Emotion Recognition in Videos Through Deep Neural Network Models

Suxi Li
Stanford University
suxi2024@stanford.edu

Yichen Jiang
Stanford University
ycjiang@stanford.edu

Senyang Jiang
Stanford University
senyangj@stanford.edu

Abstract

Emotion recognition plays a crucial role in interpersonal social interactions, making it essential for robots and human-centered systems to accurately identify and respond to human emotions. While high accuracy has been achieved in classifying emotions from static images, recognizing emotions in videos presents a more complex and realistic challenge. This paper explores and implements various deep learning architectures to enhance the accuracy of human emotion detection in videos. We used RAVDESS dataset in our project and experiment with different dataset preprocessing techniques, such as resizing and data augmentation to boost model performance. Our best model achieved an accuracy of 84%. Finally, we present our results and discuss future work necessary for further advancements in this field.

1. Introduction

Human emotion recognition has long been a crucial research area for human-interacting robotics, as an intelligent system can only effectively interact with humans after accurately identifying their emotions. Emotions can be conveyed through both visual (body gestures, facial expressions) and non-visual (speaking tones, word choice) cues. In 1967, Mehrabian demonstrated that 55% of emotional information is communicated visually, while vocal and verbal information account for 38% and 7%, respectively [7]. Our deepest emotions often manifest not through words, but through subtle nuances in our facial expressions, such as the rising corners of the mouth or furrowed brows. Therefore, this research focuses primarily on emotion recognition using features extracted from visual facial expressions in videos.

In this project, we evaluate some popular deep learning solutions to recognize emotions from facial expressions detected in the videos, trained and tested with the RAVDESS dataset[5]. We begin with an early fusion approach as a baseline. This method reshapes the input 4D video frames into a three-dimensional matrix by combining the channel and time sequence dimensions. The resulting matrix is then fed into a basic 2D Convolutional Neural Network (CNN) to predict the emotion label. Subsequently, we implement a late fusion model, which concatenates the output feature maps extracted from each video frame and uses these concatenated features to predict the emotion class. To leverage the transitions between neighboring video frames and extract temporal features, we introduce a more advanced 3D CNN model, which achieves better accuracy by utilizing this temporal information.

Finally, we employ a Recurrent Convolutional Neural Network (RCNN) to more effectively utilize the sequential nature of video frames. Our RCNN model extracts features for each time frame by incorporating the feature map from the previous time step and the information from the same frame at a shallower layer. This approach allows us to include both deep features extracted from each time frame and temporal features from the transitions between frames, providing the model with more comprehensive and informative data.

The specific structure of the dataset and the detailed implementation of the models will be discussed in following sections.

2. Problem Statement

Emotion recognition using facial expressions is a well-established area of research within computer vision. In our project, the input to the emotion recognition model is a video of human speech, which consists of a sequence of frames. We can express the input to the model as a 4D tensor of size $T \times C \times H \times W$, where T is the number of input frames, which can be smaller than the total number of frames in the video via sampling. C is the channel size of each frame (which is usually 3), and H and W are frame height and width respectively. The output of the model is a class label l that categorizes the video into different emotions.

In this project, we will initially apply basic convolutional methods such as early fusion and late fusion, which we have learned in class, to video data for classifying emotions. We will then advance our approach by implementing a hy-

brid model that combines Convolutional Neural Networks (CNNs) for spatial feature extraction with Recurrent Neural Networks (RNNs) for temporal sequence modeling.

We hypothesize that a larger, more complex model will significantly outperform simpler models in the task of emotion classification. Although we will not explore new architectures in this project, we will implement multiple established networks widely utilized in computer vision tasks. We will conduct experiments to provide insightful analysis on the behavior and performance of these networks.

This project will benefit us in two significant ways: firstly, it will enhance our experience with processing video data and employing computer vision techniques. Secondly, it will lay a strong foundation for us to venture into more sophisticated computer vision projects.

3. Related Work

The six principal emotions—happiness, sadness, anger, fear, surprise, and disgust—were initially identified by Ekman and Wllace [2] in the field of emotion recognition. Neutral was included in most human recognition datasets later on. The RAVDESS dataset also includes calm as an additional emotion, resulting in eight basic emotions in this project.

