detail for a simple CNN:

- Input Layer: The Input Layer will hold the raw pixel values of the image. For example, if our images have the dimensions [32x32x3], then the width is 32, the height is 32, and there are 3 color channels R, G, B
- Convolutional Layer: A Convolutional Layer will compute the output of neurons that are connected to local regions in the inputeach computing a dot product between their weights and a small region they are connected to in the input volume. For our example, this would result in volume [32x32x10] if we choose to use 10 filters.
- Pooling Layer: A Pooling Layer will perform a downsampling operation along the spatial dimensions (width, height), resulting in volume such as [16x16x12].
- Fully-Connected Layer: A Fully-Connected Layer will compute the output scores, resulting in volume of size [1x1x7], where each of the 7 numbers corresponds to a possible outcome, which for our experiment will be a facial expression. Each neuron in this layer will be connected to all the neurons in the previous layer.

In this way, CNNs transform the original input image layer by layer from the original pixel values to the final class outputs (2). Figure 1 shows an example of a typical CNN using these layers.

1.4. Hypothesis

As previously stated, our first experiment is comparing the results of classifying all 7 facial expressions with the results of classifying just 2 facial expressions (happy and sad). For this experiment, we hypothesize that the accuracy of classifying 7 facial expressions will be lower than the accuracy for classifying just happy and sad. We believe that classifying all 7 emotions will be the hardest problem because the model may learn some features that overlap across multiple emotions. For example, the model could learn that frowning is a feature of both sad and angry from the training data. Also, we believe that classifying happy vs. sad should be a relatively easy problem because happy and sad are such opposite emotions. Happy is clearly a positive emotion, and sad is clearly a negative emotion. It would be much easier to misclassify anger vs. disgust, for example, because these are both negative emotions. However, if the model can only choose between happy and sad, it should be relatively difficult to misclassify these emotions.

Our second experiment is comparing the results of our model classifying all 7 facial expressions with the results of people classifying all 7 facial expressions. For this experiment, we hypothesize that peoples' accuracy in labeling these images correctly should be higher than our model's accuracy. We believe that people should be very good at identifying facial expressions because we do this everyday when we interact with other people. We have grown up reading people's faces and deciphering emotions, so this task should be pretty easy for people to do.

2. Experimental Methods

2.1. Data Set

We obtained the data set that we used to train and test our CNN from the Kaggle website that was initially used for one of their competitions. The competition is called "Challenges in Representation Learning: Facial Expression Recognition Challenge," and it took place 5 years ago (1). This data set contains gray scale images that are each 48x48 pixels. The faces in each image have been automatically registered so that the face is more or less centered and occupies about the same amount of space in each image. For the purposes of the competition, the data set was split up into three groups: a private test set, a public test set, and a training set. For the purposes of our experiment, we combined all three data sets into one, for a total of 35,887 images.

This data set contains 7 different facial expressions as previously discussed: angry, disgust, fear, happy, sad, surprise, and neutral. In total, there are 4,953 images of angry, 547 images of disgust, 5,121 images of fear, 8,989

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Total	4,953	547	5,121	8,989	6,077	4,002	6,198
Test Set	958	111	1,024	1,774	1,247	831	1,233
Training Set	3,995	436	4,097	7,215	4,830	3,171	4,965

Figure 2. This table shows the distribution of our data set for the first experiments with the 80/20 split. The first row shows the total number of images for each expression. The second and third rows show the total number of images in the test set and training set.

Figure 3. Some example images from our data set.

images of happy, 6,077 images of sad, 4,002 images of surprise, and 6,198 images of neutral. Reference Figure 2 for the distribution. As we can see from the numbers for this distribution, we have the most amount of data for happy, sad, and neutral, and we have the least amount of data for disgust. This could imply that our model will learn how to classify happy, sad, and neutral very well because it will be able to learn from processing more data. This also implies that our model may not learn how to classify disgust well because there is much less data for our model to learn from.

