

**Concordia University**

**Department of Computer Science & Software Engineering**

**COMP 472**

**Artificial Intelligence (Winter 2024)**

**Education MindA.I.lytics**

**Report: Part 2**

Submitted To

[**Dr. René Witte**](https://moodle.concordia.ca/moodle/message/index.php?id=41074754)

**Team: AK\_16**

<https://github.com/RevoSam/COMP472>

We certify that this submission is the original work of members of the group and meets the Faculty’s Expectations of Originality

|  |  |  |
| --- | --- | --- |
| **Sami Najim** | **Atul Gupta** | **Fida Inthesar** |
| Sami Najim 21289640 Data Specialist | Atul Gupta  40188227  Training Specialist | Fida Inthesar  40132564  Evaluation Specialist |

Dated: April 02, 2024

**Contents**

[Emotions 2](#_Toc163037055)

[Datasets 2](#_Toc163037056)

[Data Cleaning 4](#_Toc163037057)

[Labeling 6](#_Toc163037058)

[Dataset Visualization 7](#_Toc163037059)

[A. Class Distribution 8](#_Toc163037060)

[B. Sample Images 9](#_Toc163037061)

[C. Pixel Intensity Distribution 11](#_Toc163037062)

[CNN Architecture 14](#_Toc163037063)

[Main Model [Model 1 with 3 Convolutional Layers] 14](#_Toc163037064)

[All Models’ Architectures 15](#_Toc163037065)

[Main Model 15](#_Toc163037066)

[Variant 1 15](#_Toc163037067)

[Variant 2 16](#_Toc163037068)

[Model Training 17](#_Toc163037069)

[Evaluation 18](#_Toc163037070)

[Performance Metrics 18](#_Toc163037071)

[Main Model metrics: 19](#_Toc163037072)

[Variant 1 metrics: 19](#_Toc163037073)

[Variant 2 metrics: 20](#_Toc163037074)

[Confusion Matrix Analysis 21](#_Toc163037075)

[Kernel Variations 23](#_Toc163037076)

[Main Model: Kernel Size 3x3 23](#_Toc163037077)

[Main Model: Kernel Size 5x5 25](#_Toc163037078)

[Main Model: Kernel Size 7x7 26](#_Toc163037079)

[Conclusion 27](#_Toc163037080)

[Appendix A 28](#_Toc163037081)

[References 30](#_Toc163037082)

# Emotions

Our team decided to analyze the emotions: *Neutral, Focused, Surprised, Happy*

# Datasets

Our team believed that a great dataset for facial emotion recognition should balance diversity in expressions, backgrounds, and participants. Controlled environments, frontal angles, and diverse demographics were essential in dataset selection. Each dataset presented unique challenges and strengths, contributing to our model's robustness and effectiveness. Careful consideration of factors such as bias and ethical considerations was paramount to ensure the model's applicability across varied scenarios and demographics. A summary of all datasets can be found in Table 1.

Our first dataset is the AMFD dataset [1], containing approximately 200 images evenly distributed across the emotion groups. Each participant's contribution of both neutral and happy expressions, along with consistent frontal poses against a neutral white background, helps to eliminate external influences. This dataset was reviewed minimally since its goal of capturing facial emotions aligns with our objectives.

The CK dataset [2,3] consists of 500 images portraying neutral, happy, surprised, and focused expressions. All images are in black and white, offering a consistent aesthetic yet requiring careful review to remove slight redundancies. The distinction between neutral and focused expressions presents a significant challenge, necessitating meticulous curation during model training. However, CK's diverse set of expressions captured in a controlled environment makes it an invaluable resource for training models to recognize emotions effectively.

The FFHQ dataset [4], characterized by emotions like happiness, neutrality, and surprise, adds complexity with its diverse backgrounds. The abundance of neutral and surprised expressions, compared to happy ones, complements what we lacked in other datasets. The inclusion of non-frontal poses and an imbalanced distribution of emotions may challenge the model's adaptability to real-world scenarios, highlighting the need for adaptability to different poses and backgrounds to enhance the model's robustness.

The OSF Indian Database [5], featuring 180 participants, showcases happy, focused, and surprising emotions against a white background, adding diversity to the dataset. The controlled environment simplifies emotion verification, and the clear portrayal of expressions strengthens the dataset. However, its specific demographic focus necessitates broader population considerations to prevent model bias toward specific ethnic characteristics.

The TFEID Asian database [6], characterized by close-up-colored images with minimal background, offers a concise and focused dataset for facial emotion recognition. The minimal background interference enhances emotion clarity, creating an ideal training set for our project. However, like the Indian database, we considered potential biases due to its regional specificity to ensure the model’s generalizability across diverse demographics.

The Real and Fake Face Detection dataset [7], encompassing all emotion groups, presents significant challenges in filtering from a larger database with cropped pictures of individuals' faces. Due to the uncontrolled environment and factors like odd angles, many images were removed from the dataset. Despite these challenges, its real-world scenarios highlight its relevance, but the curation process is crucial for ensuring model effectiveness and avoiding bias during data selection.

Lastly, the TDP dataset [8], featuring around fifteen participants expressing happiness, surprise, and neutrality against varying backgrounds, offers a small-scale yet diverse dataset. It provides a glimpse into real-world scenarios without compromising our training set goals, despite its smaller size.

|  |  |  |
| --- | --- | --- |
| **Dataset** | **Sample Image** | **Licensing** |
| AMFD |  | Creative Commons Attribution 4.0 International Public License |
| CK |  | Custom  non commerical |
| FFHQ |  | Creative Commons BY 2.0, Creative Commons BY-NC 2.0,  Public Domain Mark 1.0, Public Domain CC0 1.0, or U.S. Government |
| OSF |  | Apache License 2.0 |
| Real and Fake Face Detection |  | [CC BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/) |
| TFEID |  | Others  (Not provided) |
| Training Data Profacial  [TDP] |  | Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) |

Table 1: Overview of different datasets used.

# Data Cleaning

Data cleaning involved multiple steps to ensure standardization in quality and size. We manually inspected each image to ensure its quality, eliminating pictures that were corrupted or noisy. By prioritizing image quality from the outset, we negated the need for adjustments in brightness, contrast, or angle. To exclude children, we removed all images of individuals suspected to be under 18 years of age.

Our quality standards required extra time to find datasets meeting our criteria, as many images were unfit for our project. Finding sufficient images for each emotion category proved challenging, with some emotions more difficult to source online. To avoid unwanted biases from an imbalance of images, we set aside excess pictures as potential backups, striving to maintain a similar number of images in each category. Initially planning to use only black and white images, we ultimately chose not to alter colors due to the lack of a color criterion in the assignment, preserving the option to grayscale pictures later using the PIL library if necessary.