Most modern methods for facial expression recognition (FER) are based on deep learning and have achieved remarkable success in predicting emotions. These approaches generally employ different types of deep neural networks with convolutional layers. Due to their versatility in numerous computer vision tasks, convolutional neural networks (CNNs) have proven highly effective in emotion recognition from video-based datasets. Andrej *et al*. [4] provided an extensive empirical evaluation of CNNs on large-scale video classification using a dataset of 1 million YouTube videos belonging to 487 classes. The authors investigated three approaches - early fusion, late fusion, and slow fusion - for fusing information across the temporal domain. We will apply these three approaches to the RAVDESS dataset in this project and conduct thorough experiments to explore their effectiveness in emotion recognition. Our goal is to determine which fusion method yields the highest accuracy and robustness in predicting the eight basic emotions present in the dataset. Through these experiments, we aim to gain insights into the optimal strategies for leveraging temporal information in video-based emotion recognition tasks.

Hybrid network proved to be a promising approach to improve the accuracy of emotion recognition. Fan *et al*. [3] presents a video-based emotion recognition system that utilizes a hybrid network, consisting of Recurrent Neural Network (RNN) and 3D Convolutional Network (C3D). This paper primarily focuses on extracting temporal information from videos, rather than solely relying on spatial information. Unlike many previous models that extracted static frames from videos and treated them as images, this approach aims to leverage the dynamic aspects of video data to predict emotions more accurately. The paper utilized C3D to model appearance and motion information simultaneously, and further combining it with Long Short-Term Memory (LSTM), which is proved to be efficient in dealing with long sequences of input data. Two-stream networks proposed by Simonyan and Zisserman [11] that separate motion and appearance for action recognition in videos achieve better results. Similarly, Manalu *et al*. [6] developed a hybrid Convolutional Neural Network – Recurrent Neural Network (CNN-RNN) model that is adept at detecting human emotions derived from facial expressions based on video data. Three models – MobileNetV2-RNN, InceptionV3-RNN, and custom CNN-RNN – are developed for the classification. The authors concluded that the developed models demonstrate enhanced efficiency in distinguishing nuanced emotions on Emotional Wearable Dataset 2020. It is worth noting that this dataset also consists of amusement, enthusiasm, awe, and liking that were not often explored in other datasets.

In facial expressions, much of the information is often derived from specific parts of the face, such as the mouth and eyes, while other parts, like the ears and hair, contribute minimally. Therefore, an effective machine learning framework should concentrate on these crucial facial regions and be less sensitive to less informative areas. Because of the close relationship between face alignment and facial expression recognition, Tautkute *et al*. [12] proposed to build upon Deep Alignment Network (DAN), an innovative facial landmark detection model, to exploit the location of these facial landmarks (like eyes, lips, and eyebrows) and to use such information to improve the accuracy of video emotion recognition. Minaee *et al*. [9] proposed a deep learning approach based on attentional convolutional network, which is able to focus on important parts of the face and achieves significant improvement over previous models on multiple datasets, including FER-2013, CK+, FERG, and JAFFE. The authors also use a visualization technique which is able to find important face regions for detecting different emotions, based on the classifier's output.

Recently, there has been a growing interest in multimodal emotion recognition. Pan *et al*. [10] provides a comprehensive review of multimodal emotion recognition from the perspectives of multimodal datasets, data preprocessing, unimodal feature extraction, and multimodal information fusion methods in recent decades. This paper summarizes an emotional recognition system into four pipeline stages: data collection, data preprocessing, emotional feature extraction and emotion recognition. It was also pointed out that by properly fusing different modalities, emotional recognition performance can be enhanced because different modalities complement each other. The

work by Middya *et al*. [8] explores model-level fusion to find out the optimal multimodal model for emotion recognition using audio and video modalities. Separate feature extractor networks for audio and video data are proposed, and an optimal multimodal emotion recognition model is created by fusing audio and video features at the model level. Aziz *et al*. [1] proposed MMTF-DES, a unified multimodal transformer-based framework with image-text pair settings to identify human desire, sentiment, and emotion. In this paper, the authors finetuned two pre-trained multimodal transformer models, Vision-and-Language Transformer (ViLT) and Vision-and-Augmented-Language Transformer (VAuLT), as multimodal encoders to extract effective integrated contextual-visual features in the MMTF-DES model. The paper also experiments with different unimodal inputs, and found out all text models perform better than all image models by a large margin, and hence it concludes the multimodal human desire understanding task is a text modality-dominant task. It would be interesting to see whether using video instead of image would change the balance of dominance between different modalities in this study.