Each image contains a face with one of the 7 facial expressions. However, there is a lot of variation in the faces shown in each of the images. In Figure 3 we can see some example images from our data set. There is a wide variety of ages that are represented in our data set, from babies to adults. There are also some obstructions in the images, such as the man's hand covering part of his face in the image for fear. Also, we discovered that not all of the images in the data set are of real people. There are a few images that contain faces of cartoons depicting one of the facial expressions. This wide variety in the data set is a good thing. Having so many types of images in our training set will allow our CNN to be trained on a wide variety of images. The model will hopefully be able to generalize better to future images, which is the goal of all machine learning tasks.

2.2. Architecture of our CNN

In the introduction, we showed an image of what a typical CNN looks like. After much experimentation, the structure that gave us the best results was one that has a similar simple structure. In order to create our CNN, we used Conx, which is built on top of Keras.

In Figure 4 the CNN that we created is displayed. At the bottom of the figure, we have our input layer that takes in each image one at a time. Next, our CNN consists of two convolutional layers that are each followed by a pooling layer. After these four layers, we have a flattening layer that transforms our 2-dimensional output from the second pooling layer into a 1-dimensional vector. After the flattening layer there is a dropout layer. A dropout layer acts as a regularizer because it forgets a specified percentage of the pixel values when an image propagates through the network. In this way, the dropout layer prevents overfitting because if the network sees the exact same image again, it won't remember it because it forgot some percentage of the pixel values the first time the image propagated through. Then right before the output layer we have three fully connected layers.

We went on to perform our classification tasks with this neural network.

2.3. Classifying 2 Emotions vs. 7 Emotions

The first experiment conducted tested the difficulty of the presented task. A classifier, like a CNN, should perform better when distinguishing between two distinct labels. In our case, the CNN had to distinguish between a face that can be classified as either happy or sad. In contrast, when having to classify a range of different labels, such as the full range of seven emotions presented in this experiment (anger, disgust, fear, happy, sad, surprise,

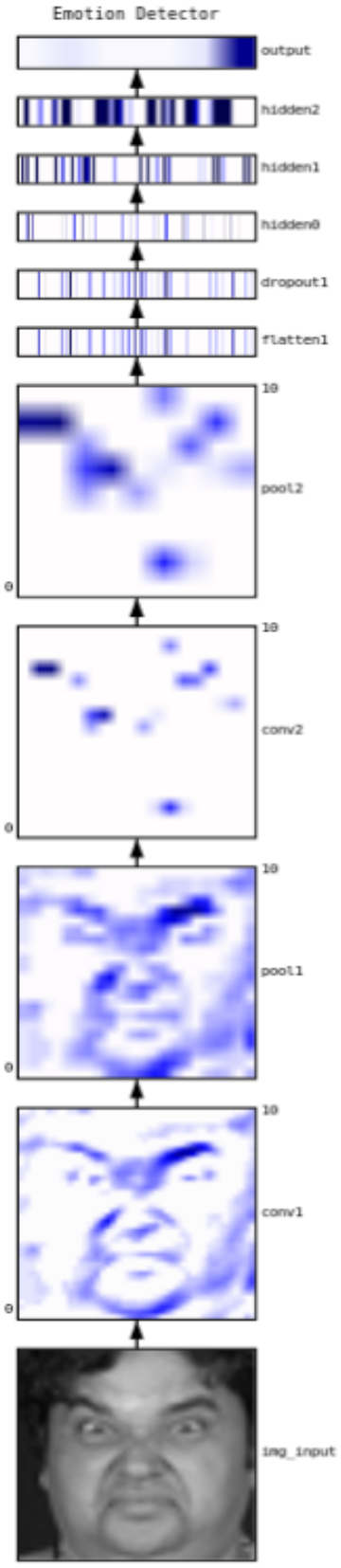

Figure 4. The CNN that we created.

neutral), it should perform with less accuracy. Thus, our first experiment tested our classifier's performance between classifying happy vs. sad and classifying all 7 emotions.

