# Development of a Multi-Scale Model for Enhanced Performance by Using a Modified Soft-Margin SoftMax Loss and its Application for Facial Expression Recognition

By
Armin Nabaei

Admin Supervisor:

Dr. Zahangir Kabir

Department of Electrical and Computer Engineering Concordia University

Montréal, Québec, Canada

May 2024

# Content Presentation:

**Introduction**

- Facial Expression Recognition (FER)

**Literature Review**

- FER models
- CNN vision networks
- Cross-Entropy and SoftMax

**Methodologies**

-Model

-SoftMax Loss

**Results**

-Model Evaluation

-SoftMax Loss Evaluation

-Framework Evaluation on FER Datasets

**Conclusion**

# Introduction

## Early Neural Network Models for Facial Expression Recognition (FER):

- Classification via Multi-layer Perceptron (MLP) or RBF
- Convolutional Neural Networks (CNN) emerged in the 2013 FER Workshop [1]
- Integrating hand-crafted modules like SIFT with CNN [2] [3]
- Evolution led to hybrid models [4]
- Progression to attention-based hybrid methods [5]

## Facial Expression Learning Objectives:

- Extract discriminative features
- Leverage low-entropy, highly expressive features:
- Overcome limited data challenges and discuss overfitting risks
- Handle low-resolution images
- Address intra-class variability and inter-class similarity

## Proposed Solutions in Automated Facial Expression Recognition (AFER) [6] [7] [8] [9] [10] [11] [12] [13]:

- Introduce auxiliary loss functions for regularization or augmented input
- Use multi-scale inputs to learn local features and understand comprehensive feature map structure
- Apply label smoothing to provide better differentiation between similar classes
- Implement Deep Metric Learning (DML) to encourage generation of consistent feature vectors
- Use suitable optimization techniques to manage learning rate mitigation linearity or exponentiality

# LITRETURE REVIEW

## Main CNN Challenges:

- Handling noisy real-world data
- Choosing the correct hypothesis
- Incorporating inductive bias
- Depth causing diminishing spatial resolution
- Absence of long-range dependencies

## Characteristics of Existing CNN Vision Networks:

Vision Network Architectures Typically inspire and influence each other. The key design factor for all networks is efficient Convolutions

VGG [14] -> ResNet [15]- > DenseNet [16]

NIN [17] - > Inception [18]

**Inception Properties:**

- Uses an auxiliary classifier to improve training stability and enhance gradient signals.
- Enhances the network's ability to learn complex patterns at various scales.
- Employs Label-Smoothing Regularization (LSR) for cross-entropy loss.
- Inception U-Net **[19]**introduced a deep U-Net architecture.

**DenseNet Properties:**

- Each layer accesses feature maps from all previous layers.
- Enhances feature propagation and reuse.
- Maintains a global network state via feature maps.
- A compact, easy-to-train model that is parameter-efficient.
- Efficiently operates with narrower layers.
- Dense U-Net [20] introduced a deep U-Net architecture.

**ResNet Properties:**

- Uses residual modules to reduce the need for numerous parameters [21], [22].
- Varying model depths without significantly increasing resource usage allow detailed exploration of complex patterns.
- Balances depth and computational efficiency.
- Deep Residual U-Net [15] introduced a deep U-Net architecture.

**U-Net Properties:**

- The contraction/analysis path operates like a conventional convolutional network.
- The initial U-Net segment functions similarly to the VGG network.
- The expansion/synthesis path consists of transposed convolution layers.
- The U-shaped design is crucial for context-based learning.
- Skip connections recover spatial information lost during processing.
- Enhanced model performance through loss penalties.
- Efficient GPU memory usage and feature extraction from scaled images.

**VGG Properties:**

- Highlights the importance of depth in visual representation frameworks.
- Demonstrates robust generalization across various datasets.
- Suited for localization and classification tasks.
- Normalization (Local Response Normalization) is not used.
- VGG-UNET [23] introduced a deep U-Net architecture.