4. Data

The dataset we use is RAVDESS[5], which is a validated multimodal database of emotional speech and song. The database is gender balanced consisting of 24 professional actors, vocalizing same statements in a neutral North American accent. Speech includes calm, happy, sad, angry, fearful, surprise, and disgust expressions, and the song contains calm, happy, sad, angry, and fearful emotions. Each expression is produced at two levels of emotional intensity (normal, strong), with an additional neutral expression. The data split we used only consisted of speech actors, no songs. It consists total of over 1,400 video clips with resolution of 1280 * 720, with each less than 5 seconds. We randomly split the dataset into three parts, where 70% is the training set, 10% is the validation set and 20% is the test set.

To make the input to our model manageable, we preprocess each clip by sampling one frame every 0.5 seconds for a total of 6 frames, starting from 0 second. Hence we get a sequence of 6 frames for each clip, which we then use as input to our emotion recognition model. The bottom of figure 1 shows 5 of the frames sampled from one video clip. In later experiments, we also investigate into the effect of increasing frame samples to 10 or 16 per video.

We chose to use the RAVDESS dataset because of its simplicity and well-classified emotional states, which we found to be inclusive and general. In our project, we used only the raw video data and disregarded the audio part of the dataset due to time constraints. Our future research plans include incorporating the audio data to evaluate the accuracy improvement with the additional modality. Additionally, we would like to extend our work to other emotional classification benchmark datasets.

5. Methods

We implemented and experimented with several different deep learning models, and we discuss about them in this section.

5.1. Early Fusion Model

The initial architecture we employ in our project, which also serves as our baseline, is the early fusion model. This model receives input in the shape of (T, C, H, W) and straightforwardly reshapes the matrix to $(T * C, H, W)$. This process allows the model to superficially integrate temporal information from the videos. After reshaping, the input is fed into a standard 2D Convolutional Neural Network (CNN) to determine the emotion label, as shown in Figure 1. We have chosen to implement the early fusion model due to its widespread use in video classification tasks. The simplistic reshaping at the start of the 2D CNN architecture enables it to extract temporal features from the transitions between different video frames, making it an appropriate starting point for our project.

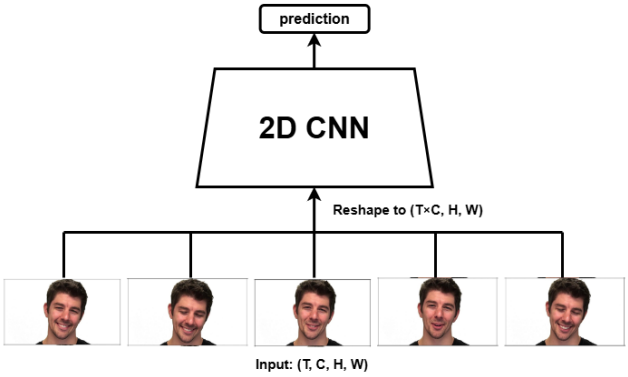

Figure 1. The structure of early fusion model. The input frames will be reshaped to be a three-dimensional matrix and be passed into a 2D CNN model to predict emotion label

The 2D CNN model used in our early fusion model has the following structure:

We use the first four convolution layers to attempt to extract deep features from the frames, and the receptive field of the last convolution layer covers the whole H×W region of the input of the first convolution layer, enabling the model to learn more comprehensive features form the raw frames. Batch Normalization and ReLU activation are applied after each convolution layer, and we also apply dropout layer after each fully connected layer to prevent potential overfitting.

	# Kernels	Kernel Size	Stride	Padding
Conv1	64	3	2	1
Conv2	128	3	2	1
Conv3	256	3	1	1
Conv4	512	3	1	1
FC1	output size: 512			
Dropout	p = 0.5			
FC2	output size: 1024			
Dropout	p = 0.5			
FC3	output size: 8			

Table 1. Early Fusion Architecture

	# Kernels	Kernel Size	Stride	Padding
Conv1	16	3	2	1
Conv2	32	3	2	1
Conv3	64	3	1	1
Conv4	128	3	1	1
FC1	output size: 512			
Concat	output size: 512*num_frames			
Dropout	p = 0.5			
FC2	output size: 1024			
Dropout	p = 0.5			
FC3	output size: 8			

Table 2. Late fusion Architecture

5.2. Late Fusion

Next, we transition to the late fusion model, which differs structurally from the early fusion model. In the late fusion model, each frame in the sequence is fed into the same 2D CNN model. The extracted feature maps, shaped (T, C', H', W'), are flattened and passed into subsequent fully connected layers to predict a single emotion label. The structure of late fusion model is demonstrated in Figure 2.