With regards to the size of the images, we excluded images that were cropped too closely to the individual's face, showed the whole body, were not rectangular in shape, or the image contained too few pixels that enlargements would distort the facial expressions. To ensure consistency, we ran a python script to gather information pertaining to the size and distribution of the images (see table 2).

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Width | Height | Count | | 600 | 600 | 519 | | 1024 | 1024 | 508 | | 640 | 490 | 331 | | 1080 | 1445 | 300 | | 2444 | 1718 | 215 | | 480 | 600 | 114 | | 640 | 480 | 21 | | 2448 | 3264 | 15 | | 720 | 480 | 10 | | 1944 | 2592 | 9 | | 1280 | 720 | 6 | | 1448 | 3216 | 3 | | 3880 | 5184 | 3 | | 3456 | 4608 | 3 | | 2592 | 4608 | 3 | | 2556 | 3408 | 3 | | |  |  |  | | --- | --- | --- | | Width | Height | Count | | 2464 | 3280 | 3 | | 2304 | 1120 | 3 | | 1472 | 3264 | 3 | | 3888 | 3888 | 3 | | 1072 | 1009 | 1 | | 497 | 600 | 1 | | 954 | 1093 | 1 | | 570 | 588 | 1 | | 943 | 529 | 1 | | 904 | 736 | 1 | | 790 | 849 | 1 | | 784 | 849 | 1 | | 783 | 808 | 1 | | 739 | 769 | 1 | | 696 | 786 | 1 | |
| Table 2: Distribution of number of images per image size | |

Based on preliminary examination of the width and height of our images, we noticed that the range of sizes of the images proved to be quite varied. We would need to rescale our images to achieve uniformity; however, we wanted to minimize potential distortion of the images. Given that scaling down images does not affect the quality of the images, we agreed that 640 x 640 is the preferred size as the bulk of our images fell in that range.

The largest of the images, which represent a small percentage of the overall datasets, were pre-processed either manually in photoshop, in an additional script for resizing, or a combination of both techniques. Once those images were resized, we regrouped them with remaining images which were then resized to 640 x 640 via the script we wrote using PIL library. Table 2 shows these three stages in which the large images go through: Original, Preprocessed, and Final.

|  |  |  |
| --- | --- | --- |
| Original | Preprocessed | Final |
|  |  |  |
| 1080 x 1445 pixels | 1080 x 1080 pixels | 640 x 640 pixels |

Table 3: Various stages of processing large images

# Labeling

After determining which datasets met our criteria, we established definitions for happy, surprised, neutral, and focused expressions. Five of the seven datasets utilized in the project were pre labeller. Even though most datasets came pre-labeled, they lacked sufficient images for our project, necessitating manual labeling for the Real and Fake Face Detection Dataset and the FFHQ Dataset for a significant portion of our work.

Identifying happy expressions was the simplest, characterized by upward turning lips, sometimes showing teeth, and often accompanied by crinkling at the eyes' outer corners and slightly raised eyebrows, creating an open facial expression. Surprised expressions were more challenging to find, marked by widened eyes, eyebrows raised towards the hairline, and horizontal forehead wrinkles, with subjects typically having their mouths open to varying extents. These expressions were grouped together if they shared any of these characteristics. Neutral expressions were more readily available in pre-labeled datasets, defined by a relaxed facial demeanor with eyes focused directly on the camera or a distant object. Distinguishing focused expressions proved most difficult; we sought signs of engagement different from neutral expressions, such as tension around the eyes, a slight frown or squint, a forward-leaning posture, or intense eye gaze. Some neutral images were reclassified as focused upon closer examination.

For manual labeling of the Real and Fake Face Detection Dataset and the FFHQ Dataset, we utilized Labelbox[9], creating individual accounts to upload images. We developed an ontology within Labelbox to categorize images into our four selected emotions, opting for radio buttons over dropdown lists for quick labeling via hotkeys. This process, supported by Labelbox's efficient UI and shortcuts, allowed us to label images in approximately four seconds each. Labelbox's JSON file export feature facilitated organizing images into correctly labeled folders through a script we developed with guidance from Labelbox's documentation[10]. The main limitation encountered was the usage cap on free Labelbox accounts, preventing its use for the extensive FFHQ Dataset.

For the FFHQ Dataset, needing only focused or surprised images, we divided the dataset into thirds to optimize efficiency, starting at different points. This strategy enabled rapid collection of focused images, while surprised expressions took longer to identify, sometimes yielding up to 40 images in a single folder. We manually reviewed the dataset, removing unfit images.

Upon processing and categorizing each dataset, we uploaded the data to GitHub, keeping datasets in separate folders with subfolders for clear identification and citation of each image.

# Dataset Visualization

# 

The visualization of our dataset played a crucial role in understanding its structure, diversity, and potential biases. Utilizing Python’s Matplotlib library, we conducted a comprehensive analysis to visualize class distributions, sample images, and pixel intensity distributions. These visualizations provided us with valuable insights, enabling informed decisions for further data preprocessing and augmentation strategies.

Class distribution analysis highlighted the need for balancing, sample images showcased the dataset’s diversity and flagged potential issues, and pixel intensity distributions shed light on variations in image quality and lighting conditions. Together, these visualizations guided our approach to creating a robust and well-prepared dataset for training our emotion recognition model, ensuring that it is not only diverse but also representative and balanced across different classes.

