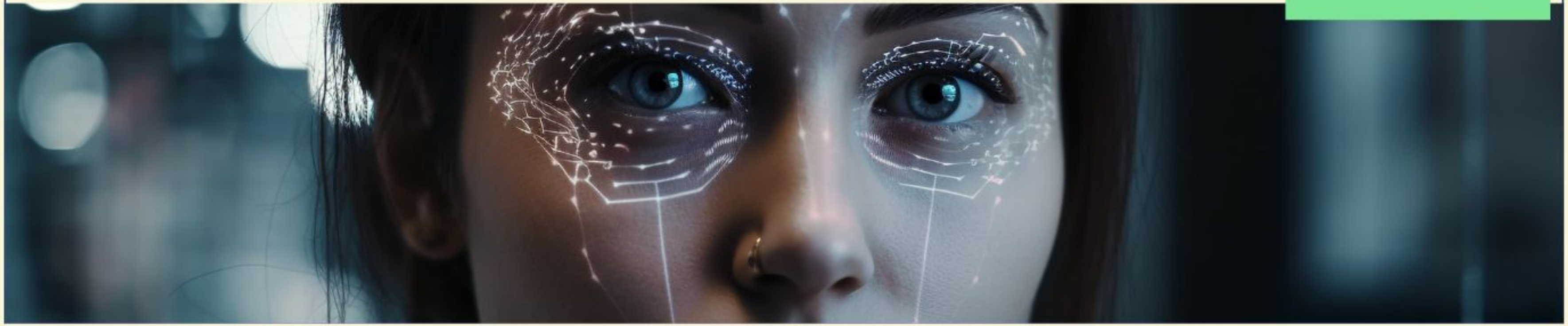
Abhishek Bhadre
Parth Domadia

CV Project



Team Members of the CV Project

Introducing the core team behind the CV Project

- **Abhishek Bhadre**

  A20580236

- **Parth Domadia**

  A20586569

FACIAL EXPRESSION RECOGNITION

Developing a Robust Facial Expression Recognition System

This project explores a self-supervised approach utilizing Heatmap Neighbor Contrastive Learning to enhance FER, comparing it with MoCov2 for effectiveness.

Advancements in Facial Expression Recognition

Importance of FER

Facial Expression Recognition is essential for affective computing and enhancing human-computer interactions.

Applications of FER

Used in emotion-aware assistants, mental health diagnostics, and behavioral analysis.

Challenges in Traditional FER

Traditional models rely on labeled datasets, facing issues with label inconsistencies and high costs.

Generalization Limitations

Existing models struggle to generalize to unseen data, limiting their effectiveness.

HNCL Approach

HNCL enhances FER by creating positive pairs from facial landmark heatmaps, improving representation learning.

Intra-Class Representation

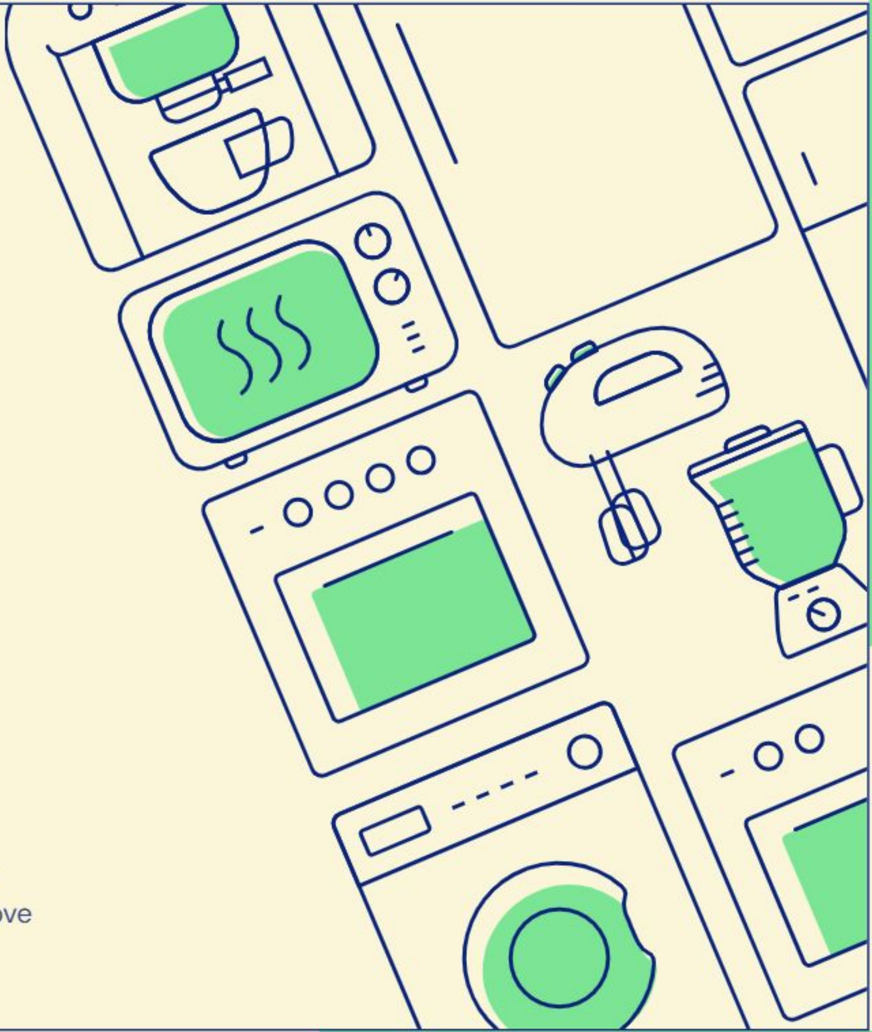
The method captures semantic similarities in expressions, addressing weaknesses in augmentation methods.