In order to train and test the classifier for the happy vs. sad labeling task, the data set was manipulated such that the classifier would only be processing images labeled as either happy or sad. The images were copied from the original data set containing all 35,887 images into a new data set and split 80/20: eighty percent of the happy vs. sad data set was used for training and twenty percent was used for testing the classifier. For this training set, there were 7,219 happy images and 4,833 sad images. After training, our model was ready for the testing set, which consisted of 1,770 happy images and 1,244 sad images. The performance of this experiment was then compared to our seven emotion classifier.

The seven emotion classifier used all of the original data set. The original data set containing all 35,887 images was also split 80/20. For this training set, there were 3,995 anger images, 436 disgust images, 4,097 fear images, 7,215 happy images, 4,830 neutral images, 3,171 sad images, and 4965 surprise images. After training, our model was ready for the testing set. The test set consisted of 958 anger images, 111 disgust images, 1,024 fear images, 1,774 happy images, 1,247 sad images, 831 surprise images, and 1,233 neutral images.

It is worth noting that when labeling two distinct emotions, it is easier to see the different features that the model creates when attempting to generalizations for the labeling task. Thus, classifying happy vs. sad aided in the building of the CNN that was implemented throughout all of our experiments.

2.4. Our CNN vs Swarthmore Students

The second experiment measured our network's performance when compared to the performance of Swarthmore students. For this particular experiment, a new data set was created from the original data set for the training and testing. We hand-selected twenty-one images from our original data set, picking three images for each of the emotions. These images created our new test set (see Figure 5). The three images for each emotion ranged in difficulty.

We then continued to create our training set. This consisted of every single image other than the twenty-one images selected for the test set. The model is meant to compete with Swarthmore College students, so we wanted to make sure that the model had as much information to process in order to create the best features it could for generalizing a relationship between facial expressions and emotions. For this training set, there were 4,950 anger images, 544 disgust images, 5,118 fear images, 8,986 happy im-

Figure 5. The test set of 21 images for our second experiment.

ages, 6,074 neutral images, 3,999 sad images, and 6195 surprise images. After training the model, we were able to use propagate the test set through the CNN and observe how many images the model classified correctly.

We compared our model's success at classifying these twenty-one images with the success of 118 Swarthmore students. To achieve this, we created a Google Form. The form consisted of each picture, followed by a question that asked the participating student to label it as portraying one of the seven emotions. This form was posted on a public Facebook page housing over 1,000 Swarthmore students, in which they could participate at their own will. A total of 118 anonymous submissions were collected from the Swarthmore student community. The students were not told how many images of each label were present. They were given minimal information, simply being asked to label the images.

3. Results

For the first experiment, our model performed as predicted, being able to classify two distinct emotions better than a range of seven. The training time for the model to reach max training accuracy and minimum loss for classifying the training took 20 epochs. On the other hand, when having to differentiate between seven emotions, the classifier needs 50 epochs to converge to its max training accuracy and minimum loss.

After training the network, we were able to propagate an image through the CNN and see some of the features that it created. The results obtained for classifying happy vs. sad reached a max accuracy of 84%. However, when propagating an image through the network trained for classifying between seven different emotions, the network performed with an accuracy of 51%.

For our second round of experiments, we saw our model attempt to perform against Swarthmore students. The model was trained for 50 epochs, as that is when it converged to its best training accuracy and minimum loss. After training, the model was ready to be tested with the twenty-one image test set.

The features that the model created establishes relationships between facial features and an emotion. If you look at Figure 4, the first convolution layer and pooling layer began to create features that highlighted and honed in on specific areas of the face. In this case, it can be noted that this particular feature created for this image focused on the position of the eyebrows, the curvature of the mouth and the different folds created in the gentleman's face to exhibit this feeling of disgust. However, in the second convolution layer, it is hard to tell what the model actually hones in on, and the clear distinction of different facial features disappears. This is an example of only one of the features created for this particular image.