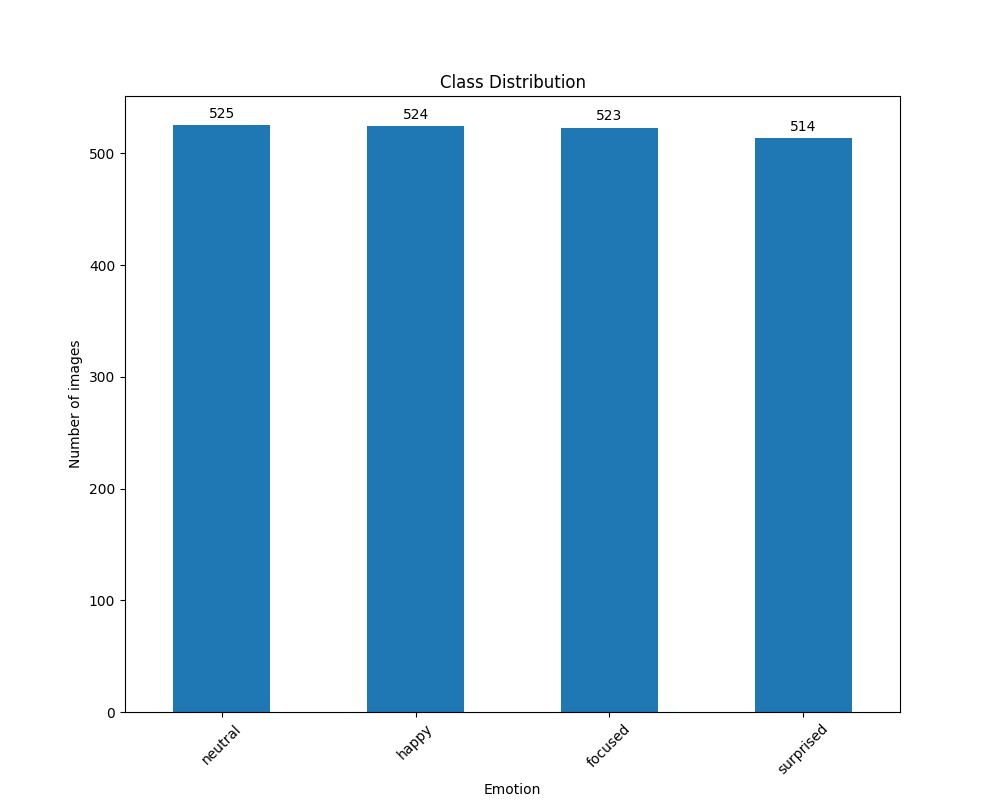
## Class Distribution

To assess the balance of our dataset, we plotted the number of images in each emotional category (happy, surprised, neutral, and focused) using a bar graph **(figure 1)**. This visualization highlighted disparities in class representation, revealing that some emotions were overrepresented while others were underrepresented. For instance, 'happy' images were more abundant, reflecting easier availability or identification, whereas 'focused' expressions were less common, indicating a potential need for targeted data augmentation or collection to balance the dataset.

Based on the analysis we augmented the class representation to maintain uniformity across all classes by removing abundant images of overrepresented classes. The exact numbers on each bar provided a clear, quantifiable measure of the class distribution, which was instrumental in planning our next steps for dataset augmentation and ensuring that our model training would not be biased towards more frequently represented classes.

Figure 1: Class distribution bar graph

|  |  |
| --- | --- |
| **Initial Distribution**  neutral 1078  happy 588  focused 544  surprised 500 | **Final Distribution**  neutral 525  happy 524  focused 523  surprised 514 |



## Sample Images

Understanding the content and quality of our dataset was furthered by presenting collections of 25 randomly chosen images for each class, arranged in a 5x5 grid **(Figures 2 - 5)**. This exercise was not only instrumental in visually assessing the diversity and consistency of image representation within each category but also highlighted any anomalies or mislabels that might have escaped earlier scrutiny.

Ensuring that these images were randomly selected upon each execution of our code allowed for a dynamic assessment of our dataset, preventing oversight of potential issues confined to a subset of the data. By resizing each sample image to a calculated ratio using our defined heuristic, we maintained consistency in visual presentation, which was essential for accurately gauging the dataset’s characteristics.