## Error Function

1) $\text{Error} = \mathrm{E}[(\widehat{\theta_m} - \theta)^2] = \text{Bias}(\widehat{\theta_m})^2 + \text{Var}(\theta_m) + \text{Irreducible error}$

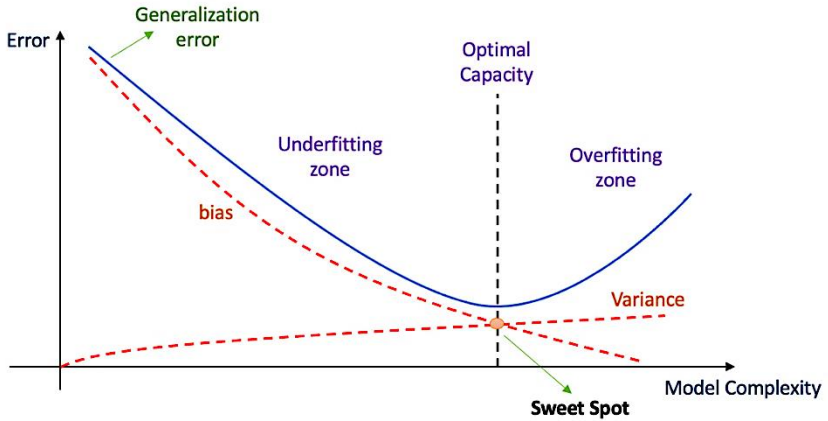
**Error Function Overview:**

- **Bias:** Difference between average predictions and actual values.
- **Variance:** Inconsistency in predictions, showing data spread.
- **Irreducible Error:** Noise inherent in data.

**Model Behavior:**

- Few parameters: High bias, low variance.
- Many parameters: Low bias, high variance.

**Goal:** Balance bias and variance to minimize overall error.



## Noise-Robust Loss Functions:

**Cross Entropy (CE):** Effective for handling both closed-set and open-set data.

2) $$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log\left(\left(p(y_i)\right)\right) + (1 - y_i) \cdot \log(1 - p(y_i))$$

**Key Points on Loss Functions:**

- **Purpose:** Adjust weights during backpropagation to minimize error and enhance prediction.
- **Strategic Selection:** Methods that select hyperparameters for penalty in cross-entropy can effectively project samples in embedding space.
- **Benefits:** Improves early learning speeds and accelerates convergence.

**Challenges in Model Training:**

- **Overfitting:** Minimizing the error function improves predictions but can lead to overfitting when weights become too large or 'saturated.'
- **Overfitting Risk is** particularly high at the start of training due to noisy labels.
- **Saturated Neural Units:** Produce minimal gradients, impacting learning.
- **Ignoring Hard Samples:** Reduces the effective number of samples for learning, impacting model performance.

**Solution:** Refine learning strategies with advanced regularization methods to enhance backpropagation.

## Properties of CE Using Regularization Techniques:

**Note :** learning is the design of a function, not only selecting parameters

- **Optimization:** Adjustments to cross-entropy address optimization challenges.
- **Focus on Predictions:** Increases attention on well-predicted samples.
- **Emphasis on Difficulty:** Concentrates more on challenging samples.

## Enhancing CE Performance: Addressing Imbalanced Datasets in Facial Expression Recognition (FER) for Robust learning.

- Real-World Weighted Cross-Entropy (RWWCE) [24] introduced a notable regularization technique in CE

3) $$L = -\frac{1}{M}\sum_{m=1}^{M}[\,W_{f_n} \times y\,\log\big(h_\theta(x)\big) + W_{f_p} \times (1-y) \times \log(1-h_\theta(x_m))]$$

$W_{f_p}$ = false positives, $W_{f_n}$ = false negatives

- Modified Cross-Entropy Method by R. Machlev [25] Introduces a cut-off point to penalize errors, altering sample weighting.

