Data Collection and Real-Time Facial Emotion Recognition in iOS Apps with CNN-Based Models

Damian Valles[1*], Lois Adrianne R. Umali[2], Thomas Paveglio[1], Josiah N. Brinson[3], Mohammed Hyder[4], Gavin E. Jackson[5], Dylan Hall[5], John W. Farrell III[6], Yumeng Li[6], Semih Aslan[1], Maria D. Resendiz[2], Ting Liu[6]

[1]*Ingram School of Engineering, Texas State University, San Marcos, TX 78666, USA*
[2]*Department of Communication Disorders, Texas State University, Round Rock, TX 78665, USA*
[3]*Department of Physics, Wake Forest University, Winston-Salem, NC 27109, USA*
[4] *Department of Mathematics and Computer Science, Rhodes College, Memphis, TN 38112, USA*
[5]*Department of Computer Science, Texas State University, San Marcos, TX 78666, USA*
[6]*Department of Health and Human Performance, Texas State University, San Marcos, TX 78666, USA*
{*corresponding author, dvalles@txstate.edu}

***Abstract*—This study analyzes and compares deep learning models, including Naïve-CNN, VGG16, EfficientNetV2, and MobileNetV2, for facial emotion recognition using the FER2013, and collected Zoom datasets are presented. The paper discusses the data collection of human subjects over Zoom to gather quality samples for the seven emotions targeted with the classifiers. Preprocessing steps are considered to enhance histogram information, brightness contrast, and augmentation of both datasets. The performance of these models was evaluated based on their accuracy, precision, recall, and F1-score. An iOS app was developed to test the trained models in real-time with YouTube videos. The Naïve-CNN and VGG16 models demonstrated the highest performance, while the EfficientNetV2 and MobileNetV2 models showed potential for further improvement during training and testing. The iOS app implementation showed the expected weak results of the trained models but observed capacities that the model provided for actual utilization on mobile devices. The work provides valuable insights into the challenges faced by deep learning CNN-based models for facial emotion recognition and suggests directions for future research.**

Keywords—data collection, labeling, face emotion, CNN models, app

I. Introduction

This research examines the need for an engineering solution for facial expression recognition for children with Autism Spectrum Disorder (ASD). Communication with children frequently extends beyond verbal exchanges, encompassing nonverbal aspects such as facial expressions, body language, physical contact, eye contact, personal space, and tone of voice. Positive nonverbal communication can enhance emotional connections, while negative nonverbal communication can lead to feelings of rejection or disappointment.

Facial expressions, which arise from the coordination of 80 facial muscles, represent one of the most complex aspects of nonverbal communication to master. Ekman et al. [1] identified six universal facial expressions across all cultures - anger, disgust, fear, happiness, sadness, and surprise. This work also considers a seventh emotion in neutral. Most facial expression recognition research is based on these six expressions, with some studies also including a neutral category [2][3][4]. Accurately identifying facial expressions is a complex task for children, especially those with ASD.

This work was funded by the National Science Foundation – Research Experiences for Undergraduates (NSF-REU) under Grant #2150135.

This research presents a deep learning examination to assist children with ASD in correctly classifying facial expressions. An updated VGG16 from previous work in [5][6][7], Naïve Convolutional Neural Network (Naïve-CNN) [8], EfficientNetV2 [9], and MobileNetV2 [10] models are employed, combined with computer vision and image processing techniques, to compute facial expressions. Following the development of the primary model, an iOS app was updated and tested with emotion metrics from various individuals to assess the models' performance. The app, compatible with all iOS devices, uses its camera to detect and classify facial expressions, displaying them on-screen with emoticons when tested with YouTube video clips.

This work also presents the data collection design and considerations of recording human subjects over Zoom. The Zoom videos are sampled with frames to be labeled as subjectively accurate based on designed interview questions. Overall, the paper presents an end-to-end process, development, and evaluation of these model based on the datasets obtained.

II. Data Collection and Labeling

The Institutional Review Board (IRB) #7453 approved the data collection and labeling study. Participants were recruited from a public university that is a Hispanic Serving Institution in central Texas.

A. Data Collection

The data collection involved 208 participants interviewed by undergraduate and graduate research assistants over Zoom. The interviews consisted of two major sections: a demographic survey completion and an interview session. Participants were instructed to demonstrate each of the seven universal emotions (i.e., happy, sad, angry, fear, neutral, surprised, disgust) at the beginning and end of the interview. Four types of questions (i.e., elicitation, narration, description, comparison) were utilized as prompts to record emotional portrayal in conversation. Interviews were uploaded to a secure database. Fig. 1 shows the interview process for the participants over Zoom. Table I shows an example of the prompting done during the interviews.

B. Data Curation

Data models from interview videos were processed by research assistants into shorter clips using Adobe Premiere Pro. The interviewee's questions and remarks were removed from the

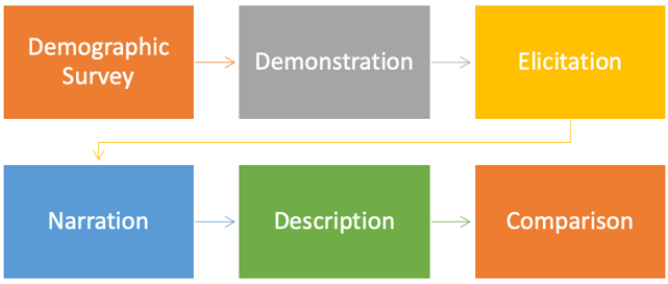

Fig. 1. The interview process for collecting emotional reactions over Zoom.

TABLE I. PROMPT EXAMPLES DURING ZOOM INTERVIEWS

Interview Section	***Prompt***
Demonstration	"Can you show me…"
Elicitation	"What's the first thing that comes to mind when I say…"
Narration	"Tell me a story about a time that made you…"
Description	"Describe what happens to a person's face when they are…"
Comparison	"What emotion do you find the hardest/easiest to show/interpret…"

videos. Each participant's answer for each question was exported as a single video clip. Curated video clips were organized into two categories: facial and speech models. Initial and final demonstrations of each emotion were utilized as facial models. Emotion portrayals in conversation are to be utilized as speech models.