Comparison with MoCov2

HNCL outperforms traditional augmentation-based methods like MoCov2 in representation learning.

FACIAL EXPRESSION RECOGNITION

Enhancing Facial Expression Recognition with HNCL



This presentation explores the challenges of supervised facial expression recognition and introduces HNCL as a self-supervised approach to improve generalization across datasets.

Exploring Contrastive Learning Techniques

Insights into Contrastive Learning and HNCL Techniques

- Contrastive Learning Overview
  - Contrastive learning pulls similar samples together while pushing different ones apart, key for self-supervised learning.
- Frameworks and Achievements
  - Models like SimCLR and MoCo excel in classification and object detection, demonstrating the power of contrastive methods.
- Supervised Contrastive Learning
  - Extends contrastive learning by treating same-class samples as positives, enhancing class-wise clustering and performance.
- Domain Adaptation Techniques
  - Self-supervised methods use tasks like rotation prediction to transfer knowledge across domains without labeled data.
- Unified View of Contrastive Learning
  - Highlights strengths such as invariance, disentanglement, and structure across supervised and unsupervised settings.
- Heatmap-Based Representation
  - HNCL uses facial landmark heatmaps to identify semantically similar neighbors for contrastive training.
- Intra-Class Variation
  - HNCL enhances representation by adding meaningful intra-class variation, improving contrastive training outcomes.
- Generalization Across Datasets
  - Combining spatial cues with contrastive learning improves generalization in datasets with label inconsistencies.
- Balancing Supervised and Self-Supervised
  - HNCL effectively balances supervised and self-supervised approaches in facial emotion recognition, ensuring high performance.
- Real-World Applications
  - HNCL is ideal for scenarios with limited or noisy labels, providing a robust solution for real-world applications.