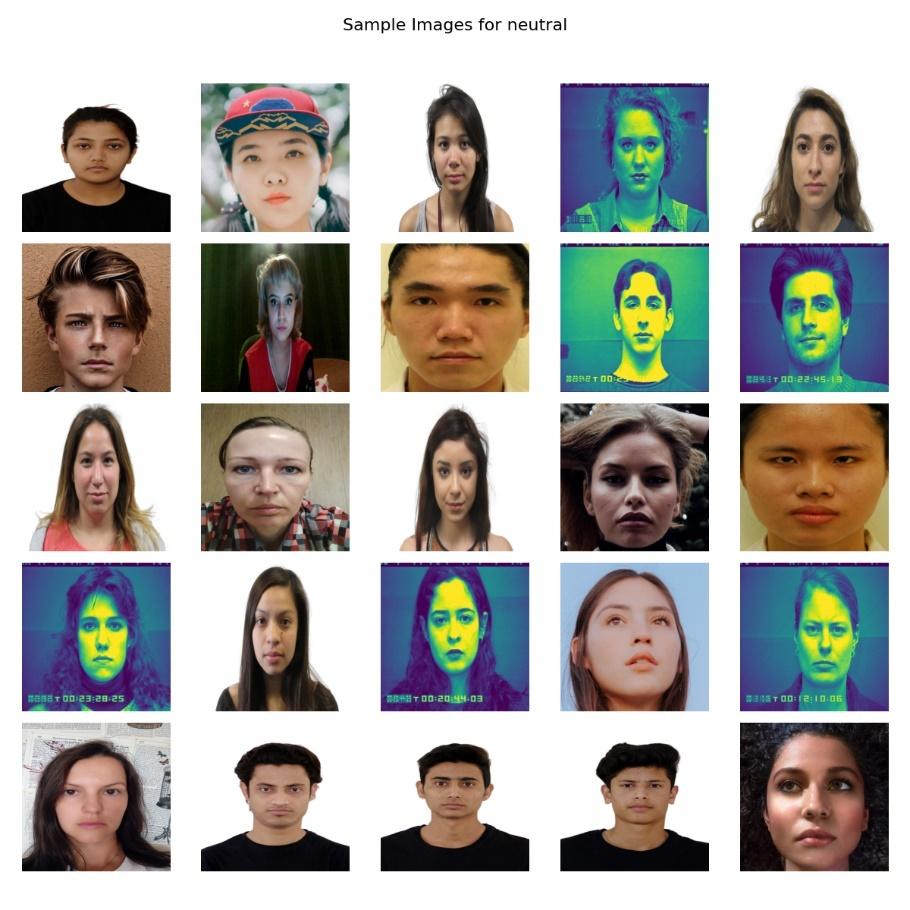


Figure 2: Sample Image Grid of `Neutral` expression

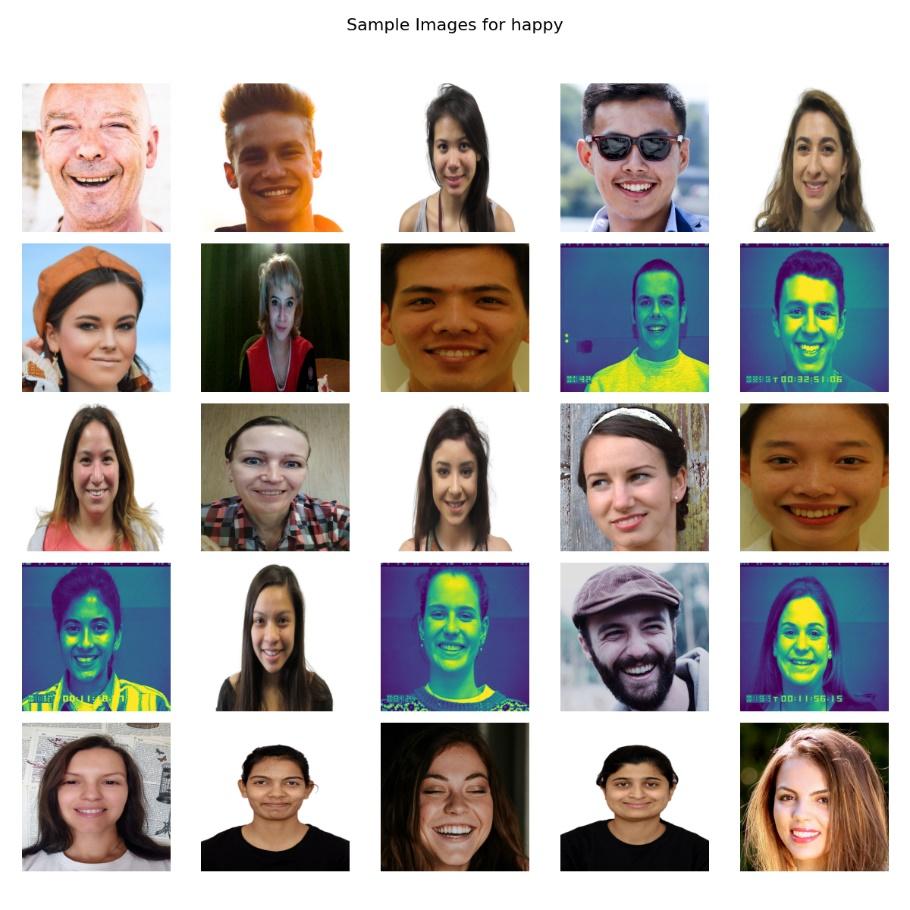


Figure 3: Sample Image Grid of `Happy` expression

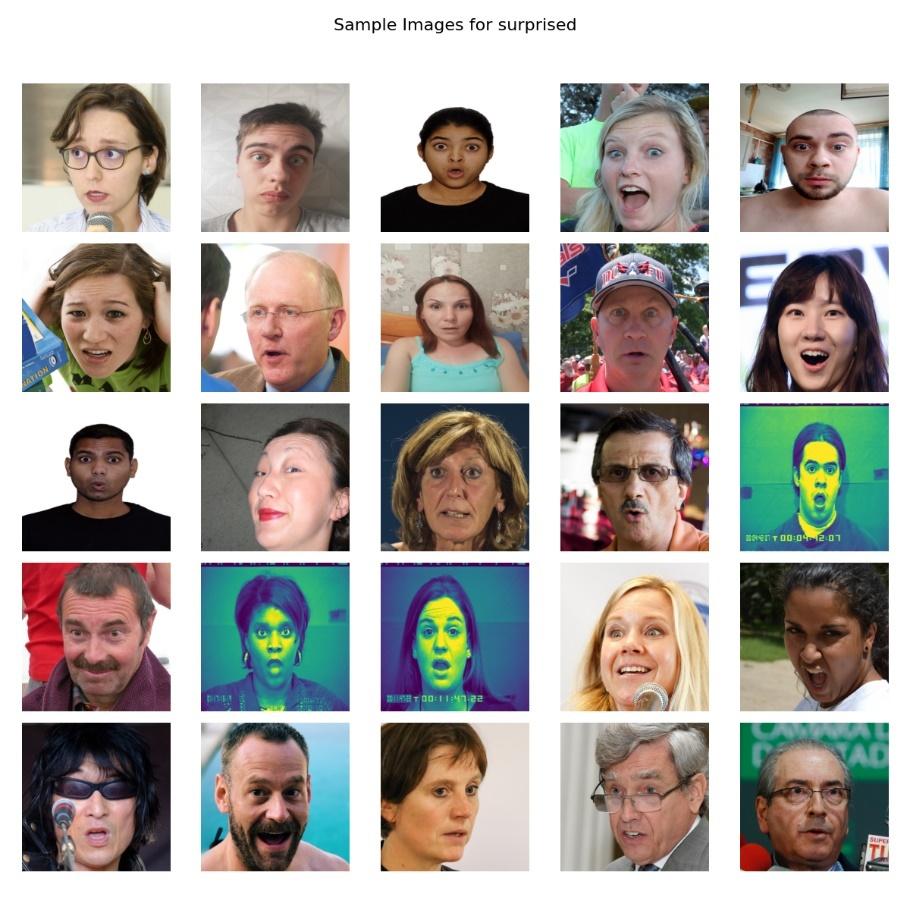


Figure 4: Sample Image Grid of `Surprised` expression

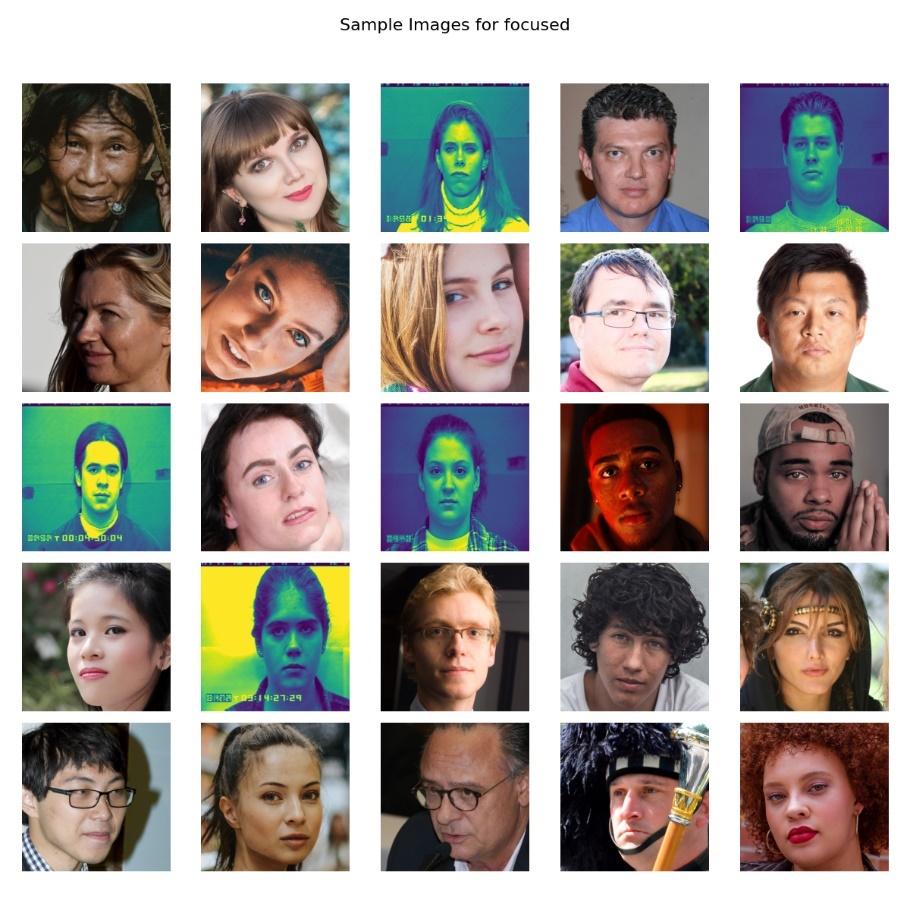


Figure 5: Sample Image Grid of `Focused` expression

## Pixel Intensity Distribution

The analysis of pixel intensity distributions for a randomly chosen subset of 25 images from each class provided insights into the dataset's lighting conditions and overall image quality **(Figures 6-9)**. We plotted histograms for each image, overlaying the intensity distributions for the **Red, Geen, and Blue** channels in color images, which was instrumental in identifying variations in lighting conditions across our dataset.