- Enhanced CE introduced by B. Shan and P.Li [26], [27] Utilizes a parameter to emphasize the importance of negative predictions, also validated as an efficient improvement by Xingiun [28]

4) $$L = -[(1-\alpha_i).\sum_{t=1}^{n}\log P_\theta(y_t|y<t,x) + \alpha_i\,.\sum_{t=1}^{n}\log P_\theta\,(\hat{y}_t|y<t,x)]$$

Where hyperparameter $\alpha$ is: $\alpha_i = m.\frac{i}{total\_iter} \implies 1 \le i \le total_{iter}\ ,\ m = 0.5.$

- The scale version of CE proposed by H Li [29] contains $\alpha$ and $\beta$ serve as balancing parameters

$$L_{overall} = \alpha . L_{Active} + \beta . L_{passive} \quad 5)$$

- Uniform weight distribution across class samples often leads to errors. This contrasts with static weighting [30], which balances minority and majority classes but doesn't distinguish between hard and easy samples like Focal Loss [31] does.

# SoftMax Function

$$6) \qquad L_{SoftMax} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{w_{y_i}^T f(x_i)+b_i}}{\sum_{j=1}^{c} e^{w_j^T f(x_i)+b_j}}$$

SoftMax by Bridle [33] derived from the Gibbs distribution, designed to optimize particle distribution in containers.

**Challenges:**

- Managing underflow and overflow risks in log and exponential operations.
- Difficulty distinguishing closely positioned classes in embedding space.
- Encouraging clear decision boundaries between classes.
- Reducing predicted samples near decision boundaries.

**Objective:**

- Enhance the model's discriminative ability by increasing the margin between classes and reducing distances within classes.
- Common enhancement feature is incorporation of margin in SoftMax terminology.
- Improve SoftMax-Loss [32] is a crucial step to leverage the strength of SoftMax function and Cross-Entropy loss.

$$\text{Softmax Loss (Multi Class Logistic Loss)} = \text{Softmax Activation} + \text{Cross Entropy}$$

## Improving Classification with Margin-Based SoftMax

if a sample is classified as class 1 or class 2, the decision boundary by conventional softmax is determined as:

$$7) \quad \left(W_1^T - W_2^T\right) x = 0$$

simplified to:

$$8) \quad (\|W_1\| \cos\theta_1 - \|W_2\| \cos\theta_2)\, \|x\| = 0$$

In Angular SoftMax by Yutian Li [33], the decision boundary is angular boundary and indicates if a sample belongs to class1 then

$$9) \quad \cos(\theta_1) > \cos(\theta_2)$$

This condition is met with a margin multiplication term and can be expressed accordingly.

$$10) \quad m\theta_1 < \theta_2$$

Here, 'm' (a positive integer) represents the distance margin for sample 'x' belonging to class 1.

**A-softmax introduces two properties:**

- **Normalization and bias zeroing**: The first adjustment involves normalizing the weight vectors (||W1|| = ||W2|| = 1) and zeros bias.
- **Incorporation of angular margin**: By incorporating an integer margin parameter m, the decision rule becomes based on whether cos(m$\theta$1) - cos(m$\theta$2) > 0 or cos(m$\theta$2) - cos(m$\theta$1) > 0, assigning samples to class 1 or class 2 respectively.

**L-Softmax** loss by Takumi [34] introduced **Hard Margin** (another type of angular margin)

$$11)\quad \left||w_1|\right|\left||x|\right|\cos(\theta_1) \geq \left||w_1|\right|\left||x|\right|\cos(\alpha\theta_1) > \left||w_1|\right|\left||x|\right|\cos(\theta_2)$$

- It's formulated by an intermediate term to aid the classification of samples near the decision boundary.
- From a different perspective, the angle in L-SoftMax is α times smaller than the angle in the original SoftMax.

The computation of hard margin, which involves both backward and forward propagations, became more manageable with **Soft-Margin** SoftMax, where the angle equation is reformulated.

$$12)\quad w_1^T x \geq w_1^T x - m > w_2^T x$$

Here, m as a real positive number represents distance margin for sample x belongs to class 1

In **Additive Margin-SoftMax (AM-SoftMax)** by Feng Wang [35], hyperparameter margin is applied in additive way and it's more interpretable compared to Angular -SoftMax loss