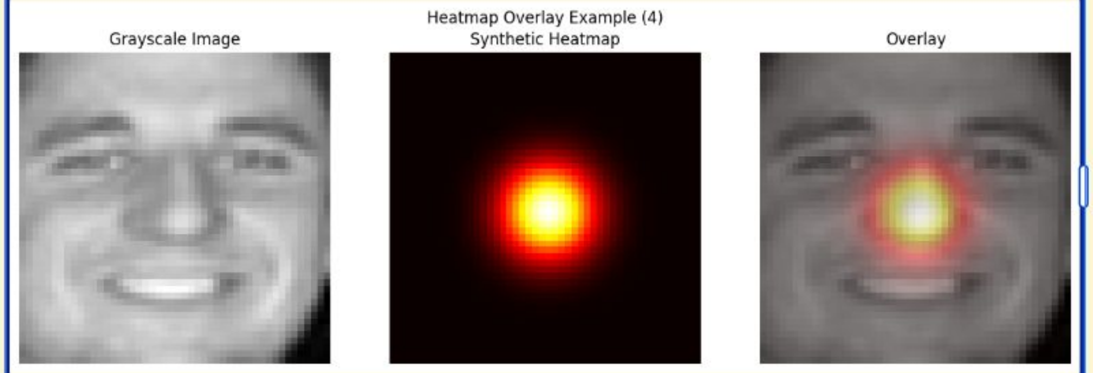
Facial Expression Recognition Datasets

Overview of datasets used for FER tasks

	a	b	c
1		RAF-DB Count	FER-2013 Count
2	Angry	1290	3995
3	Disgust	281	436
4	Fear	717	4097
5	Happy	4772	7215
6	Sad	1982	4830
7	Surprise	705	3171
8	Neutral	2524	4965

Data Preparation for CV Models

Ensuring Robustness in Computer Vision Models



Importance of Data Preparation

Essential for model robustness and generalization, especially with imbalanced datasets.

Data Augmentation Techniques

Enhances model generalization with random flips, rotations, and color jittering.

Face Detection and Cropping

Faces aligned using Haar Cascades or dlib's landmark detector to remove background noise.

Heatmap Generation for Attention

Synthetic heatmaps generated to simulate facial landmark attention with Gaussian distribution.

Grayscale Conversion and Resizing

Images converted to grayscale and resized to 48×48 pixels for uniformity.

Comparative Overview of Model Architectures

Exploring MoCo v2 and HNCL for Facial Expression Recognition

MoCo v2 Framework

MoCo v2 utilizes a momentum encoder and dynamic queue for negative samples, enhancing contrastive learning efficiency.

Wide ResNet-50-2 Backbone

Both models leverage the Wide ResNet-50-2 architecture, adapted for 2-channel input.

Contrastive Learning Mechanism

Employs InfoNCE loss for effective representation learning between query and key samples.

Momentum Encoder Updates

The momentum encoder is updated with an EMA to maintain stable key representations.

HNCL Innovations

HNCL builds on MoCo by integrating semantic neighborhood supervision for improved feature learning.

Heatmap Neighbor Selection

Uses heatmap similarity to identify k-nearest neighbors for enhanced positive sampling.

Positive Pair Expansion

Treats heatmap-similar neighbors as positive samples to enrich the learning framework.

3-Component Model Structure

HNCL includes an encoder, projector, and a predictor to prevent representation collapse.

Downstream Classifier Head

A classifier is used for fine-tuning and evaluation on labeled datasets, enhancing model applicability.

Instructions for Model Implementation

- **Folder Structure Overview**

  The project has a well-defined folder structure for models and datasets.

- **Environment Setup Steps**

  Create a virtual environment and install the necessary packages using requirements.txt.

- **Running HNCL Pretraining**

  Execute HNCL.py to train the encoder with heatmap-based pairs and save weights.

- **Running MoCov Pretraining**

  Execute Mocov2.py for instance discrimination training and log statistics.

- **Evaluation Metrics**

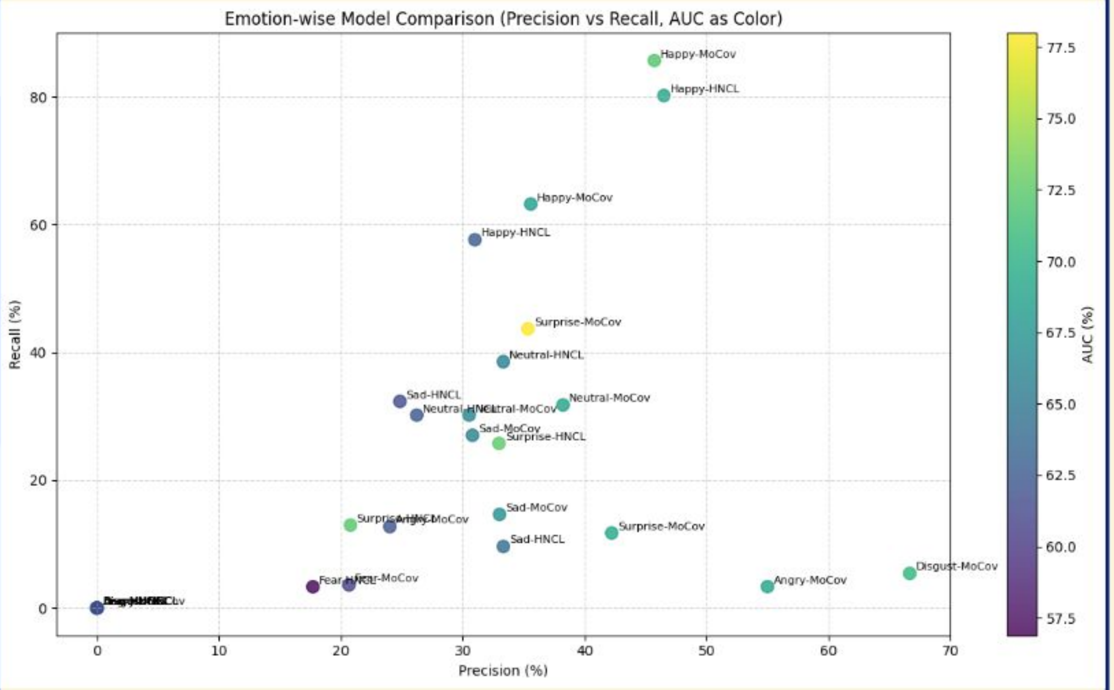
  After training, evaluation metrics include Accuracy, Precision, Recall, F1-Score, and ROC-AUC.