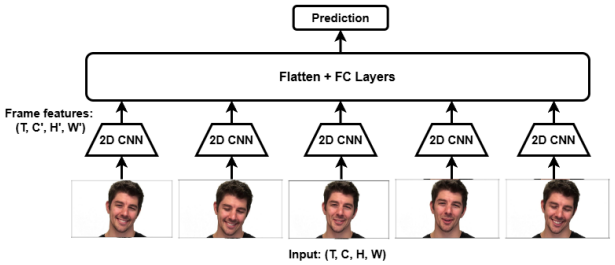

Figure 2. The structure of late fusion model. Each frame will be passed into a 2D CNN, and the resulting features extracted from all frames will be concatenated, flattened, and passed to fully connected layers to output prediction label

In contrast to early fusion model in which all raw frames are combined directly, late fusion models merge the extracted feature maps from each image, allowing each modality to be processed independently before integration. This approach facilitates the model's ability to learn better temporal features from the original video clips. By focusing on refined feature maps rather than raw data, late fusion can capture the temporal dynamics and interactions more effectively, leading to improved performance and a more nuanced understanding of the video content. This independence and specialization in processing also make late fusion models more adaptable and robust to variations and noise in the data.

The parameters in the 2D CNN used in our late fusion model are specified in Table 2.

Similar to the early fusion structure, the parameters in the late fusion model are designed to increase the receptive field and optimize the trade-off between training time and model performance. Additionally, batch normalization and dropout are incorporated to enhance model training and prevent overfitting.

5.3. 3D CNN

Both the late fusion model and the early fusion model fuse the frame sequences at certain stages before passing them into the final prediction layer. However, their ability to extract insightful and meaningful temporal information from video frames remains limited because they rely on a single layer to combine all frames, thus learning very little in the temporal dimension. To address this issue, we implement a 3D CNN model, which places greater emphasis on learning temporal features from the sequences.

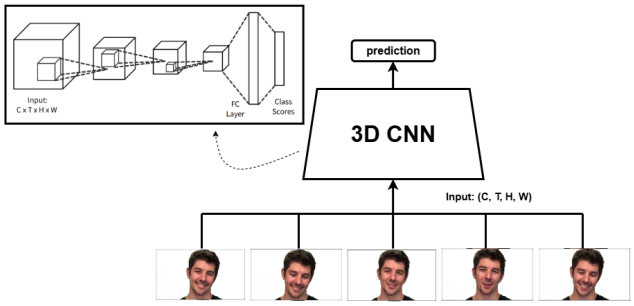

Figure 3. The structure of 3D CNN model. Instead of fusing frames or features extracted from all slices, 3D CNNs take into the video frames matrix directly without reshaping or concatenation. In each 3D convolution layer, the input is convolved with a group of three-dimensional kernels, enabling the model to learn and extract both spatial and temporal features. Fully connected layers are added at the end of the model to flatten the feature maps and predict class labels

3D Convolutional Neural Network (3D CNN or C3D) extends the capabilities of traditional 2D CNN by convolving three-dimensional kernels with 3D inputs, making them ideal for applications involving temporal sequences. Unlike 2D CNNs, which process spatial features in each frame independently, 3D CNNs analyze sequences of frames si-

multaneously, capturing both spatial and temporal information. This allows 3D CNNs to recognize motion patterns and temporal dependencies directly from the data, leading to more accurate and robust predictions in video classification tasks. Thus, we believe the 3D CNN model can offer more comprehensive insights about the relationship between neighboring frames, consequently learning more complex patterns and features along the temporal axis. The overall implementation of 3D CNN model is illustrated in Figure 3

The parameters used in our structure are listed as in Table 3. Batch normalization and ReLU activation are applied after all layers but the last fully connected layer. We also implement a 3D Max Pool layer after each 3D convolution layer to reduce the sizes of extracted feature maps, and dropout layers were added after the first two fully connected layers to prevent overfitting.

	# Kernels	Kernel Size	Stride	Pad
3D Conv1	16	3	(1, 2, 2)	1
3D Conv2	32	3	(1, 2, 2)	1
3D Conv3	64	3	(1, 1, 1)	1
3D Conv4	128	3	(1, 1, 1)	1
FC1	output size: 1024			
Dropout	p = 0.5			
FC2	output size: 512			
Dropout	p = 0.5			
FC3	output size: 8			

Table 3. 3D Convolution Neural Network

5.4. RNN-CNN

While 3D Convolutional Neural Networks (3D CNNs) excel at capturing temporal features in video data by processing sequences of frames simultaneously, they are limited in their ability to capture relationships beyond a few neighboring video slices. This limitation arises because 3D CNNs primarily focus on local temporal dependencies without considering long-term sequential information.