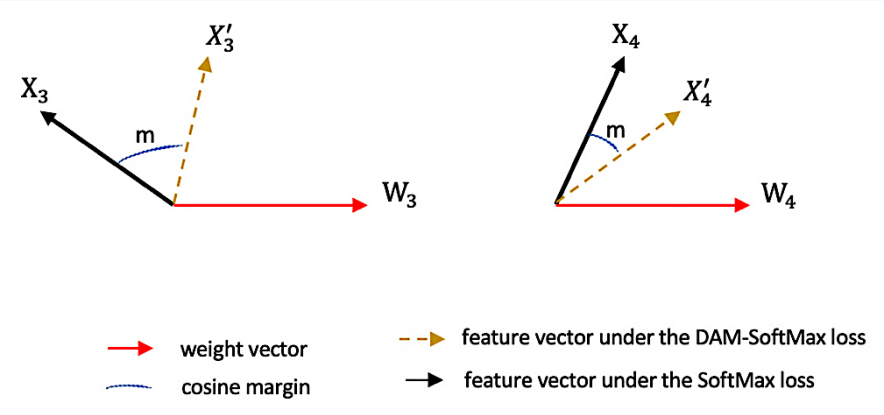
$$13)\quad \psi(\theta) = \cos\theta - m \quad \rightarrow \quad \text{where } x = \cos\theta \Rightarrow \psi(x) = x - m$$

$$14)\quad L_{AMS} = -\frac{1}{n}\sum_{i=1}^{n}\log\frac{e^{s.(w_{y_i}^T f_i - m)}}{e^{s.(w_{y_i}^T f_i - m)} + \sum_{j=1, j\neq y_i}^{c} e^{s w_j^T f_i}}$$

- Requires only forward propagation; though angular margin performs better; it is more computationally intensive.
- The boundary becomes marginal rather than a single vector.

**Note:** Introduced margin as a constant to enhance class separability does not account for the varying angles between sample features and their corresponding class centroids.

- DAM-SoftMax by D. Zhou [36] introduced a dynamic margin, penalizing smaller cosine values between a sample and its class centroid with a larger margin.




- In DAMS by S. Zhou [37], intra-class compactness and inter-class separation is amplified using double additive margin criteria and formulated as:

15) $\cos(\theta_1) - m < \cos(\theta_2) + m$ leading to $\cos(\theta_1) - \cos(\theta_2) < 2m$

16) $$L_{DAMS} = -\frac{1}{n}\sum_{i=1}^{n} \log \frac{e^{s.(w_{y_i}^T f_i - m)}}{e^{s.(w_{y_i}^T f_i - m)} + \sum_{j=1, j\neq y_i}^{c} e^{s.(w_j^T f_i + m)}}$$

# Methodology

## Model (EmoSynthNet)

**To create a robust and flexible classification CNN model for facial expression task, EmoSynthNet is influenced by feature proven beneficial models:**

- U-Net's structure for its excellent multi-scale feature mapping with limited data, using expansive path to enhance model’s performance

- VGG's backbone for extracting high-level features with small receptive fields
- Inception module for its skill in handling auxiliary classifiers as a regularizer

**Reusing feature maps is used instead of expanding the network size because it:**
- Reduces the risk of overfitting
- Optimizes computational resources
- Processes information across various scales

## MAIN BRANCH

**The main branch consists of two distinct segments: the analytical and synthetic segments.**

**Analytical Segment** (Feature Extractor and Latent Space Representation):

- Convolution with 3x3 kernels for computational efficiency
- Convolution with 1x1 kernels to enhance non-linearity
- Average pooling to maintain data trends while preserving residual information
- Activation functions to prevent vanishing gradients
- Batch normalization for consistent and stable training
- Dropout as regularization to prevent overfitting