This visualization was crucial for understanding how different lighting conditions might affect model training and performance. For instance, images with predominantly low-intensity values might be underexposed, potentially necessitating brightness normalization during preprocessing. Conversely, images skewed towards high-intensity values could be overexposed, similarly affecting model perception.

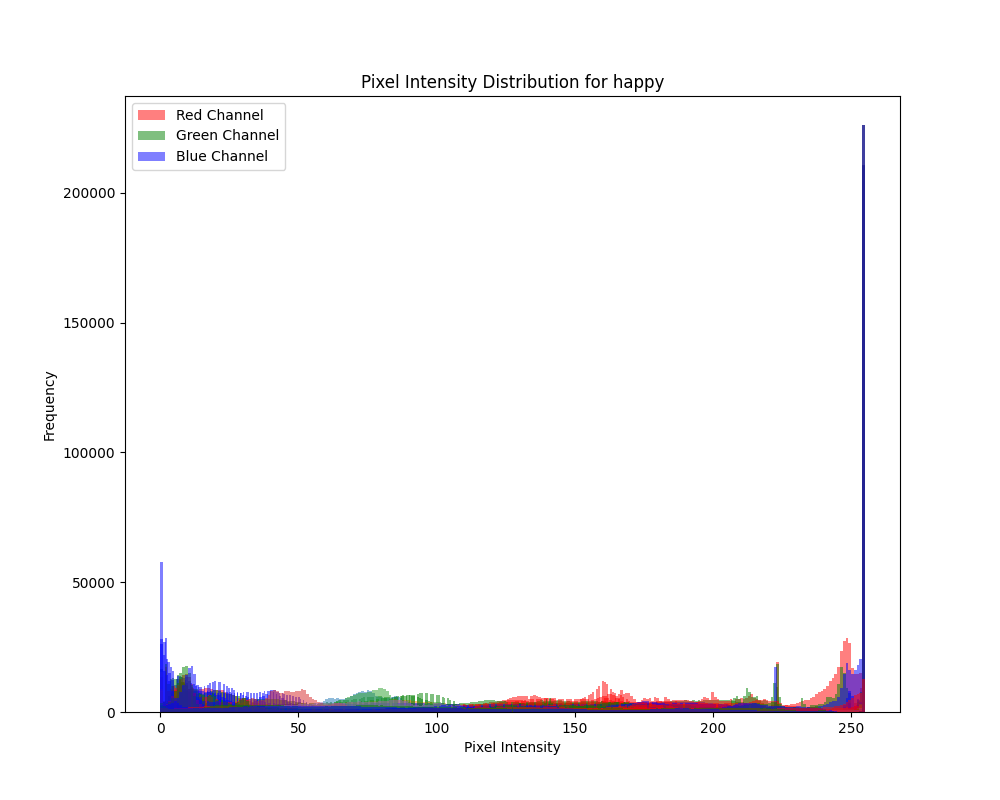


Figure 6: Pixel Intensity Distribution - Happy

A graph of a number of data

Description automatically generated with medium confidence

Figure 7: Pixel Intensity Distribution - Neural

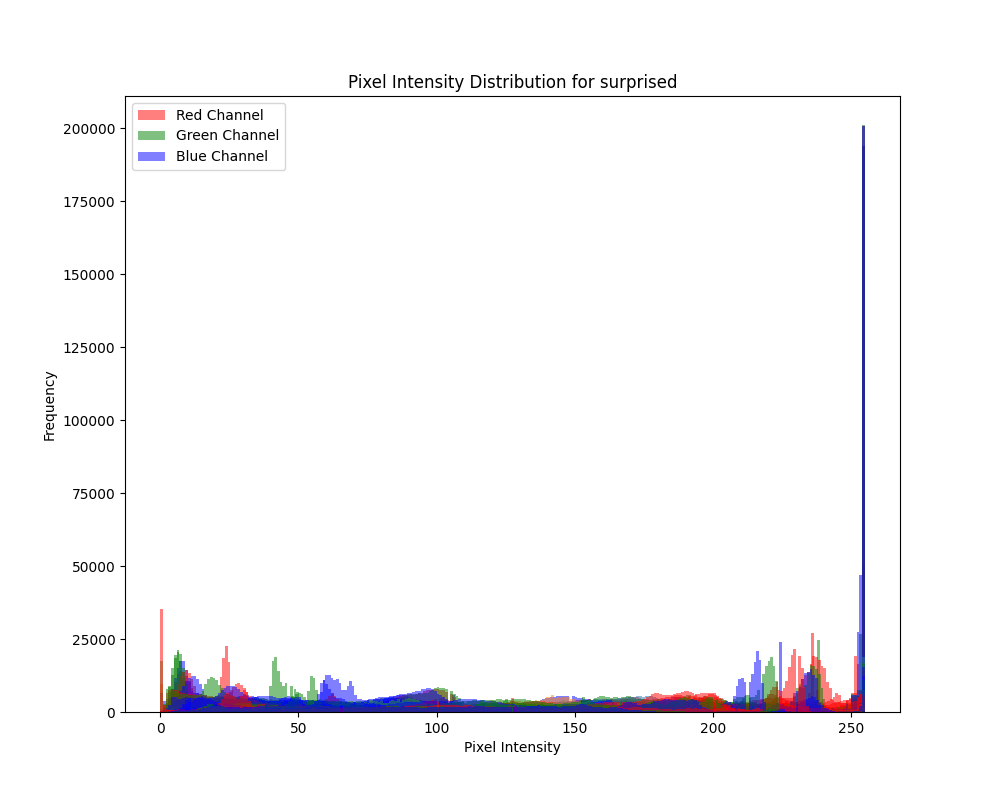


Figure 8: Pixel Intensity Distribution - Surprised

A graph of data with different colored lines

Description automatically generated with medium confidence

Figure 10: Pixel Intensity Distribution - Focused

# CNN Architecture

**Overview**  
This report extends from the initial dataset collection and analysis phase, detailing the development and evaluation of a Convolutional Neural Network (CNN) for recognizing four facial expressions: Neutral, Focused, Surprised, and Happy. Implemented in PyTorch, our approach examines CNN's architecture, training methodology, and performance evaluation, leveraging the datasets curated in the project's first phase.

