CS 6140 Project Proposal: Facial Expression Recognition Using Deep Learning

1. Project Overview This project aims to develop an advanced facial expression recognition system leveraging deep learning techniques. The model will utilize convolutional neural networks (CNNs) to detect and classify facial expressions from images. The research will focus on improving recognition accuracy by implementing depthwise separable convolutional networks and comparing the performance of various deep learning architectures, including VGG and ResNet.

2. Data Selection The primary dataset for this project is the FER-2013 dataset, which comprises 35,886 grayscale images of human faces labeled with seven distinct facial expressions: anger, disgust, fear, happiness, sadness, surprise, and neutral. The dataset is sourced from a Kaggle competition and is known for its real-world complexity and variability. The dataset is accessible at Kaggle FER-2013 Dataset.
link: https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data

3. Project Completion Plan To achieve the project objectives, the following steps will be taken:

- **Data Preprocessing**: Normalize images to 48×48 pixels, apply face detection using MTCNN, and enhance robustness using Kernelized Correlation Filters (KCF).
- **Model Selection and Implementation**:
    - Implement and evaluate CNN-based architectures including VGG and ResNet.
    - Optimize models using techniques such as dropout regularization and batch normalization.
    - Train the models using the FER-2013 dataset and compare their performance.
- **Evaluation and Analysis**:
    - Use accuracy, precision, recall, and confusion matrices to assess model performance.
    - Analyze misclassification trends and potential biases.

4. Tools and Libraries The implementation will utilize the following tools and libraries:

- **Programming Language**: Python
- **Deep Learning Frameworks**: PyTorch

- **Machine Learning Libraries**: Scikit-learn, OpenCV
- **Data Handling**: Pandas, NumPy
- **Visualization**: Matplotlib, Seaborn

5. Models and Implementation Details

- **CNN Model**: A hierarchical CNN with convolutional, max-pooling, and fully connected layers.
- **VGG-Inspired Model**: Multiple convolutional layers with increasing depth and ReLU activations.
- **ResNet Model**: Deep residual learning framework to mitigate vanishing gradients.
- **MTCNN & KCF**: Used for face detection and tracking.

6. Additional Resources and Components

- GPU for model training acceleration, Nvidia RTX 4080 super.
- Hyperparameter tuning strategies (learning rate adjustment, batch size optimization).
- Data augmentation techniques to enhance model generalization.

7. Expected Outcomes and Visualizations

- Confusion matrices displaying model classification performance.
- Graphs depicting training loss and accuracy over epochs.
- Comparative performance metrics between CNN, VGG, and ResNet architectures.
- Real-time facial expression recognition demonstration.

8. Timeline and Milestones

- **Week 10**: Data preprocessing and exploratory data analysis. Model implementation and preliminary training.
- **Week 11-12**: Model optimization and performance evaluation. Comparative analysis and results visualization.
- **Week 13**: Final report preparation and presentation.

This project will contribute to the field of human-computer interaction by enhancing the accuracy and efficiency of facial expression recognition systems. The outcomes will help advance AI-driven applications in security, healthcare, and customer engagement.