- **Configuration Notes**

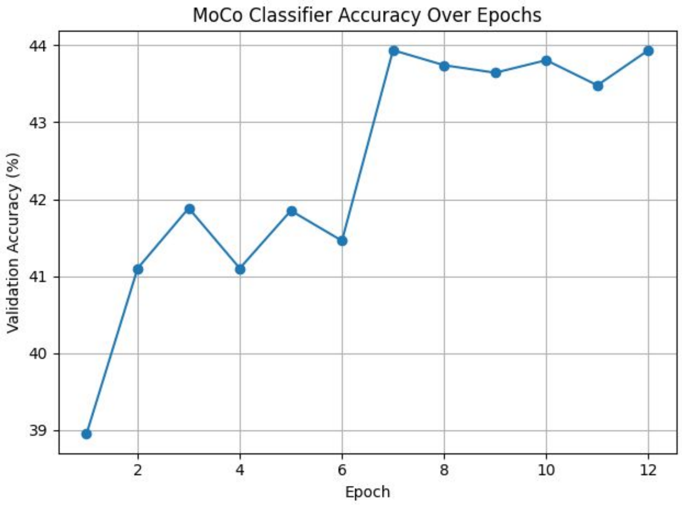
  Ensure paths and dataset folders are correctly set in the scripts for proper execution.

Results and Discussion of Emotion Models

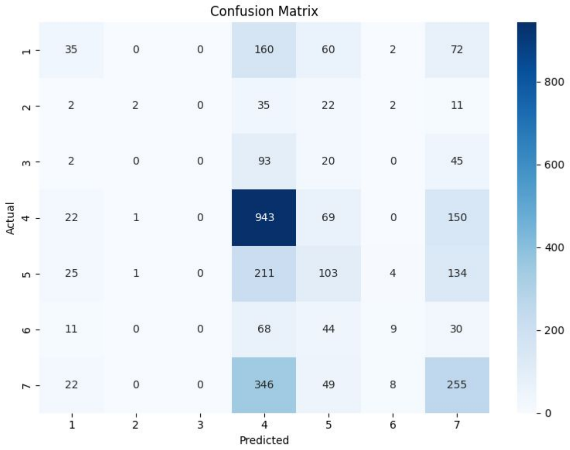
Comparative performance of emotion recognition models