C. *Data Judgment*

The current analysis only includes facial models. Curated video clips were randomized in a specific folder in a secure database. Five research assistants were instructed to independently watch and label each curated video clip with a specific emotion (i.e., happy, sad, angry, fear, neutral, surprised, disgust). Emotion labels were submitted to a designated research assistant for data analysis. A majority decision determined the final emotion label for each video clip. Research assistants convened as a group to reconcile emotional label disagreements if consensus was not reached for a video clip.

D. *Image to CSV for VGG-Model*

To convert the zoom images into a CSV format, the steps were as follows: 1) Crop the images to an *NxN* box centered on the face, 2) resize these images to 48x48x3 (height, width, BGR channels), 3) grayscaled the 48x48x3 image into 48x48x1 and encoded that into 2,304x1, 4) split the encoded images into training, public test, and private test, and 5) write the 2,304x1 images to a CSV.

When cropping the images, '*image profiles*' were used, where actors in the dataset tended to stay relatively static in the frame. Different actors had different backgrounds and used webcams of different resolutions. Taking all of this into account, to crop the entire 9,400 image FER2013 dataset to the face, a few images from each actor were passed into a program that created '*image profiles*,' a structure containing the BGR color of pre-set points in the frame, along with a crop point (x, y), crop size N, and image resolution. The crop point and size are manually found for each image profile and are a top left point above the face where an *NxN* square can enclose the entire face. The cropper then goes through each emotion-label image and compares the pre-set points' BGR color against the corresponding ones in each image profile. The sum of the 3D Euclidian distance between the image's pre-set points and the corresponding points in the image profile was calculated for each. The minimum sum was chosen as the target crop. In addition, if the current image size did not match the one designated in the image profile, that profile was skipped as the actors' webcam did not change resolution. The FER2013 dataset has a split of 0.7 for training and 0.1 for public testing and 0.2 for private testing, respectively, so the Zoom dataset will do the same.

III. PREPROCESSING

The datasets used for this work are from FER2013 and the collected Zoom interviews that captured the same emotion classes found in FER2013. The preprocessing considered the size of each dataset, the combination of the datasets, contrast enhancements, brightness adjustments, and split curation of the subjects for training and testing of the models. Given that the recorded Zoom samples are captured in the color video, extracting the frames per emotion is also considered to be in RGB format. Given the type of models used for this work, the samples were transformed to grayscale for their contrast and brightness adjustments for the data augmentation.

The first enhancement to each image sample was to convert the image to a grayscale 8-bit image and apply the Contrast Limited Adaptive Histogram Equalization (CLAHE) using OpenCV to reduce the noise when amplifying and distributing the histogram of the image. The clip limit was set to 2.0, and the tile Grid Size was set to 8x8. The CLAHE image was saved and replaced with the original image as the central sample for its indicated emotion. Fig. 2 shows an image in the middle representing the CLAHE-enhanced image of a "neutral" sample from the FER2013 dataset. The image was then increased in brightness by two levels at 1.5 and 1.75 with OpenCV. These brighter representations are the two images on the left from the center in Fig. 2. Finally, the image was decreased in brightness by two levels at 0.5 and 0.25 with OpenCV. The darker representations are the two images on the right from the center in Fig. 2. Each brightness changes were saved as a separate image sample and saved with a similar original file name. This is done due properly split the samples for training and testing phases of the models.

Fig. 2. A "neutral" emotion sample from the FER2013 dataset with its contrast variations in grayscale format.

Fig. 3 shows the same enhancements to a sample classified as "sad" from the Zoom dataset. The face representation of the subjects does not cover the entire image, which can be problematic when training the models. The sample introduces features from the background that can misrepresent as a feature

of any emotion to all the models. However, not all samples from the Zoom dataset contain background no background, as seen in Fig. 4. The sample in Fig. 4 shows a subject with the "happy" classification in its colored RGB format.

Fig. 3. A "sad" emotion sample from the Zoom dataset with its contrast variations in grayscale format.

Fig. 4. A "happy" emotion sample from the Zoom dataset with its contrast variations in RGB format.

A. *Combined Dataset*

The grayscale images have a 48x48 pixel resolution and were used to train and test the VGG-16, Naïve-CNN, and EfficientNetV2 models. This dataset combines the FER2013 and Zoom samples under seven emotion classes. This dataset helps to simplify the input and network complexity of these three models. Fig. 5 shows the Uniform Manifold Approximation and Projection (UMAP) of the train and test samples of the seven emotions of the grayscale dataset. The observation of the UMAP plot clearly shows the complexity and overlapping of the emotion classes. This will be discussed further in implementing these three models into the app. Even though the training and testing data are represented in the plot, the overlapping of the data indicates that the models will misclassify throughout all the emotion targets, given the lack of separability in the dataset. The dataset contains 225,782 image samples with a 70:30 split, with 158,045 samples for training and 67,737 samples for testing. Subject images with their enhancements were not separated for the 70:30 split to test emotions that were not previously trained in each model.

B. *Zoom Dataset*

The Zoom samples were taken as their dataset for the MobileNetV2 network. The network has a default input layer of 224x224x3 and was trained with RGB images that best help classify and extract features. The Zoom face-centered images were not grayscaled and were imported in the required resolution size. The MobileNetV2 model was used to analyze the Zoom dataset, but it was not implemented for app performance evaluation for this work.

IV. Deep Learning Models

Four CNN-based models were considered for this work: 1) VGG16, 2) Naïve-CNN, 3) EfficientNetV2, and 4) MobileNetV2.