The network learned to create relationships between facial features and emotions, performing with a max accuracy of 57% when classifying all 7 emotions. This was slightly better than the performance by an average Swarthmore student. The way students performed individually ranged from correctly labeling 6 images to correctly labeling 14 (Figure 7). The median number of images classified correctly was 10. Not a single student was able to correctly label all 21 images. On average, the performance of a typical Swarthmore student came out to be 49% for this particular classifying task.

	Accuracy
CNN: 7 Expressions 20% Test Set	51%
CNN: Happy vs. Sad 20% Test Set	84%
CNN: 7 Expressions 21 Image Test Set	57%
Humans: 7 Expressions 21 Image Test Set	49%

Figure 6. Table shows the accuracy associated with each part the experiments.

4. Discussion

As expected for our first experiment, the model performed better when classifying happy vs. sad than when classifying all seven emotions. The amount of time it took to train the network for each of the tasks hinted at the difficulty of the problem. For happy vs. sad, the network only required 20 epochs of training, as it is easier to create relationships between a face portraying a happy emotion or a sad emotion. Happy and sad are two distinct emotions that people generally represent in a similar fashion with facial expressions. This is evident in the model's performance of 84% at labeling these two distinct emotions.

In contrast, when presented with seven emotions, the model does not perform as well. These seven emotions are not as distinct, and sometimes an image may seem to portray more than one emotion. Each individual human portrays each emotion in a unique fashion. Where some emotions, like happy and sad, are extremely distinguishable, others, such as sad, disgust, and anger, may appear very similar to each other. This is one of the reasons in which we think that the model performed worse at this classification task, reaching only 51% accuracy. Even so, we were impressed that the model reached the accuracy that it did, given how difficult this labeling task is.

Our second experiment only further highlights the difficulty of this classification task. Giving the full data set as the training set, except for the twenty-one images used for the test set, improved the model's performance, bringing it up from 51% to 57%. This came to be a surprise, as we thought that the addition of a couple thousand more images into the training set would help the model create better relationships. However, although the addition of these images to the training set did improve performance, we think that the model can only improve so much with the addition of more images. This could be attributed to the fact that individuals represent emotions differently. There were images in our data set that, on first glance, appear to be emitting an emotion other than the one it is classified as. This is further supported by the performance of an average Swarthmore student.

Not only was this task difficult for our model, but it also proved to be fairly difficult for humans. Not a single student was able to classify all 21 images correctly. Not a single individual got close, with the highest number of correctly labeled images being 14. Although this performance may not be representative of humans as a whole, it does say a lot about 1) how Swarthmore students perform at this classification task and 2) the difficulty of this task.

The difficulty of this task can be attributed to a number of factors. The first reason for our model not performing better could be the data itself. If you reference Figure 2, you can see the total number of each image that was included in the whole data set. There is an uneven distribution of images among the different emotions, which could have affected how our model created relationships between facial expressions and emotions. For example, our model was probably better at classifying images that portrayed a face depicting a happy emotion given that it had so many happy images (8,989) to train on. In contrast, given only 547 images of disgust, we expect this may be a reason for the model to create a less successful relationship between an image and its label for an image portraying disgust.

Another point of error could be the method in which these images were labeled. The way these images were labeled could include some bias. This is due to this task being generally arbitrary and objective. Every individual displays emotions differently; therefore, every individual will have a different understanding of what facial expressions depict certain emotions. There is plenty of human novelty in this realm, so creating an effective generalization is a difficult task, not only for our model, but for humans in general (as displayed by Swarthmore student performance).