To address this, we transition to Recurrent Convolutional Neural Networks (RNN-CNNs), which combine the strengths of recurrent neural networks (RNNs) with CNNs to emphasize the sequential nature of video frames. In an RNN-CNN, each hidden layer takes two inputs: the feature map from the same layer at the previous time step and the feature map from the current time step but from a previous layer. This dual input mechanism enables the RNN-CNN to effectively capture both the deep spatial features in each frame and the long-term temporal dependencies between video slices. Furthermore, we introduce the Long Short-Term Memory (LSTM) structure as the RNN component in our implementation. This aims to enhance the model's ability to utilize features extracted from frames that are from the very beginning of the videos, and mitigate the issue of gradient vanishing. The structure of our RNN-CNN model is shown in Figure 4.

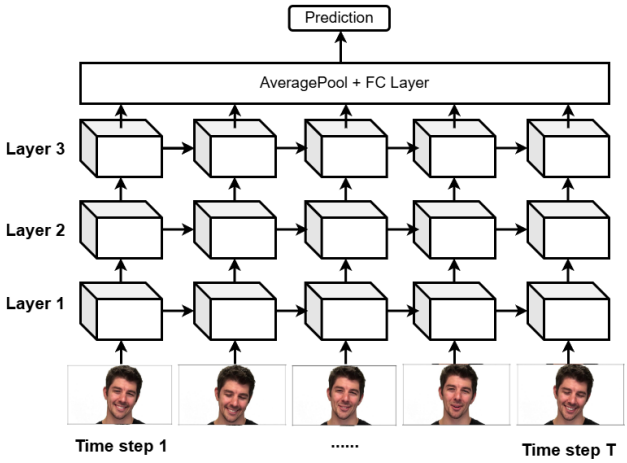

Figure 4. The structure of RNN-CNN model. The entire network uses 2D convolution layers and 2D feature maps. The outputs of each layer at each time step are dependent of both the feature map from the last layer at the same time step and the feature map from the last time step at the same layer. We implement an average layer and subsequent fully connected layer after the last layer to take all outputs from last layer into consideration to generate a prediction

The parameters used in our RNN-CNN model are listed as the following (Table 4). At each layer, the input size and the hidden size of LSTM are set to be equal to the number of output channels of that layer.

	# Kernels	Kernel Size	Stride	Pad
Conv1 + LSTM	16	3	2	1
Conv2 + LSTM	32	3	2	1
Conv3 + LSTM	64	3	2	1
Conv4 + LSTM	128	3	2	1
AveragePool	-			
FC1	output size: 1024			
Dropout	p = 0.5			
FC2	output size: 256			
Dropout	p = 0.5			
FC3	output size: 8			

Table 4. RNN-CNN model

We initially considered using a transformer to provide our emotion detection model with long-term memory as well as self-attention schemes, but eventually we chose to use RNN-CNN instead. While video transformers have the advantage of processing all frames in parallel, making them more efficient for longer sequences, our project involves relatively short video lengths with a limited number of frames. Therefore, the efficiency gain from using a transformer is not critical in our case.

6. Experiments

The experiments section is divided into three sub-sections. In the first sub-section, we will evaluate the impact of various data preprocessing techniques on model performance. Then in the second sub-section, we will compare the overall performance of above mentioned architectures: early fusion, late fusion, 3D CNN, and RNN-CNN. In the last sub-section, we will provide insightful and deeper qualitative analysis on the best model. We use accuracy as our main evaluation metric for model performance.

For all experiments except image downsize, we trained each model for 40 epochs, which is sufficient for each model to converge with no further increase in validation accuracy. We use a batch size of 10, which is not too small to make training unstable, but also not too large to make training slow. When evaluating each model on the test dataset, we take the model checkpoint after the epoch with highest validation accuracy. The experiment on image downsize uses epoch size of 25 and batch size of 3. This is primarily because models on images of original size (1280*720) has much more trainable parameters. Hence training is much slower, and we also get out of memory error on larger batch sizes.