**Architecture**

Our primary CNN model, referred to as ConvNet, incorporates deep learning principles tailored for the nuanced task of facial expression recognition. The architecture is designed to process grayscale images, systematically extracting and learning features through convolutional layers before classifying into one of the four target emotions.

# Main Model [Model 1 with 3 Convolutional Layers]

The architecture of our primary model is a testament to the power of simplicity meshed with precision. Designed to process grayscale images, it unfolds as follows:

* **Input Layer:** Single-channel input catering to grayscale images, aligning with our dataset's format.
* **Convolutional Layers:**
  + First Layer: It starts with a convolutional layer with 32 output channels and a 3x3 kernel, followed by batch normalization and ReLU activation. This layer is pivotal in initial feature extraction.
  + Subsequent Layers: The architecture extends with two more convolutional layers, each doubling the output channels from the preceding one, maintaining a 3x3 kernel size. These layers are interspersed with batch normalization, ReLU activation, and max pooling operations, culminating in a feature-rich and compact representation of the input images.
* **Flattening Layer:** A bridge between the convolutional layers and the fully connected layers, transforming the 2D feature maps into a 1D feature vector.
* **Fully Connected (Linear) Layers:** The network concludes with linear layers that distill the learned features into predictions across our four target classes. A dropout layer with a 20% dropout rate is incorporated before the final linear layer to mitigate overfitting.
* **Output Layer:** Outputs the logits corresponding to the four emotions under consideration.

This model strikes a balance between depth for capturing complex features and computational efficiency, making it adept for real-time applications.

# All Models’ Architectures

Three variants were explored to assess the impact of network depth and type

### **Main Model**

Main model of the ConvNet is characterized by its streamlined architecture, designed for efficient learning while maintaining the capacity to capture essential features for emotion recognition. This variant's construction includes:

* **Three Convolutional Layers**: These layers increase in depth from 32 to 128 output channels, with each convolution followed by batch normalization, a ReLU activation function, and max pooling. This setup ensures feature extraction at varying granularities, with pooling layers reducing dimensionality and enhancing the network's focus on relevant features.
* **Sequential Linear Layers**: After flattening the output from the convolutional layers, two linear transformations are applied. The first maps the flattened features to an intermediate dimension (equal to four times the number of output channels from the last convolutional layer), and the second maps this intermediate representation to the four target classes. A dropout layer with a 0.2 rate is incorporated to mitigate overfitting by randomly omitting features during training.

### **Variant 1**

Variant 1 represents a more complex and deeper configuration of the ConvNet, designed to explore the impact of additional convolutional layers and increased depth on the network's ability to learn and generalize. The architecture specifics include:

* **Six Convolutional Layers**: The initial layers mimic the Main Model in structure but then extend to include additional convolutional stages, ultimately expanding to 256 output channels. This expansion allows the network to learn more complex and abstract features from the facial images.
* **Enhanced Linear Section**: Post-convolution, the linear section comprises three layers, significantly increasing the dimensionality in the intermediate stage to 32 times the output channels of the final convolutional layer before reducing down to the four emotion categories. This broader intermediary space provides a richer set of features for the final classification task.

### **Variant 2**

Variant 2 introduces an innovative approach by incorporating depthwise separable convolutions into the architecture, aiming to combine the benefits of computational efficiency with the network's capability to discern intricate patterns for emotion recognition. The defining features of this variant include:

* **Depthwise Separable Convolutional Layers:** This variant adopts depthwise separable convolutions, which are a two-step process involving depthwise convolutions that apply a single filter to each input channel, and pointwise convolutions that combine the depthwise convolution outputs. This methodology significantly reduces the number of parameters compared to traditional convolutional layers, aiming for a balance between computational efficiency and the ability to capture essential features.
* The first layer is a standard convolutional layer setting the stage for feature extraction. Following this, the architecture transitions into depthwise separable convolutions, progressively increasing the complexity and depth of the feature maps while managing computational costs.
* This variant retains the structure of increasing depth, similar to the Main Model, but utilizes depthwise separable convolutions to achieve this with fewer parameters. The output channels double after each separable convolution, maintaining the hierarchical feature extraction capability while being more resource-efficient.
* **Adapted Sequential Linear Layers**: Following the convolutional stages, Variant 2 employs a unique set of linear transformations designed to refine and utilize the efficiently processed features for classification.
  + After the feature maps are flattened, they pass through an expanded linear section that initially increases the feature dimensionality more significantly than in the Main Model or Variant 1, aiming to explore a broader feature space. This is then condensed through successive layers to achieve the final emotion classification.
  + A dropout layer is strategically placed to prevent overfitting, ensuring that the network remains robust to variations within the input data.
* **Enhanced Feature Integration**: By incorporating an additional dense layer in the linear section, Variant 2 aims to integrate the extracted features more comprehensively before the final classification. This additional layer acts as a mechanism to combine and refine the features, potentially uncovering complex relationships that aid in distinguishing between the target emotions.

# Model Training

The training process was carefully designed to ensure model robustness and reliability. Key aspects include:

**Epochs**: Models were trained for a maximum of 30 epochs, with safeguards against overfitting through early stopping mechanisms based on validation loss performance.

**Learning Rate**: Initialized at 0.001, with adaptive adjustments using a scheduler to reduce the rate by 1% every 10 epochs, facilitating fine-grained learning as training progresses.

**Optimization Algorithm**: Adam optimizer was chosen for its effectiveness in handling sparse gradients and adapting learning rates, essential for converging to optimal solutions efficiently.

**Loss Function:** Cross-entropy loss, suitable for multi-class classification tasks, was used to compute the difference between predicted probabilities and actual class labels.

**Data Augmentation and Normalization:** Implemented in the dataset preparation phase to enhance model generalization by presenting varied representations of training images.

**Validation and Early Stopping:** Regular validation checks were conducted to monitor model performance on unseen data, employing early stopping to halt training when validation loss ceased to decrease, indicating potential overfitting.

# Evaluation

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Macro | | | Micro | | | Accuracy |
| P | R | F | P | R | F |
| Main model | 0.76 | 0.76 | 0.76 | 0.37 | 0.35 | 0.33 | 0.76 |
| Variant 1 | 0.5 | 0.45 | 0.42 | 0.32 | 0.34 | 0.32 | 0.45 |
| Variant 2 | 0.76 | 0.73 | 0.72 | 0.43 | 0.32 | 0.31 | 0.73 |

Post-training, models were evaluated on a held-out test set to gauge their performance comprehensively. Evaluation metrics included accuracy, precision, recall, and F1-score, alongside confusion matrices to visualize model strengths and weaknesses across the four emotional categories.