A. *The VGG16 Model*

Visual Geometry Group (VGG) is a convolutional neural network that uses 3x3 convolutional filters. 3x3 is used because

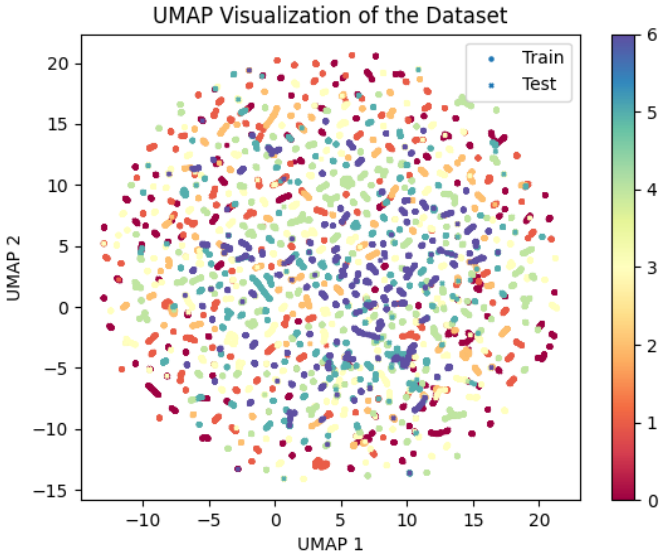

Fig. 5. UMAP plot representation of the seven emotion classes from the combined.

it is the smallest size that maintains a notion of directionality. VGG comes in two flavors, VGG16 and VGG19, the 16 and 19 representing the number of trainable layers in the network; VGG16 was used for this work. The implementation utilized Python 3.8.10 with TensorFlow 2.9.1. This model was updated from the previous work described in [1].

The described model is a VGG-inspired convolutional neural network (CNN) for image classification, constructed using Keras Sequential API. It comprises three convolutional layers, followed by batch normalization, max pooling, and dropout layers. The convolutional layers have 32, 64, and 128 filters with "ReLU" activation functions. After each pair of convolutional layers, max pooling is used to reduce the spatial dimensions, and dropout layers are employed to mitigate overfitting. The model transitions to a fully connected part composed of two Dense layers with 64 neurons and "ReLU" activation functions, accompanied by batch normalization and dropout layers. The output layer is a Dense layer with seven neurons corresponding to the number of classes, followed by an activation layer with a "SoftMax" function for multi-class classification.

B. *The Naïve-CNN Model*

The Naïve-CNN model was considered a naïve implementation of known CNN layers that provide essential feature extraction and classification outputs like those found in [2]. The consideration of the network is to understand a benchmark performance of emotion recognition through the FER2013 and Zoom image samples. Also, the naïve approach of the CNN network provides a less complex solution when implementing a mobile device such as an iPad.

The Naïve-CNN model comprises three convolutional layers, followed by batch normalization and max pooling layers. The convolutional layers have 32, 64, and 128 filters, respectively, with a kernel size of 3x3 and ReLU activation. After the convolutional layers, a Flatten layer converts the feature maps into a 1D array, followed by a fully connected

Dense layer with 512 neurons and "ReLU" activation. A dropout layer with a rate of 0.5 is employed to prevent overfitting. Finally, the output layer consists of a Dense layer with as many neurons as the number of classes, utilizing a "SoftMax" activation function for multi-class classification. The model uses image generators to preprocess and feed grayscale images from the train and test directories in batches, set at 32, resizing them to the specified target size and employing categorical labels.

C. The EfficientNetV2 Model

EmotionNetV2 was a desirable choice because it has been shown to have faster training speed and better parameter efficiency than previous models [3]. Adding layers on top of the already-trained EfficientNetV2 model aimed to refine the model to make it as accurate as possible for emotion classification. A base model was constructed using resources and starter guides on the TensorFlow Hub. The training split for the duration of this study was set to 70%, and the test split was set to 30%.

The model was constructed using the trained EfficientNetB0 model version with no included weights, average pooling value, and its input shape at 48x48x1. Its batch size was set to 32. The Input layer carries the same input shape and feeds into its dense layer of 128 nodes and the "ReLU" activation function. A dropout of 0.5 was introduced before the last dense layer. The output layer has the seven expected outputs with a "SoftMax" activation function. The optimizer was compared between the ADAM and SGD, and the SGD results are presented for this work.

D. The MobileNetV2 Model

A significant model utilized in developing the mobile application was the MobileNetV2 deep learning architecture. The primary reason for deploying this network is the provided associations of inverted residuals with linear bottlenecks [4]. This structural capability was broadly impactful as the inverted residuals have augmented layers to enable high accuracy specific to mobile and embedded vision applications. Thus, the considerable advantages of recognition in mobile applications have been widely demonstrated, with performant standards necessary for swift and efficient emotion recognition in real-time. The MobileNetV2 architecture fits well with the emotion dataset and demonstrates reliable results for the emotion recognition application.

The network was compiled using the categorical cross-entropy as the loss function and Adam as the optimizer. Hyper-tuned parameters include changing batch sizes of 44, with several epochs of 10, 50, and 100, a learning rate 0.0001, and an input image size of 224x224. Images are trained and validated to 224x224 pixels with three testing channels using 70% of the data for training and 30% for testing. Final parameters are selected based on higher training and validation accuracy and lower training and validation losses. The network is trained over 32,885 Zoom images and tested on 14,155 Zoom images via the seven emotion classes. The MobileNetV2 architecture had layers frozen for learning except for the last five, the last three, and then the final layer to check for further learning dimensionality to explore the degree of understanding of emotion context. Each three-freezing stage approached a validation accuracy of 90% well before the fifth training epoch.

V. App Process

The app's process of sending an image to the model and receiving an emotional result begins with capturing a frame from the device's camera. Once the app has taken the frame, it scans it for a face using built-in XCode and Swift features. If the frame has a face, the app will reduce the image's size to an MLModel multiarray of 1x64x64x1 with the face-centered. The app then completes the last preprocessing steps by grayscaling the image and sending it as input to the MLModel. The MLModel will return a string from its anger, happy, sad, surprise, disgust, fear, and neutral dictionary. The string is retrieved in a different method where an emoticon selected to depict each emotion is chosen based on the string given from the MLModel.