**Synthetic Segment** (Abstract Representation Decompressor):

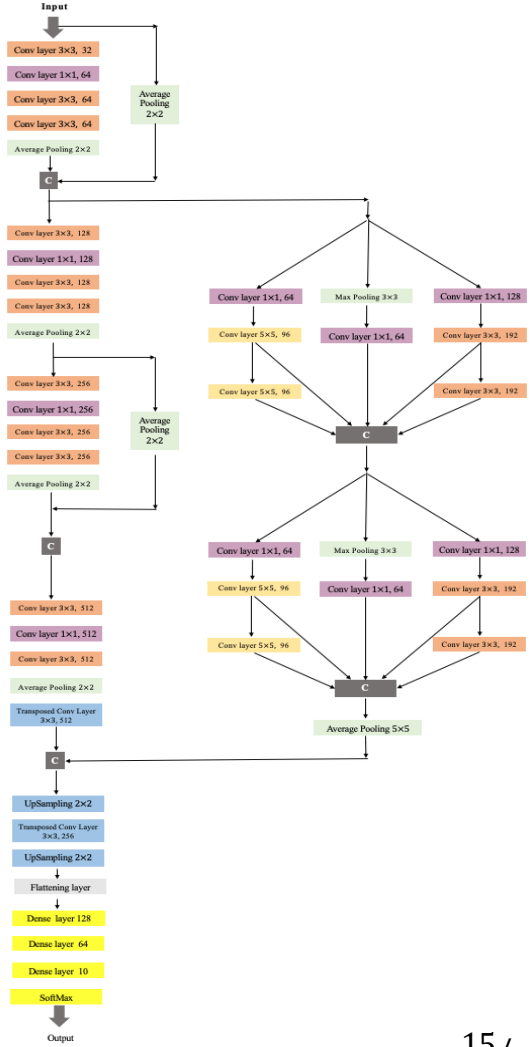
- Transposed convolution to discretize parameters
- Upsampling to enhance the clarity of feature maps

## SIDE BRANCH

**An auxiliary classifier is introduced as a regularizer to:**

- Improve gradient back-propagation
- Enhance early-stage feature discrimination
- Boost training stability

The distinction in designed auxiliary classifier with Inception module lies in the decision to minimize the width, extend the depth of each module and uses of short connections.



# Proposed SoftMax Loss

## LOSS

Symmetric Cross-Entropy (SCE) includes CE and modified version of focal loss (FL) [31]

17) $\mathbf{L_{SCE}} = \propto \mathbf{L_{CE}} + \boldsymbol{\beta}\mathbf{L_{Modified(FC)}}$

18) $\mathbf{L_{CE}}(p, y) = \begin{cases} -\log(p) & \text{if } y = 1 \\ -\log(1-p) & \text{otherwise} \end{cases}$

19) $\mathbf{L_{FC}} = -\frac{1}{N}\sum_{i=1}^{N} \alpha\,(1-y_p)^{\gamma}\log(y_p)$

$\alpha$: balanced variant $\approx [0,1]$ $\gamma$: modulating factor $\approx [0,5]$

$\propto$ and $\beta$ serve as balancing parameters to fine-tune the impact of each component

$\mathbf{L_{Modified(FC)}}$ uses terms as learnable hyperparameters to manage confidence errors and to penalize the relative loss in adaptable way for poor classification samples (hard examples)

20) $\text{alpha} = \frac{1}{N}\sum_{i=1}^{N}(y_t - y_p)^2$ 21) $\text{weight} = ||(1 - y_{true}) - y_{pred}||_2$

The aim of using **alpha** parameter is counter underlearning, and **weight** parameter efficiently modulate loss influence for confidently misclassified samples.

22) $L_{SCE} = -\frac{1}{N}\sum_{i=1}^{N} \text{alpha}\,(\text{weight}\,.\log(y_p))$

**Proposed Loss Function:**

- Effectively adjusts error impact based on classification certainty.
- Prioritizes learning on samples that require more attention.

**SCE Loss Algorithm**

Input ($y_{true}$ , $y_{pred}$)

$y_{predw} \leftarrow$ <$(1 - y_{true})$, $(1 - y_{pred})$: <> denotes multiplication

weight $\leftarrow$ $||(1 - y_{true})$ - $y_{predw}||_2$

$y_{pred} \leftarrow$ <$y_{true}$, $y_{pred}$>

alpha $\leftarrow \frac{1}{N}\sum_{i=1}^{N}(y_{true} - y_{pred})^2$

$y_p \leftarrow \sum \log_{10} y_p$ : (log normalization)

$L_{proposed} \leftarrow$ < alpha , (< weight, $y_p$ >) >

Return $L_{proposed}$

$Loss_{SCE} = \alpha L_{ce} + \beta L_{Proposed}$

Output $Loss_{SCE}$

# SoftMax

Standard Convolution Outputs Can be viewed as a Gaussian mixture distribution [38].