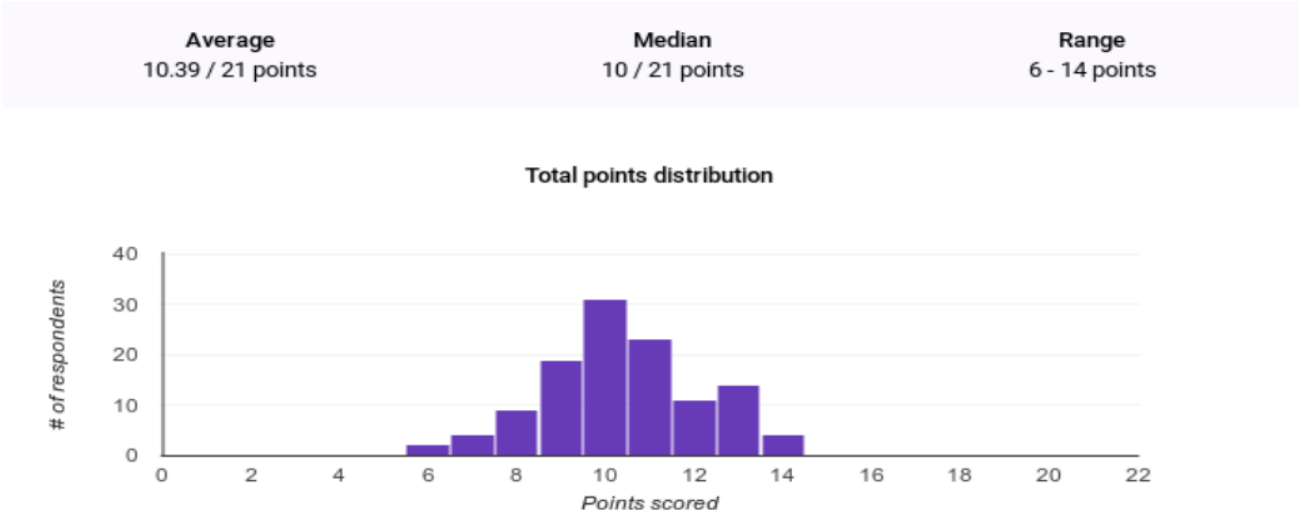

Figure 7. This graph shows how the average Swarthmore student performed when tasked with classifying 21 images into the 7 categorical emotions.

5. Conclusion

In conclusion, one of our hypotheses held to be true. Our hypothesis for the first experiment was correct in that the model was able to more accurately classify two facial expressions when compared to classifying seven facial expressions. It is generally easier for the model to create a relationship between facial expressions and two distinct emotions, such as happy and sad. On the other hand, the model struggled to create a good generalization when presented with more emotions to classify that were not as distinct (e.g. sad, angry, disgust). This proved to be true, as our network was able to correctly classify 84% of the test data for happy vs. sad, whereas it was only able to successfully classify 51% of the test data for all 7 emotions.

For our second experiment, we expected human performance to trump our model's performance, but this was not the case. By increasing the training set, the model's accuracy rose to 57%. Although not a large improvement, we found that our model was able to outperform Swarthmore students. This displayed the sheer difficulty of the task at hand, and that our model was performing fairly well.

This classification task proved to be more difficult than expected. This objective task is arbitrary; people depict different emotions via facial expressions in different ways. Humans tend to rely more on social context in order to identify certain emotions rather than just looking at facial expressions. Without that context given in the setting of our classification task, it is difficult for humans to perform this task. Since our model does not rely on this context and only relies on its training data, our model has the upper-hand in this setting.

Acknowledgments

We would like to thank Lisa Meeden who helped us tremendously with our experiments. She helped us preprocess our data, which made conducting our experiments so much easier.

References

[1] "Challenges in Representation Learning: Facial Expression Recognition Challenge," *Kaggle*. [Online]. Available: https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge.

[2] "Convolutional Neural Networks (CNNs / ConvNets)," *Github*. [Online]. Available: http://cs231n.github.io/convolutional-networks/.

[3] G. Guo, S. Z. Li, and K. Chan, "Face Recognition by Support Vector Machines," *IEEEXplore*, Mar. 2000.

[4] R. Adolphs, "Perception and Emotion: How We Recognize Facial Expressions," *Sage*, vol. 15, no. 5, pp. 222–226, 2006.

[5] S. Lawrence and C. Lee Giles, "Face Recognition: A Convolutional Neural-Network Approach," *IEEE Transactions On Neural Networks*, vol. 8, no. 1, pp. 98–113, Jan. 1997.