The design purpose of the app's user interface is to maintain point-and-shoot simplicity. The interface contains only three components to achieve the point-and-shoot. The first component is the camera display, which takes up the entire screen and is what the user mainly sees. The camera display allows the user to see what the device's camera is pointed towards, enabling them to ensure that the entire face of the individual is visible to the camera. The second component is the model's activation button, which contains text prompting the user to press it. This component is placed in the bottom center of the screen and works in tandem with the third component: the emotion display box. The emotion display box is located at the top center of the screen and displays the resulting emotion received from the implemented model. The overall process to use this app is to point the device's camera at another person's face and hold down the activation button.

To test the app and model, sixty videos and short clips were gathered, each representing one of the six emotions [11-46]. These short clips were then sorted into groups of ten based on the emotion they portrayed, resulting in ten videos or short clips representing each emotion. The app with a specific model was installed onto a 9th-generation iPad to commence the test. A clip from the collected dataset was played onto a monitor, and the device with the app was pointed at the screen to record the resulting emotion into a confusion matrix to test the app. This process was repeated for each model with the same videos and short clips used each time. This test aimed to grade each MLModel on its ability to correctly identify a set of emotions. This process is well-suited to the purpose because the videos gathered show a variety and diversity of conditions that may be encountered while using the app.

VI. Results

A. Naïve-CNN Results

The learning curves for the accuracy and loss training and testing lines are shown in Fig. 6. The curves show a robust fitting as the number of epochs increases to 20. The model demonstrates that it distinguishes the different classes even though the UMAP plot shows the classes tightly correlated. Fig. 7 shows the confusion matrix of the model when predicting from the test split samples. The model achieved a training accuracy of 95.43% and a test accuracy of 95.39%, demonstrating excellent performance with an overall accuracy of 95.39%, precision of 95.39%, recall of 95.39%, and an F1-score of 95.38%.

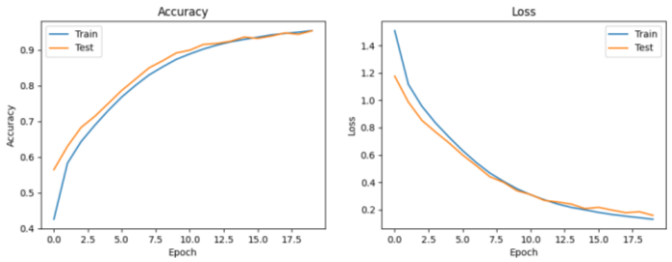

Fig. 6. The accuracy and loss training and testing learning curves of the Naïve-CNN model.

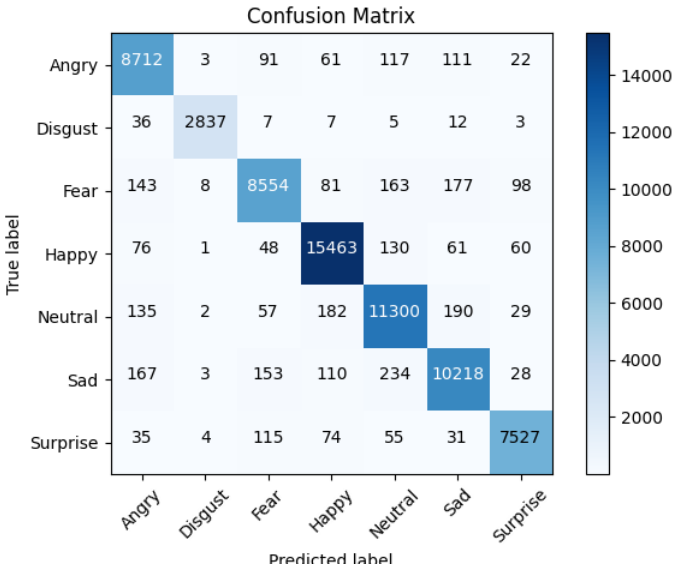

Fig. 7. The test confusion matrix results of the Naïve-CNN model.

B. *VGG16 Results*

A weakness of both the Zoom and FER2013 dataset is the class imbalance. There are far fewer samples of "disgust" than average and far more "happy" samples than average. In testing, the surplus of "happy" samples was not as big of a problem as the deficit of "disgust" samples, and "neutral"' was slightly over-predicted, so class weights of {0: 1, 1: 1.5, 2: 1, 3: 1, 4: 1, 5: 1, 6: 0.9} were used in TensorFlow's fit method as the argument of "class_weight."

Fig. 8 shows the accuracy and loss learning curves of the VGG16 model once the weights were implemented in the model. The curves show the desired fit and convergence of both curve trends for the model. Fig. 9 shows the confusion matrix of the VGG16 model as it evaluates the test split of the combined dataset. The weights provide a desired outcome of predictions for all the classes and minimize the misclassifications of emotions given the complexity of the UMAP separability of the targets. The model exhibited strong performance with an accuracy of 93.65%, an F1-score of 93.92%, a recall of 92.14%, and a precision of 95.19%.

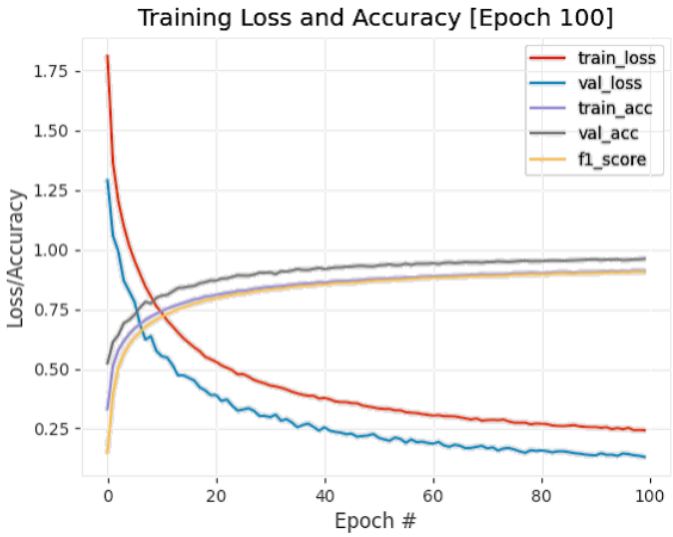

Fig. 8. The accuracy and loss training and testing learning curves of the VGG16 model.