**Challenge**: Characterizing the model using density estimation with the maximum likelihood criterion

**Margin Equation:**

- Represent probability distributions across different classes for faster convergence.
- Introduces adaptable hyperparameter 'M' to allow variable margin distances for classes with varying densities.
- 'M' is adjusted based on the largest variation and mean in each class's data distribution.

$$23)\quad M_v \leftarrow \sum \text{ArgMax}\,(x) + \in$$

$$24)\quad M_c \leftarrow \text{Cliping}\,(0 < \log M_v \leq 4)$$

$$25)\quad M_U \leftarrow \text{Mean}\,(x) + M_c$$

$M_U$ represents the maximum variation applied to the mean of each distribution in a multi-class classification task. Therefore, for a sample belongs to class 1, the margin selection formula can be expressed as:

$$26)\quad W_1^T x \geq w_1^T x - m \gg w_1^T x - M_U \geq w_2^T x$$

**MAM-SoftMax algorithm**

Input (x)

$$M_V \leftarrow \sum_i \text{argmax}\,(x_i) + \text{Initialized value} \quad (\text{defult} \approx 0.2)$$

$$M_C \leftarrow 0 \leq \text{Clipping}(M_V) \leq 4$$

$$\exp M_U \leftarrow \sum \text{Mean}\,(x_i) + \log_{10} M_C$$

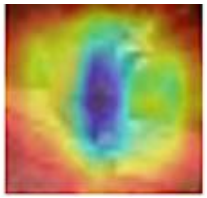
$$\exp(x - M_U) \leftarrow \exp(x) * \exp(-M_U)$$

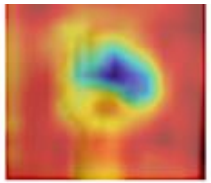
$$S_{MAM} \approx \text{Proposed SoftMax} \leftarrow -\log\left(\frac{e^{w_{y_i}^T x_i - M_U}}{e^{w_{y_i}^T x_i - M_U} + \sum_{j \neq i}^{k} e^{w_j^T x_j}}\right)$$

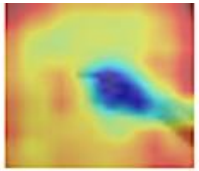
Return $S_{MAM}$

# RESULTS

**Model Evaluation:**



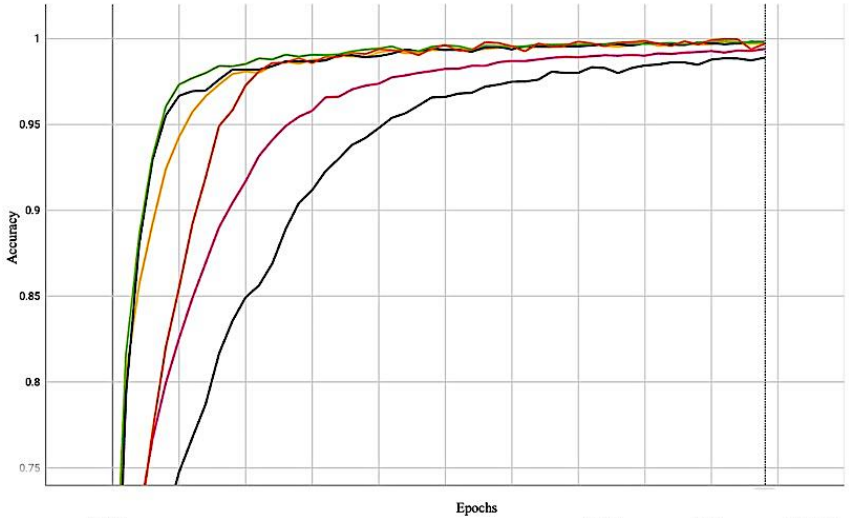


| Methods | Parameters (Millions) | FLOPS (Billion) | Approx. Size (in MB) | Depth |
|---|---|---|---|---|
| MobileNetV2 [39], | 4 | 0.006 | 14 | 105 |
| EfficientNetB0 [40], | 5 | 0.008 | 29 | 132 |
| ResNet50 [41], | 26 | 0.078 | 98 | 107 |
| VGG19 [42], | 144 | 0.417 | 549 | 19 |
| DensNet121 [16], | 8 | 0.057 | 33 | 242 |
| **EmoSynthNet,** | **21** | **1.120** | **82** | **34** |