We employ the Adam optimizer, known for its efficiency in both computational and memory requirements. After hyperparameter tuning, we configured the Adam optimizer with a learning rate of 0.001 and a weight decay of 0.0005 to prevent overfitting. This configuration allows all models in our experiment to converge at a reasonable speed. Additionally, we use a learning rate scheduler to reduce the learning rate by a factor of 0.5 if the monitored metric does not improve for 2 consecutive epochs. This helps to fine-tune the training process and improve model performance.

6.1. data processing techniques

6.1.1 Image downsize

The original image frames extracted from the video have a very high resolution of 1280 by 720 pixels. Training models on these high-resolution images is very slow. To address this, we evaluate the effectiveness of training with original images (1280x720) versus downsized images (224x224). We hypothesize that training with downsized images will be much faster due to the significantly reduced number of convolution operations. Additionally, we expect that the accuracy on the datasets will remain similar. The image size of 224x224 is commonly used in pre-trained vision models such as VGG and ResNet, which suggests that downsizing to this resolution should still retain the essential facial details necessary for classifying emotions accurately.

In this experiment, we trained the early fusion model and late fusion model on both the original dataset and downsized dataset. Table 5 shows the results from both models. As shown in the table, the training time per epoch is significantly shorter for the downsized dataset. Surprisingly, the test accuracy is also higher for the downsized dataset. This may be because it is easier for the model to identify emotion-indicating regions in smaller images.

Due to the overwhelming benefits of using the downsized dataset, and given the time constraints in running different models, we decided to use only downsized dataset (224*224) for all remaining experiments.

	Training time per epoch	Test Accuracy
Early Fusion	3 minutes $\rightarrow$ 6 seconds	59.7 $\rightarrow$ 60.5
Late Fusion	3 minutes $\rightarrow$ 5 seconds	63.5 $\rightarrow$ 66.4

Table 5. Image Resize: number before the arrow is result from the original dataset, and number after the arrow is the result from the downsized dataset. We see downsize dataset is better in terms of training time and test accuracy.

6.1.2 Data Augmentation

Data augmentation is a techinique often used for improving model robustness and generalizability. In this experiment, we evaluated the model performance on data-augmented dataset. We apply the following image transformations with a certain probability during training:

- Randomly adjust the contrast, saturation, hue, brightness, and also randomly permutes channels (with probability of 0.5). We used PyTorch's `RandomPhotometricDistort()` function with default parameters.
- Horizontal Flip (with probability of 0.5).

Table 6 shows the model accuracy without and with data augmentation for all three models using our downsized dataset. We see that late fusion performances better with data augmentation, while the performance deteriorated for early fusion and 3D CNN. As a result, there is no consistent results that indicate data augmentation improves model performance. Table 7 provides more results where we do not see consistent benefit of data augmentation either. This might be because our model already learns robust feature representations for each emotion even without data augmentation. It is also possible that our current data augmentation scheme is not optimal, and we need to change the transformations used in order to get better results.

6.1.3 Sample more frames

Initially, we sampled 6 frames from each video. We also created additional datasets with 10 and 16 frames per video, where the frames are equally spaced in time. We hypothesize that including more frames as inputs to the model

	Test Accuracy
Early Fusion	71.1 → 63.4
Late Fusion	66.6 → 72.8
3D CNN	77.4 → 70.4
RNN-CNN	70.4 → 62.7

Table 6. Data Augmentation: number before the arrow is without data augmentation, and number after the arrow is with data augmentation. We do not see consistent benefit of data augmentation.

will increase its accuracy. This is because a larger number of frames improves the likelihood of capturing frames that clearly indicate specific emotions.

Table 7 shows the result with different number of samples. As expected, for each model architecture, training with dataset of more frame samples per video leads to higher test accuracy. This leads us to think that for all computer vision tasks on videos, sampling frequency should be an important hyperparameter on the final model performance. It appears that the model performances is positively correlated with the number of frames per video, but the performance gain will diminish at some point.

	6 Frames	10 Frames	16 Frames
Early Fusion	71.1	74.7	76.5
Late Fusion	66.6	81.3	80.3
3D CNN	77.4	83.7	84.4
RNN-CNN	70.4	76.8	80.6

Table 7. Sample more frames: more frames leads to better performance

6.2. Model Comparison

In this section, we conducted thorough experimentation with all model/dataset combinations. For each model architecture, we trained it on datasets with different number of frames and with/without data augmentation. Table 8 shows the results of all combinations, and it's very clear that 3D CNN has the best performance both in terms of best accuracy and average accuracy. Late fusion is the next best, whereas Early fusion and RNN-CNN has the lowest accuracy. The order of accuracies of 3D CNN, late fusion and early fusion is consistent with what we learned in the lecture on video classifications, though the RNN-CNN performance is worse than expected. We think RNN-CNN might require more architecture tuning in order to further improves the accuracy, since its architecture is more complex than the other three models.