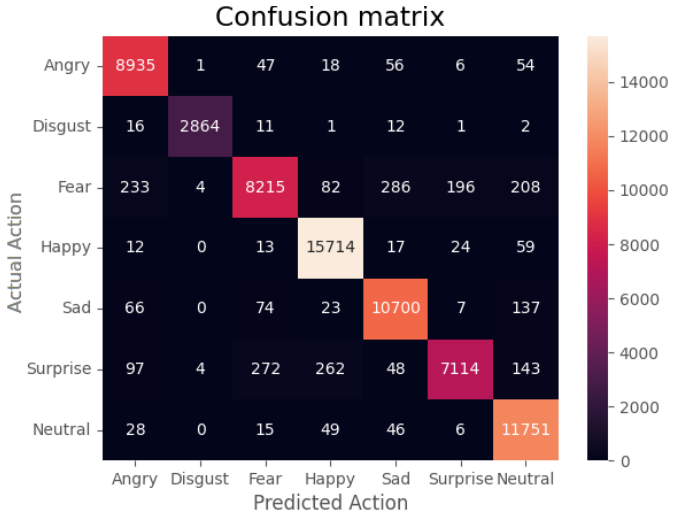

Fig. 9. The test confusion matrix results of the VGG16 model.

C. *EfficientNetV2 Results*

The EfficientNetV2 model was tested with varying datasets to gauge the initial performance of the model. The Zoom samples were not considered for the following results. The FER2013-CLAHE dataset performed relatively well compared to the other datasets and was roughly on par with the original FER2013 dataset, as shown in Table II. This suggests that the histogram equalization did not significantly impact the model's training ability. Furthermore, the one recorded test run performed on images preprocessed using principal component analysis (PCA) did not result in a good accuracy of ~25%. Therefore, only the original FER2013 dataset was used in the final set of runs.

TABLE II. FER2013 ANALYSIS WITH EFFICIENTNETV2

Dataset	*Training Acc.*	*Test Acc.*	*Final Test Loss*	*Num. of Correct Test Random Images*
FER2013	57%	63%	1.66	10 out of 16
FER2013-CLAHE	40%	57%	2.67	14 out of 32
FER2013-PCA	24%	27%	2.01	2 out of 8

The EfficientNetV2 model showed its best accuracy learning curves in Fig. 10. The curves show no overfitting behavior but lack of performance of the model. The recognition of emotions is complex between the classes as seen in the UMAP plot in Fig. 5. Also, the FER2013 dataset might not contained enough image samples to properly adjust all the weights in the model and show more of an underfit behavior even though features in each sample were CLAHE enhanced.

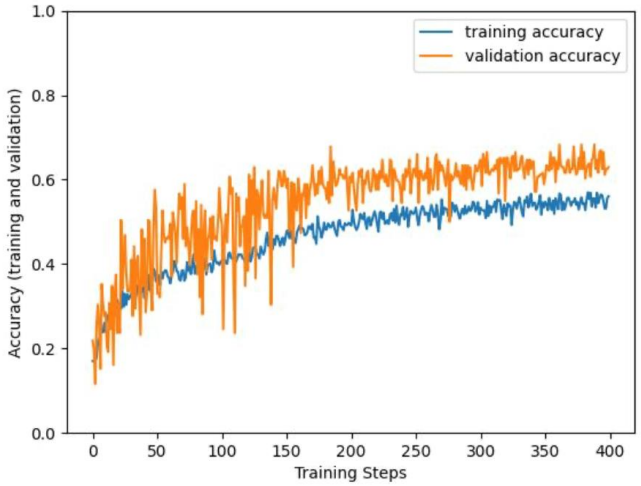

Fig. 10. EfficientNetV2 training and validation curves from the FER2013 dataset.

The combined dataset of the FER2013 and Zoom samples was then fed to the network and tested the capacity of efficiency of this type of model. Fig. 11 shows the learning curves of the model by showing the right trends of the model for its accuracy and loss. The curves were trained up to 30 epochs even though the curves had not converged. This model was tested in this level of training to observe its performance once implemented in the app. The prediction performance of the model is shown in the confusion matrix in Fig. 12. The model demonstrated good performance, achieving a training accuracy of 89.51% and a test accuracy of 87.12%, with an overall accuracy of 87.12%, precision of 87.11%, recall of 87.12%, and an F1-score of 87.04%.

D. MobileNetV2 Results

The MobileNetV2 model was trained and tested with the RGB samples from the Zoom dataset. This model was not considered to be tested on the app until it is better tuned with better-weighted considerations for training individual emotion classes. As seen in Fig. 13, the model has a solid true-positive performance for "neutral," but its overall performance for all emotion classes is well spread out through all emotions. The accuracy of each emotion was low performing from 11.3% up to 18.9% with an overall accuracy and all metrics at 14.5%.

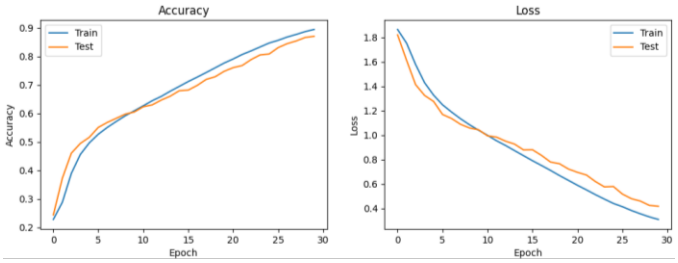

Fig. 11. The accuracy and loss training and testing learning curves of the EfficientNetV2 model using the FER2013 and Zoom samles.