**CIFAR10**

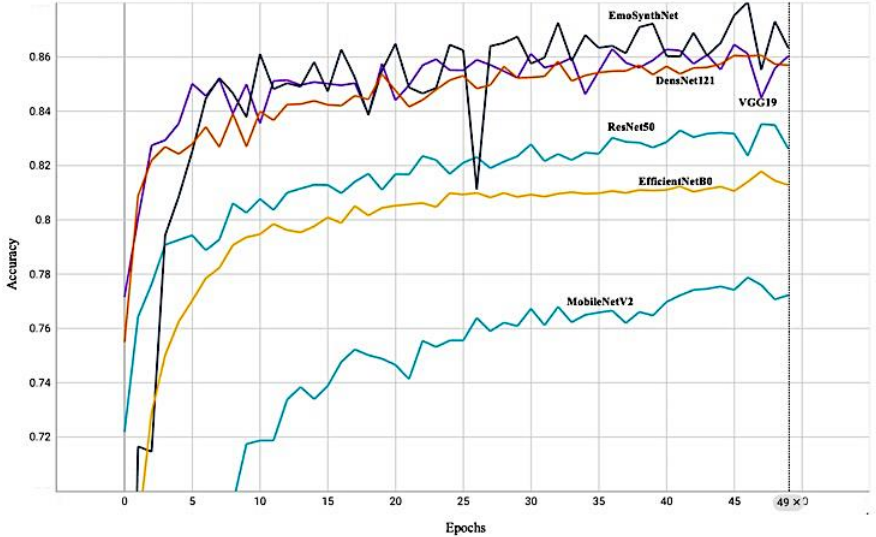
| Methods | Optimizer | Loss | Accuracy | Error | Precision | Recall | F1-Score |
|---|---|---|---|---|---|---|---|
| MobileNetV2, | Adam | CE | 77.23% | 1.20 | 78.30% | 76.65% | 0.77 |
| EfficientNetB0, | Adam | CE | 81.26% | 0.93 | 82.24% | 80.87% | 0.81 |
| ResNet50, | Adam | CE | 82.59% | 1.10 | 82.91% | 82.21% | 0.82 |
| DensNet121, | Adam | CE | 85.68% | 0.85 | 86.00% | 85.34% | 0.85 |
| VGG19, | Adam | CE | 86.03% | 1.05 | 86.35% | 85.78% | 0.86 |
| **EmoSynthNet,** | **Adam** | **CE** | **86.32%** | **0.67** | **87.03%** | **85.99%** | **0.86** |

## Train




| Run | Value | Step | Relative |
|---|---|---|---|
| DenseNet121/20240303-053151/train | 0.9977 | 49 | 35.2 min |
| EfficientNetB0/20240303-064455/train | 0.994 | 49 | 22.67 min |
| EmoSynthNet(ours)/20240303-224554/train | 0.997 | 49 | 2.169 hr |
| MobileNetV2/20240303-062040/train | 0.9889 | 49 | 17.52 min |
| ResNet50/20240303-043535/train | 0.998 | 49 | 22.32 min |
| VGG19/20240303-051625/train | 0.9971 | 49 | 10.87 min |