6.3. Qualitative Analysis

In this section, we further analyze the performance of the best model from section 6.2. The best performing model is 3D CNN model trained on dataset with 16 frames and no data augmentation.

Among all the emotions, the model performs best for neutral expressions, achieving 100% accuracy, as shown in Figure 5. It also achieves a high accuracy of 94.7% for disgust expressions. However, the model struggles with surprised and fearful expressions, reaching only about 75% accuracy for these emotions. This difficulty arises because surprised and fearful expressions have similar facial features, making them hard to distinguish. Interestingly, our model achieved very high accuracy for disgust expressions, which previous research identified as one of the hardest emotions to distinguish.

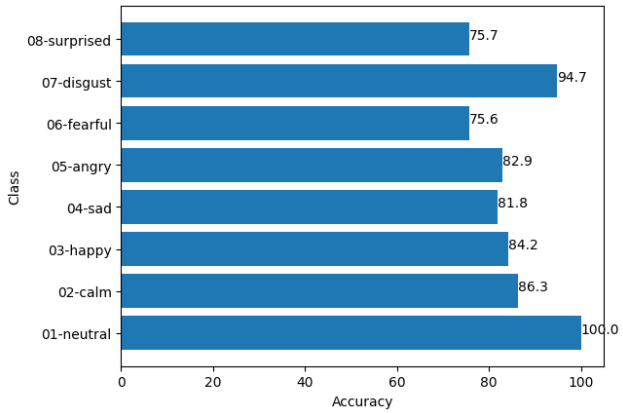

Figure 5. model accuracy on different classes. The model performs best on neutral expressions and worst on surprised and fearful expressions.

Figure 6 shows the confusion matrix, where we summarize model prediction in matrix form. We see that two errors occurs the most (5 times), and these are: predicting calm as neutral, predicting angry as sad. Calm and neutral are hard to distinguish with only image input, since they have very similar expressions. This can be illustrated by their average images in Figure 7. Angry and sad would be easy to differentiate from human's perspective, but these two expressions do have similar features, such as frowned face, where it is easy for model to get confused.

Figure 8 shows some examples misclassified by the model. Some of the examples shown are hard for us to classify as well, such as the fourth and fifth one. It would be much easier for us if we can listen to the actor's speech. This indicates that video-only emotion classification might not be the most effective, and adding audio as another modality could greatly improve model accuracy. We also notice that even though some examples have the same emotion label, the facial expression of the actor can be pretty different. For example, the true class for second and third row is 'angry', though it is much more obvious for a human to determine third row as 'angry' than the second row.

	6 Frames	6 Frames +Augment	10 Frames	10 Frames +Augment	16 Frames	16 Frames +Augment	*Average Accuracy*
Early Fusion	71.1	63.4	74.7	77.2	76.5	**80.3**	*73.9*
Late Fusion	66.6	72.8	81.3	**81.7**	80.3	77.5	*76.7*
3D CNN	77.4	70.4	83.7	76.8	**84.4**	78.5	*78.5*
RNN-CNN	70.4	62.7	76.8	78.5	**80.6**	72.0	*73.5*

Table 8. Model accuracy on datasets with different processing techniques. Best accuracy for each model is highlighted. Average accuracy is computed per row. The order of performance is: 3D CNN $>$ late fusion $>$ early fusion $\approx$ RNN-CNN.

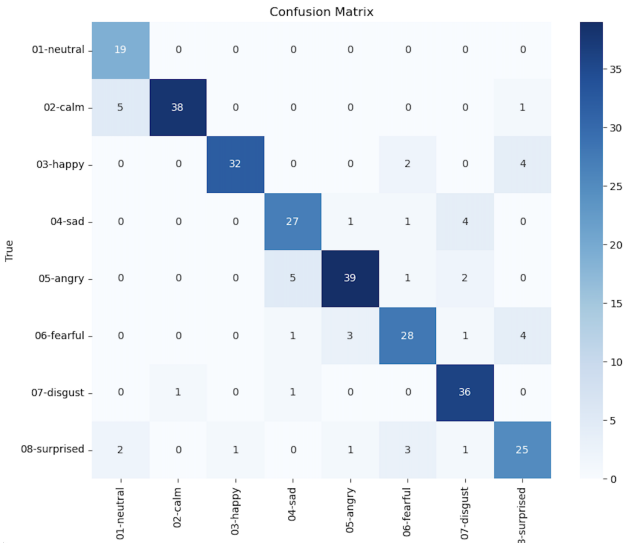

Figure 6. Confusion matrix. X axis is predicted class and Y axis is true class. Two errors that occurs the most are: predicting calm as neutral, predicting angry as sad.