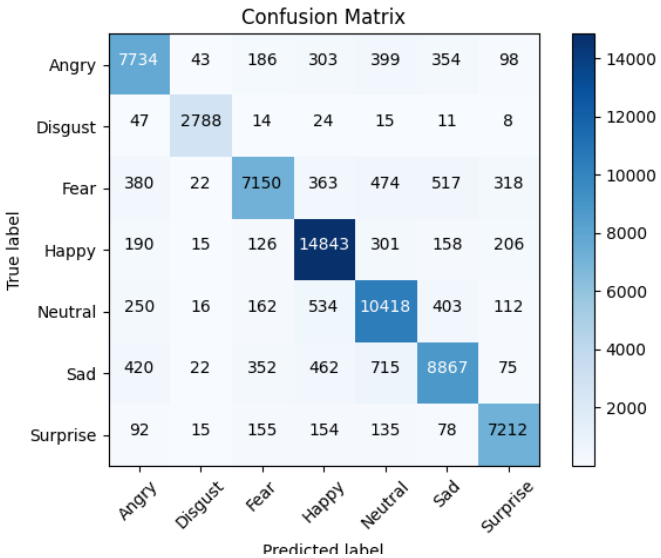

Fig. 12. The test confusion matrix results of the EfficientNetV2 model using the Zoom and FER2013 dataset.

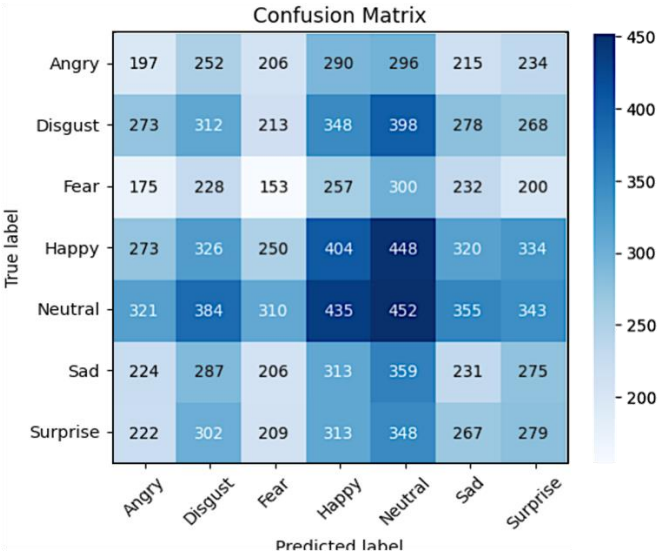

Fig. 13. The test confusion matrix results of the MobileNetV2 model using the Zoom dataset.

E. App Results

The results of the VGG model displayed a likeness and draw towards picking "happy." Fig. 14 show the classification of the trained VGG16 MLModel from the test videos. The results clearly indicate how the model can recognize a "happy" facial emotion even though it is a not a simple to recognize it. However, given the facial features of the right example in Fig. 14 show that wide mouth expression can indicate a smile and translating such reaction as a "happy" class.

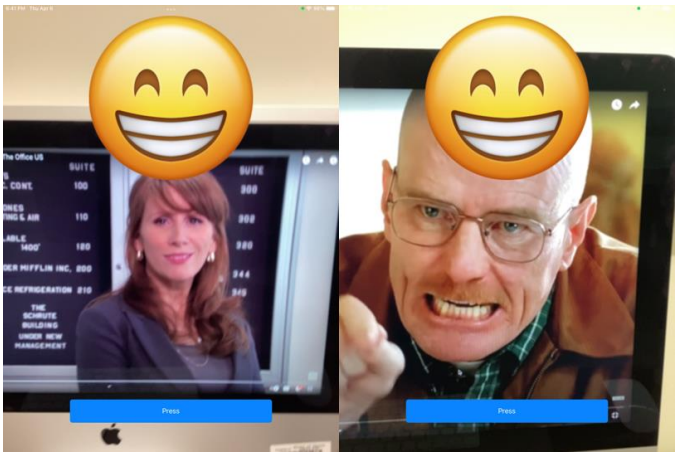

Fig. 14. The trained VGG MLModel emoticon results in the iPad showing a (left) correct "happy" emotion classification, and (right) an incorrect "happy" emotion classification.

The second model tested was the Naïve-CNN trained MLModel and had a similar tendency as the VGG16 model in which it was drawn to pick a certain emotion at a disproportionate rate. The left image in Fig. 15 shows the "surprise" emotion from an exaggerated portrayal of the emotion from an actor. These "surprise" classifications usually a not as precisely recognized given that nominal reactions are less exaggerated. The right image in Fig. 15 shows a misclassification of what the actor shows a smile "happy" emotion and the model displays "anger."

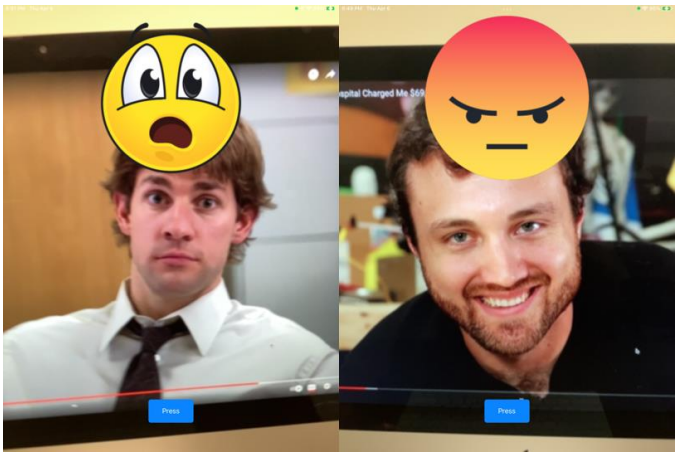

Fig. 15. The trained Naïve-CNN MLModel emoticon results in the iPad showing a (left) correct "surprise" emotion classification, and (right) an incorrect "anger" emotion classification.

The third model tested was the EfficientNetV2 model. While testing this model it evenly dispersed its results among the four emotions and entirely avoided three of them. The left image in Fig. 16 shows the "sad" emotion correctly has the actor displays a subjective "sad" emotion. The right image in Fig. 16 shows a misclassification of what the actor shows a smile indicating "happy" emotion and the model displays "surprise."

Fig. 16. The trained EfficientNetV2 MLModel emoticon results in the iPad showing a (left) correct "sad" emotion classification, and (right) an incorrect "surprise" emotion classification.