Final testing evaluation metrics using standalone testing script:

## Performance Metrics

The detailed comparison of model variants illuminated the nuanced impacts of network depth and convolutional layer configurations. Specifically, variations in kernel sizes elucidated the balance between computational efficiency and the ability to capture detailed versus broader facial features, with larger kernels favoring the latter at the expense of increased model complexity.

**Comparison between 3 Models**

Overall, it was observed that main model had the best performance compared to the other variants. This model achieved an accuracy of 0.76. It shows relatively balanced precision, recall, and F1-scores across all emotion classes, with F1-scores ranging from 0.67 to 0.88. The micro-average F1-score is 0.33, indicating a fair overall performance across all classes.

The difference between the main model and variant 1 is considerably large. However, the main model outperforms variant 2. This indicates that the main model’s performance is balanced across all the metrics. However, this model still struggles to differentiate with focused and neutral expressions. This indicates that the model might not be effectively capturing the nuances of these emotions, leading to lower precision and recall scores for these cases. Subjective true labels of the dataset is also another factor resulting in deviations from actual true emotion classification.

### **Main Model metrics:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Emotion** | **Precision** | | **Recall** | **F1-Score** | **Support** |
| **focused** | 0.70 | 0.71 | | 0.70 | 78 |
| **happy** | 0.82 | 0.76 | | 0.79 | 78 |
| **neutral** | 0.67 | 0.67 | | 0.67 | 78 |
| **surprised** | 0.86 | 0.91 | | 0.88 | 78 |
|  |  |  | |  |  |
| **Accuracy** |  |  | | 0.76 | 312 |
| **Macro Avg** | 0.76 | 0.76 | | 0.76 | 312 |
| **Weighted Avg** | 0.76 | 0.76 | | 0.76 | 312 |
| **Micro Avg** | 0.37 | 0.35 | | 0.33 |  |

In contrast to the first model, variant 1 displays lower overall performance, with an accuracy of only 45%. The precision, recall, and F1-scores for all emotion classes are notably lower compared to the first model, with F1-scores ranging from 0.27 to 0.53, notably lower across all emotion categories. The lower micro-average F1-score of 0.32, indicating a worse overall performance compared to the first model, the lower accuracy suggests significant misclassification or imbalance issues.

### **Variant 1 metrics:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Emotion | Precision | Recall | F1-Score | Support |
| focused | 0.49 | 0.46 | 0.47 | 78 |
| happy | 0.54 | 0.18 | 0.27 | 78 |
| neutral | 0.57 | 0.33 | 0.42 | 78 |
| surprised | 0.39 | 0.83 | 0.53 | 78 |
|  |  |  |  |  |
| Accuracy |  |  | 0.45 | 312 |
| Macro Avg | 0.50 | 0.45 | 0.42 | 312 |
| Weighted Avg | 0.50 | 0.45 | 0.42 | 312 |
| Micro Avg | 0.32 | 0.34 | 0.32 |  |

This model achieved an accuracy of 0.73, slightly lower than the main model but significantly higher than the first variant. It shows varied precision, recall, and F1-scores across emotion classes, with F1-scores ranging from 0.59 to 0.84. The micro-average F1-score is 0.31, indicating a moderate overall performance across all classes.

### **Variant 2 metrics:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Emotion | Precision | Recall | F1-Score | Support |
| focused | 0.77 | 0.71 | 0.74 | 78 |
| happy | 0.77 | 0.63 | 0.73 | 78 |
| neutral | 0.60 | 0.59 | 0.59 | 78 |
| surprised | 0.77 | 0.92 | 0.84 | 78 |
|  |  |  |  |  |
| Accuracy |  |  | 0.73 | 312 |
| Macro Avg | 0.73 | 0.73 | 0.73 | 312 |
| Weighted Avg | 0.73 | 0.73 | 0.72 | 312 |
| Micro Avg | 0.43 | 0.32 | 0.31 |  |

In conclusion, the first model appears to be the most balanced and performs relatively well across all emotion classes. The third model also performs reasonably well but shows some variations in performance across different emotion classes. The second model performs significantly worse than the other two, indicating potential issues such as imbalance or misclassification problems.

## Confusion Matrix Analysis

The confusion matrices provide a visual representation of the model's predictions across the emotion classes, depicting the areas of strength and weakness. In the main model’s confusion matrix, we observe a diagonal pattern, indicating that the model's predictions align closely with the ground truth labels for most classes. However, there are instances of misclassification between focused and neutral, suggesting that the model struggles to differentiate between these two emotions.

**A blue squares with white text

Description automatically generated**

Confusion Matrix for the Main Model

In Main model, which has a streamlined architecture with three convolutional layers, the confusion matrix reveals similar patterns of misclassification, particularly between the focused and neutral. This suggests that while the simplified architecture may offer computational efficiency, it may also sacrifice discriminative capability for certain emotions, leading to confusion between similar facial expressions. The misclassification patterns seem to be more evenly distributed across the classes, with no single class dominating the misclassifications. Overall, the number of misclassifications appears to be lower compared to the main model.

A blue squares with white text

Description automatically generated

Confusion Matrix for Variant 1

In Variant 1, which adopts a more complex architecture with six convolutional layers, the confusion matrix shows even worse performance in discriminative capability, with more instances of misclassification between the emotion classes. There are still areas of confusion, particularly between the ‘surprised’ and the rest of the classes, indicating that the increased depth may not fully address the model's challenges in distinguishing between subtle variations in facial expressions. Additionally, there are noticeable misclassifications between the "focused" and "surprised" classes. Overall, the misclassification patterns in variant 1 seem to be more varied and less consistent across classes compared to the other variants.

A blue squares with white text

Description automatically generated

Confusion Matrix for Model 3

Variant 2 appears to have a relatively balanced distribution of misclassifications across classes, but with a number of misclassifications for the "focused" and "neutral" classes. Variant 2 shows a more evenly distributed pattern of misclassifications compared to the main model, with generally lower misclassification rates. Variant 1 demonstrates a higher degree of misclassifications, particularly for the ‘surprised’ class, suggesting potential issues with model performance or training data. Overall, the main model seems to have the most favorable confusion matrix, indicating better performance in terms of minimizing misclassifications across different classes. The confusion matrices highlight the importance of carefully designing CNN architectures to balance computational efficiency with distinguishability, as evidenced by the varying patterns of misclassification across the main model and its variants.