Figure 7. Average image for each class.

Figure 8. Examples of misclassified images. Due to space constraints, only 6 out of 16 frames are shown for each example.

7. Conclusion/Future Work

In this project, we explored four model architectures, which are early fusion, late fusion, 3D CNN and RNN-CNN, on the task of emotion recognition on videos. We found that 3D CNN performs best both in terms of best accuracy and average accuracy. This is expected because 3D CNN places greater emphasis on learning temporal features from the sequences than early fusion or late fusion. We are also surprised that RNN-CNN is not the best performing model, and it might be because the architecture is more complex and hence requires more hyperparameter tuning. We also experimented with different data processing techniques. We found image downsizing and increasing number of frame samples to be useful in improving test accuracy, and image downsizing also reduce training time significantly. Data augmentation, however, does not consistently improve test accuracy in our experiments.

In future research, we aim to incorporate more advanced and powerful models such as Transformers and the Deep Alignment Network (DAN), which can potentially enhance our emotion detection system by focusing more precisely on specific facial features like eyebrows and lips [12]. While the dataset used in our current project has been beneficial, it is somewhat simplistic. For future work, we plan to employ a more realistic dataset, featuring people speaking and gesturing in natural settings. This will help train our model to be more reliable and applicable in everyday scenarios.

Additionally, we intend to integrate multi-modal structures into our future projects. By combining models that analyze the emotional content of speech and voice, we can capture non-visual cues, which play a significant role in revealing human emotions, and these non-visual information is necessary if we want to build a more comprehensive and nuanced emotion detection system.

8. Contributions & Acknowledgements

All teammates contribute equally in this project. Below is the work done by each teammate:

- Senyang Jiang: Model architecture tuning, model training and result collection.
- Suxi Li: Literature review, model architecture tuning, results interpretation.
- Yichen Jiang: Model architecture writing and data processing.

We want to thank our CA Anwesha for providing guidance in determining project scope and writing final report.

References

[1] A. Aziz, N. K. Chowdhury, M. A. Kabir, A. N. Chy, and M. J. Siddique. Mmtf-des: A fusion of multimodal transformer models for desire, emotion, and sentiment analysis of social media data, 2023.

[2] P. Ekman and W. V. Friesen. Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17(2):124, 1971.

[3] Y. Fan, X. Lu, D. Li, and Y. Liu. Video-based emotion recognition using cnn-rnn and c3d hybrid networks. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction (ICMI '16)*, pages 445–450. ACM, 2016.

[4] A. Karpathy, T. Leung, G. Toderici, R. Sukthankar, S. Shetty, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[5] S. Livingstone and F. Russo. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PLoS ONE*, 13(5):e0196391, 2018.

[6] H. V. Manalu and A. P. Rifai. Detection of human emotions through facial expressions using hybrid convolutional neural network-recurrent neural network algorithm. *Intelligent Systems with Applications*, 21:200339, 2024.

[7] A. Mehrabian and S. Ferris. Inference of attitudes from nonverbal communication in two channels. *Journal of consulting psychology*, 31 3:248–52, 1967.

[8] A. Middya, B. Nag, and S. Roy. Deep learning based multimodal emotion recognition using model-level fusion of audio-visual modalities. *Knowledge-Based Systems*, 244:108580, 03 2022.

[9] S. Minaee and A. Abdolrashidi. Deep-emotion: Facial expression recognition using attentional convolutional network. *arXiv preprint arXiv:1902.01019*, Feb 2019.

[10] B. Pan, K. Hirota, Z. Jia, and Y. Dai. A review of multimodal emotion recognition from datasets, preprocessing, features, and fusion methods. *Neurocomputing*, 561:126866, Dec 2023.

[11] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*, 2014. Version 2, 12 Nov 2014.

[12] I. Tautkute and T. Trzcinski. Classifying and visualizing emotions with emotional dan. *Fundamenta Informaticae*, 160(1-2):1001–1016, 2018.