The app was tested with different YouTube videos of different show clips that show highly exaggerated facial emotion reactions from actors. The app was tested with the VGG16, Naïve-CNN, and EfficientNetV2 models. The results of the VGG16 model are shown in Table III. The results show a significant bias towards "happy" and the perfect accuracy for the "happy" emotion in its true-positive red value. The model struggled by not identifying any "disgust," "sad," or "neutral" emotions from the video clips.

TABLE III. VGG16 MLMODEL YOUTUBE TEST RESULTS

	VGG16			
	Happy	***Surprise***	***Anger***	***Fear***
Happy	**100%**			
Surprise	60%		20%	20%
Anger	70%		30%	
Disgust	50%		50%	
Fear	90%			10%
Sad	80%	10%	10%	
Neutral	40%		40%	20%

Table IV shows the results of the Naïve-CNN model when testing with the YouTube clips. The "surprise" class was the best performing true-positive recognition of the model at 60%. This model presented existent "sad" misclassifications more than those in Table III. The "neutral" emotion is more distributed, and half of its recognitions are labeled "sad." This can be true due to the complexity of neutral features required to understand the difference between sadness and neutrality without any

context for the expression. The model struggled by not identifying any "disgust," or "neutral" emotions from the video clips.

TABLE IV. NAÏVE-CNN MLMODEL YOUTUBE TEST RESULTS

	Naïve-CNN				
	Happy	***Surprise***	***Anger***	***Fear***	***Sad***
Happy	10%	60%	20%		10%
Surprise		**60%**	10%		30%
Anger	20%	40%	10%		30%
Disgust		80%	20%		10%
Fear	20%	40%		10%	30%
Sad	50%	20%			30%
Neutral	20%	10%	20%		50%

Table V shows the EfficientNetV2 results when tested with the same YouTube clips. The model's best true-positive performance was recognizing "sad" at 50%. The model could not recognize the same emotions as the last two but included the "fear" emotion. However, the model did have a more distributed classification with the four emotions and fewer gaps throughout the testing. A notable result is the misclassification of the actual "sad" and prediction of "surprise." Traits of these two emotions can resemble many facial features that can genuinely confuse the model with any context of the situation. The model struggled by not identifying any "disgust," "fear," or "neutral" emotions from the video clips.

TABLE V. EFFICIENTNETV2 MLMODEL YOUTUBE TEST RESULTS

	EfficientNetV2			
	Happy	***Surprise***	***Anger***	***Sad***
Happy	20%	40%	20%	20%
Surprise	30%	40%		30%
Anger	40%	10%	20%	30%
Disgust	20%	50%	10%	20%
Fear	20%	60%	20%	
Sad	20%	20%	10%	**50%**
Neutral	20%	20%	10%	50%

VII. CONCLUSIONS

The analysis of the Naïve-CNN, VGG16, EfficientNetV2, and MobileNetV2 models for facial emotion recognition revealed varying performance degrees. The Naïve-CNN and VGG16 models exhibited the highest performance, with overall accuracies of 95.39% and 93.65%, respectively. The EfficientNetV2 model showed good performance with an accuracy of 87.12%, while the MobileNetV2 model underperformed with an overall accuracy of 14.5%. The iOS app testing demonstrated the real-world applicability of the models, albeit with some limitations in recognizing certain emotions.

These findings highlight the potential of deep learning models for facial emotion recognition while also pointing to the need for further research to address the challenges faced by these models. Future work could focus on optimizing model architecture, addressing the class imbalance, and incorporating additional context information to improve recognition performance across a broader range of emotions and datasets.

Lastly, the data collection process must be refined to capture more distinct features that are more natural to people in person. The Zoom interviews help capture the context of emotions, but it does not capture the genuine emotion of fear or anger when the human subject is not in environments that cause these emotions in real-time. The improvements in the data collection will improve the decision-making of the classifiers.

REFERENCES

[1] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion.," *J. Pers. Soc. Psychol.*, vol. 17, no. 2, pp. 124–129, 1971.

[2] B. Gepner, C. Deruelle, and S. Grynfeltt, "Motion and Emotion: A Novel Approach to the Study of Face Processing by Young Autistic Children," p. 11.

[3] S. J. Weeks and R. P. Hobson, "The Salience of Facial Expression for Autistic Children," *J. Child Psychol. Psychiatry*, vol. 28, no. 1, pp. 137–152.

[4] R. P. Hobson, "The Autistic Child's Appraisal of Expressions of Emotion: A Further Study," *J. Child Psychol. Psychiatry*, vol. 27, no. 5, pp. 671–680.

[5] M. I. Ul Haque and D. Valles, "Facial Expression Recognition Using DCNN and Development of an iOS App for Children with ASD to Enhance Communication Abilities," *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, New York City, NY, USA, 2019, pp. 0476-0482. doi: 10.1109/UEMCON47517.2019.8993051.

[6] M. I. U. Haque and D. Valles, "Facial Expression Recognition from Different Angles Using DCNN for Children with ASD to Identify Emotions," 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2018, pp. 446-449, doi: 10.1109/CSCI46756.2018.00090.

[7] M. I. U. Haque and D. Valles, "A Facial Expression Recognition Approach Using DCNN for Autistic Children to Identify Emotions," 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 2018, pp. 546-551, doi: 10.1109/IEMCON.2018.8614802.

[8] S. Raschka and V. Mirjalili, Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2, 3rd ed. 2019.

[9] Tan, Mingxing and Le, Quoc V. (2021). EfficientNetV2: Smaller Models and Faster Training. https://doi.org/10.48550/arxiv.2104.00298 [Accessed: Mar. 17, 2023].

[10] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4510-4520.