## Kernel Variations

### **Main Model: Kernel Size 3x3**

The kernel variants explore the impact of different sizes on the Main Model’s performances, offering a deeper insight into the trade-offs between spatial granularity and computational cost. In kernel size of 3, we observe moderate performance across the classes, with balanced precision and recall scores indicating consistent discrimination between emotions. The precision, recall, and F1-scores for each emotion class vary between 0.43 and 0.63. Overall accuracy is 0.52, with micro-average precision, recall, and F1-score being 0.52, 0.49, and 0.43 respectively. However, there is always room for improvement, especially in distinguishing between the focused and neutral classes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Emotion | Precision | Recall | F1-Score | Support |
| focused | 0.44 | 0.45 | 0.45 | 2310 |
| happy | 0.59 | 0.57 | 0.58 | 2310 |
| neutral | 0.44 | 0.43 | 0.43 | 2310 |
| surprised | 0.61 | 0.63 | 0.62 | 2310 |
|  |  |  |  |  |
| Accuracy |  |  | 0.52 | 9240 |
| Macro Avg | 0.52 | 0.52 | 0.52 | 9240 |
| Weighted Avg | 0.52 | 0.52 | 0.52 | 9240 |
| Micro Avg | 0.49 | 0.43 | 0.44 |  |

A blue squares with white text

Description automatically generated

Confusion Matrix for Main Model - Kernel Size 3x3

### **Main Model: Kernel Size 5x5**

When deploying a larger kernel size of 5, there is improved performance, with higher precision and recall scores across most emotion classes. Overall accuracy improved slightly to 0.54, with micro-average precision, recall, and F1-score being 0.55, 0.46, and 0.42 respectively. This proves that the larger kernel size enables the model to capture more detailed features, distinguishing between emotions better. There is still some confusion between certain classes, indicating that optimization may be necessary to further enhance the model’s performance.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Emotion | Precision | Recall | F1-Score | Support |
| focused | 0.46 | 0.51 | 0.48 | 2310 |
| happy | 0.64 | 0.50 | 0.56 | 2310 |
| neutral | 0.45 | 0.49 | 0.47 | 2310 |
| surprised | 0.65 | 0.66 | 0.66 | 2310 |
|  |  |  |  |  |
| Accuracy |  |  | 0.54 | 9240 |
| Macro Avg | 0.55 | 0.54 | 0.54 | 9240 |
| Weighted Avg | 0.55 | 0.54 | 0.54 | 9240 |
| Micro Avg | 0.46 | 0.42 | 0.43 |  |

A blue squares with white text

Description automatically generated

Confusion Matrix for Model 1 with Kernel Size 5x5

### **Main Model: Kernel Size 7x7**

With kernel size of 7, there are mixed results. We have slightly diminished performance compared to kernel size of 5. Overall accuracy decreased to 0.50, with micro-average precision, recall, and F1-score being 0.51, 0.45, and 0.44 respectively. While precision and recall scores for certain emotions remain consistent, there is a noticeable decrease in performance for others, suggesting that the larger kernel size may cause an over-smoothing of features leading to a loss of distinguishability. The confusion matrix indicates further shifts in misclassifications compared to the previous variants.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Emotion | Precision | Recall | F1-Score | Support |
| focused | 0.41 | 0.48 | 0.44 | 2310 |
| happy | 0.57 | 0.47 | 0.51 | 2310 |
| neutral | 0.44 | 0.43 | 0.43 | 2310 |
| surprised | 0.61 | 0.63 | 0.62 | 2310 |
|  |  |  |  |  |
| Accuracy |  |  | 0.50 | 9240 |
| Macro Avg | 0.51 | 0.50 | 0.50 | 9240 |
| Weighted Avg | 0.51 | 0.50 | 0.50 | 9240 |
| Micro Avg | 0.45 | 0.44 | 0.43 |  |

A blue squares with white text

Description automatically generated

Confusion Matrix for Main Model - Kernel Size 7x7

In summary, it was observed that kernel size variations also play a crucial role in the performance of convolutional neural networks, particularly in tasks like emotion classification. Experimentation and careful tuning become necessary to find the optimal kernel size for a given task and dataset, as evident from above results.

# Conclusion

This project's exploration into CNN architectures for facial emotion recognition has underscored the delicate interplay between model complexity, feature extraction capabilities, and generalization performance. The findings advocate for further research into network configurations, training strategies, and the integration of more diverse datasets to enhance model robustness and applicability across varied demographic and environmental contexts.

Future refinements may explore advanced regularization techniques, alternative activation functions, and more sophisticated learning rate schedules, alongside expanding the scope of recognizable emotions to encompass a broader spectrum of human expressions.

# Appendix A

**Evaluation predictions for the three models**



**Sample Predictions from Model 1**



**Sample Predictions from Model 2**



**Sample Predictions from Model 3**

# References

[1]

‘MULTIRACIAL FACE DATABASE’, sciplab. Accessed: Mar. 03, 2024. [Online]. Available:<https://jacquelinemchen.wixsite.com/sciplab/face-database>

[2]

Lucey, J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression" Accessed: Mar. 03, 2024. [Online]. Available: [CK+ dataset (kaggle.com)](https://www.kaggle.com/datasets/shuvoalok/ck-dataset)

[3]

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression. Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010), San Francisco, USA, 94-101. Accessed: Mar. 03, 2024. [Online]. Available: [CK+ Dataset | Papers With Code](https://paperswithcode.com/dataset/ck)

[4]

‘NVlabs/ffhq-dataset’. NVIDIA Research Projects, Mar. 02, 2024. Accessed: Mar. 03, 2024. [Online]. Available:<https://github.com/NVlabs/ffhq-dataset>

[5]

S. Tewari, S. Mehta, and N. Srinivasan, ‘IIMI Emotional Face Database’, May 2023, Accessed: Mar. 03, 2024. [Online]. Available:<https://osf.io/f7zbv/>

[6]

‘spenceryee/CS229’. Accessed: Mar. 03, 2024. [Online].

[7]

‘Real and Fake Face Detection’. Accessed: Mar. 03, 2024. [Online]. Available:<https://www.kaggle.com/datasets/ciplab/real-and-fake-face-detection>

Available:<https://github.com/spenceryee/CS229>

[8]

‘Facial Emotion Recognition Dataset’. Accessed: Mar. 03, 2024. [Online]. Available:<https://www.kaggle.com/datasets/tapakah68/facial-emotion-recognition>

[9]

‘Labelbox docs’, Labelbox docs. Accessed: Mar. 03, 2024. [Online].

Available:<https://docs.labelbox.com/>

[10]

‘labelbox-python/examples/basics/data\_rows.ipynb at master · Labelbox/labelb ox-python’, GitHub. Accessed: Mar. 03, 2024. [Online].

Available: [https://github.com/Labelbox/labelbox-python/blob/master/examples/ basics/data\_rows.ipynb](https://github.com/Labelbox/labelbox-python/blob/master/examples/%20basics/data_rows.ipynb)