Mocov Plots

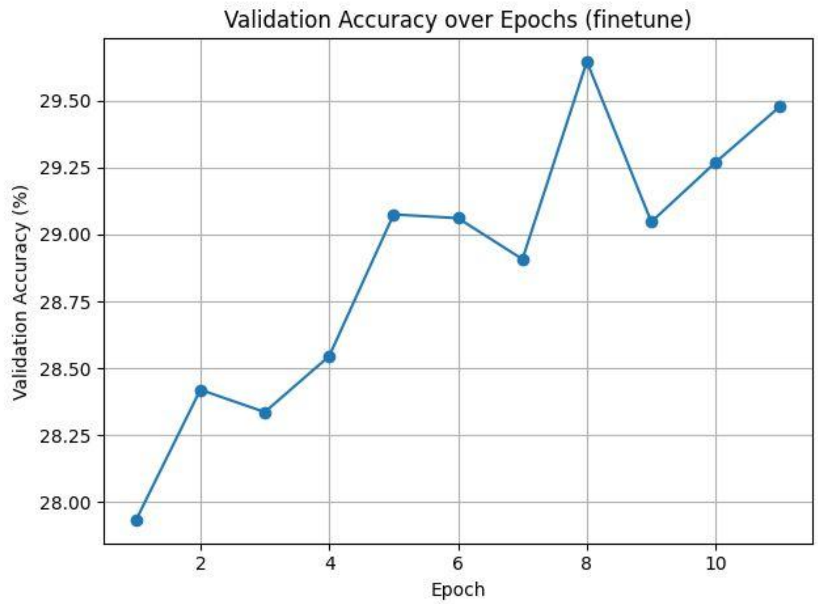


MoCo Classifier Accuracy Over Epochs
Validation Accuracy (%)
Epoch
39
40
41
42
43
44
2
4
6
8
10
12

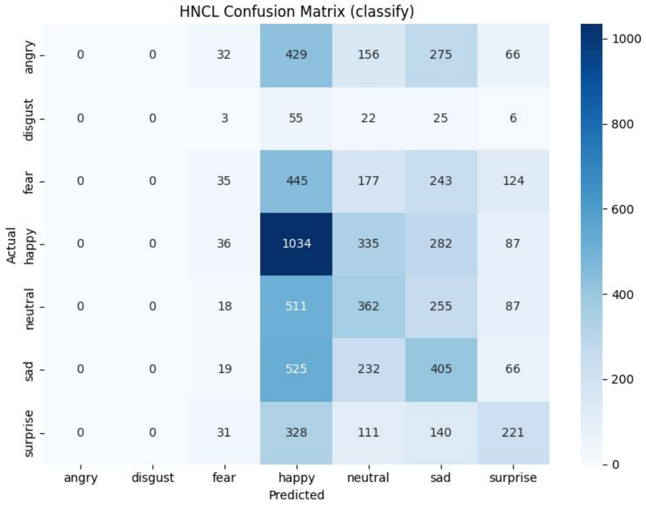


Confusion Matrix
Actual
Predicted
35 0 0 160 60 2 72
2 2 0 35 22 2 11
2 0 0 93 20 0 45
22 1 0 943 69 0 150
25 1 0 211 103 4 134
11 0 0 68 44 9 30
22 0 0 346 49 8 255
1
2
3
4
5
6
7
0
200
400
600
800

HNCL Plots



Validation Accuracy over Epochs (finetune)
Validation Accuracy (%)
29.50
29.25
29.00
28.75
28.50
28.25
28.00
2
4
6
8
10
Epoch



HNCL Confusion Matrix (classify)
Actual
Predicted
angry
disgust
fear
happy
neutral
sad
surprise
0 0 32 429 156 275 66
0 0 3 55 22 25 6
0 0 35 445 177 243 124
0 0 36 1034 335 282 87
0 0 18 511 362 255 87
0 0 19 525 232 405 66
0 0 31 328 111 140 221
1000
800
600
400
200
0

Performance Comparison of HNCL vs MoCov

Analyzing the strengths of HNCL and MoCov models

- **HNCL Outperforms MoCov**

  HNCL consistently shows superior performance, especially on classes with subtle expression differences.

- **Competitive Nature of MoCov**

  Despite its slower training, MoCov remains competitive due to fewer architectural components.

- **Generalization of HNCL**

  HNCL demonstrates better generalization on underrepresented classes in FER-2013 dataset.

- **Training Time Trade-off**

  HNCL incurs a minor trade-off in training time due to heatmap processing and neighbor matching.

Performance Comparison of HNCL vs MoCov

Analyzing the strengths of HNCL and MoCov models

- **HNCL Outperforms MoCov**

  HNCL consistently shows superior performance, especially on classes with subtle expression differences.

- **Competitive Nature of MoCov**

  Despite its slower training, MoCov remains competitive due to fewer architectural components.

- **Generalization of HNCL**

  HNCL demonstrates better generalization on underrepresented classes in FER-2013 dataset.

- **Training Time Trade-off**

  HNCL incurs a minor trade-off in training time due to heatmap processing and neighbor matching.