[11] Rotten Tomatoes TV, "Breaking Bad - You Cut Me Out Scene (S2E6) | Rotten Tomatoes TV," *YouTube*. Jun. 03, 2021. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=kQehbqmkPNg

[12] Rotten Tomatoes TV, "Breaking Bad - Where is the Money? Scene (S4E11) | Rotten Tomatoes TV," *YouTube*. Jun. 19, 2021. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=UyvFiG1K0W0

[13] Tambet, "'Not quite my tempo' - Whiplash (2014) scene," *YouTube*. Apr. 06, 2015. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=xDAsABdkWSc

[14] The Office, "Saddest Moment - The Office US," *YouTube*. Oct. 05, 2019. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=S14aCP3e3yI

[15] The Office, "Jim's Cell Phone Prank on Andy - The Office," *YouTube*. Jan. 11, 2020. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=aOmEhX1LzZ0

[16] The Office, "Jim Carrey Interviews For Regional Manager - The Office US," *YouTube*. Mar. 18, 2019. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=IKW9esYJW0s

[17] Michael Reeves, "I Built A Surgery Robot," *YouTube*. Apr. 28, 2020. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=A_BlNA7bBxo&t=650s

[18] William Osman, "Built My Own X-Ray After a Hospital Charged Me $69,210.32," *YouTube*. Aug. 20, 2021. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=IiJAq53knwc

[19] Key & Peele, "Mr. Garvey Is Your Substitute Teacher - Key & Peele," *YouTube*. Aug. 30, 2021. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=OQaLic5SE_I

[20] Night Watch, "Troy - Achilles Vs Giant Full Fight | Night Watch [1080p HD Blu-Ray]," *YouTube*. Mar. 01, 2023. Accessed: Apr. 24, 2023. [Online]. Available: https://www.youtube.com/watch?v=RE-Dds-yH0M

[21] ove20ro, "Gladiator the final battle HD," *YouTube*. Feb. 13, 2011. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=zmEU7WVXo8c

[22] Movieclips, "Pixie (2020) - Killer Step-Brother Scene | Movieclips," *YouTube*. Mar. 20, 2023. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=yNl-WSRnUek

[23] All Out Action, "'I Can't Touch A Gun' Scene | Hacksaw Ridge," *YouTube*. Oct. 10, 2022. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=zjIW3Vz1ufc

[24] The Office, "Scranton Branch is Closing - The Office US," *YouTube*. Mar. 11, 2023. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=2yxtUxEDG9s

[25] The Office, "The Best of Angela - The Office US," *YouTube*. Mar. 03, 2018. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=w8TeV93Ji7M

[26] The Office, "Dwight Betrays Michael - The Office," *YouTube*. Jan. 04, 2020. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=c_NeL9Ng2Mg

[27] The Office, "Michael Is Dating Pam's Mom - The Office US," *YouTube*. Mar. 23, 2020. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=ugm3hm6nxkI

[28] The Office, "Jim's Radio Prank on Dwight - The Office," *YouTube*. Dec. 10, 2019. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=lWY1p7kKgfI

[29] Helen Villacarlos, "Shocked face meme," *YouTube*. May 15, 2018. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=N8cUy163_rg

[30] Vanity Fair, "Lily Collins Touches a Hedgehog, Cockroaches & Other Weird Stuff | Vanity Fair," *YouTube*. Aug. 03, 2017. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=BC6Ra5giYaI

[31] Vanity Fair, "Liza Koshy Touches a Bearded Dragon, Chinchilla & Other Weird Stuff in the Fear Box | Vanity Fair," *YouTube*. Nov. 21, 2017. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=80GNn_eIOhI&list=RDQMsbxmkmeXVvY&start_radio=1

[32] Saul, "Every Time Mike Outsmarted His Enemies | Breaking Bad & Better Call Saul," *YouTube*. Sep. 27, 2022. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=XQQI72wQjEA

[33] Yellowstone, "Best of The Duttons vs. Trespassers | Yellowstone," *YouTube*. May 24, 2022. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=0dMxuqciYNk

[34] The Office, "The Office but it Gets Progressively More Chaotic - The Office US," *YouTube*. Mar. 23, 2023. Accessed: Apr. 24, 2023. [Online]. Available: https://www.youtube.com/watch?v=EHLGvu0P5bk

[35] The Office, "Best of Michael Scott - The Office US," *YouTube*. Feb. 22, 2019. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=IBJJrZ5LAVQ

[36] Parks and Recreation, "Educating Andy Dwyer | Parks and Recreation," *YouTube*. Jan. 01, 2020. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=O_TsgP4ls5g

[37] Comedy Bites, "Ron Swanson HATES... | Parks and Recreation | Comedy Bites," *YouTube*. Nov. 20, 2019. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=4-4ejRFhqt0

[38] Yellowstone, "The Evolution Of Walker | Yellowstone," *YouTube*. Jul. 04, 2022. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=PVHf0jbLF-k

[39] Prisoner, "Smile (2022) All The Best Scenes.," *YouTube*. Nov. 16, 2022. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=UI-JKD-PxWQ

[40] Marvel Media Posts, "Marvel's Super Sad Scenes (Tear Jerker)," *YouTube*. Jul. 01, 2021. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=0aK-5M2Ot3M

[41] Peacock, "Yellowstone | Most Watched Yellowstone Moments (Part 1)," *YouTube*. Oct. 27, 2022. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=gAkB9W3dgkw

[42] ALTER, "Horror Short Film 'Other Side of the Box' | ALTER," *YouTube*. Dec. 25, 2020. Accessed: Mar. 18, 2023. [YouTube Video]. Available: https://www.youtube.com/watch?v=OrOYvVf6tIM

[43] Good Mythical Morning, "4 Grossest Foods Taste Test," *YouTube*. Mar. 30, 2018. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=wGSHDpX2_sA

[44] Out of Context Brett Cooper, "Brett is grossed-out!," *YouTube*. Aug. 20, 2022. Accessed: Mar. 17, 2023. [Online]. Available: https://www.youtube.com/watch?v=jVQw03pleQA

[45] Paul Lester, "community crack," *YouTube*. Jul. 03, 2021. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=n75qoa2RR8Y

[46] iDecimate, "Homelander Being an Evil Douche For 14 Minutes Straight," *YouTube*. Jul. 11, 2022. Accessed: Mar. 18, 2023. [Online]. Available: https://www.youtube.com/watch?v=Emdjw2Fo3v4.