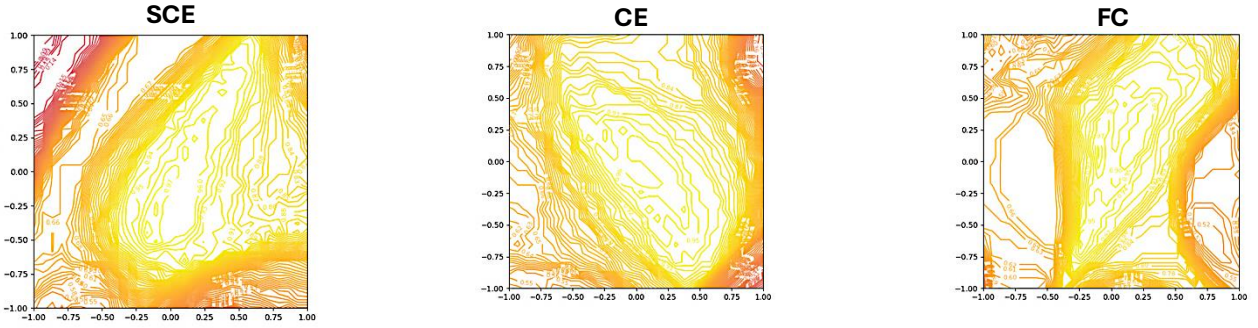
## Test



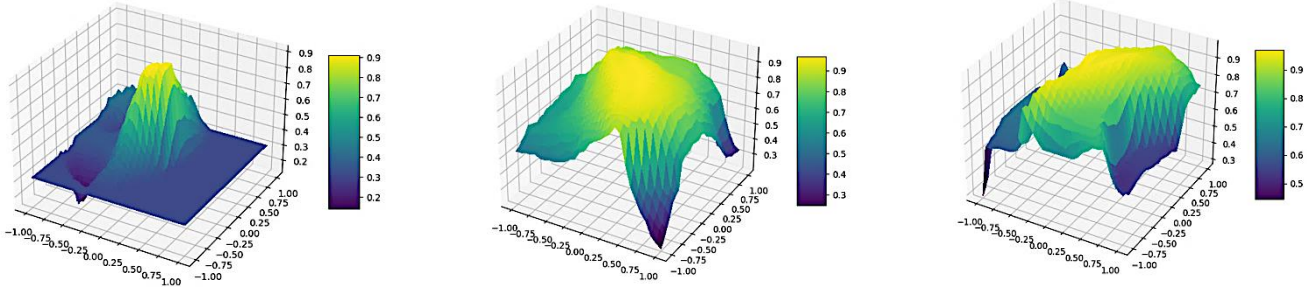

| Run | Value | Step | Relative |
|---|---|---|---|
| DenseNet121/20240303-053151/validation | 0.857 | 49 | 35.2 min |
| EfficientNetB0/20240303-064455/validation | 0.8128 | 49 | 22.67 min |
| EmoSynthNet(ours)/20240303-224554/validation | 0.8633 | 49 | 2.169 hr |
| MobileNetV2/20240303-062040/validation | 0.7722 | 49 | 17.52 min |
| ResNet50/20240303-043535/validation | 0.8261 | 49 | 22.32 min |
| VGG19/20240303-051625/validation | 0.8604 | 49 | 10.87 min |

# SoftMax & LOSS Evaluation

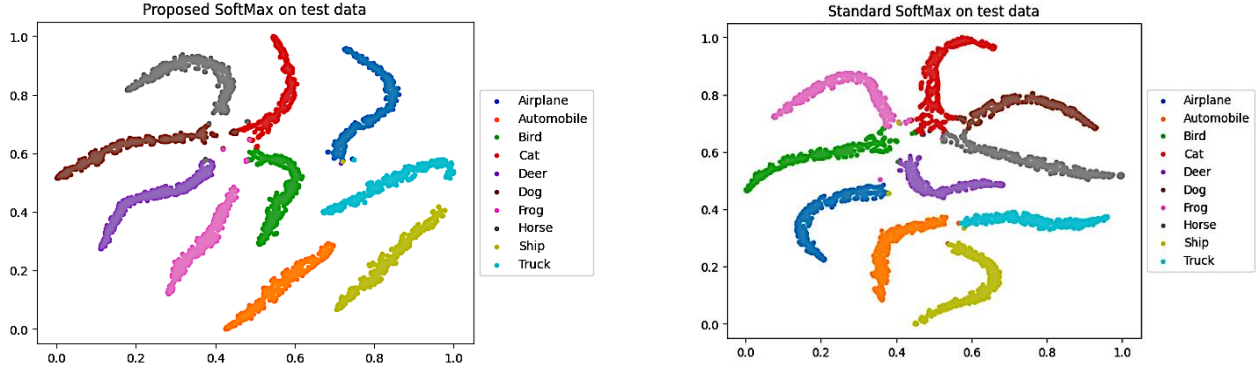
**Contour Loss**



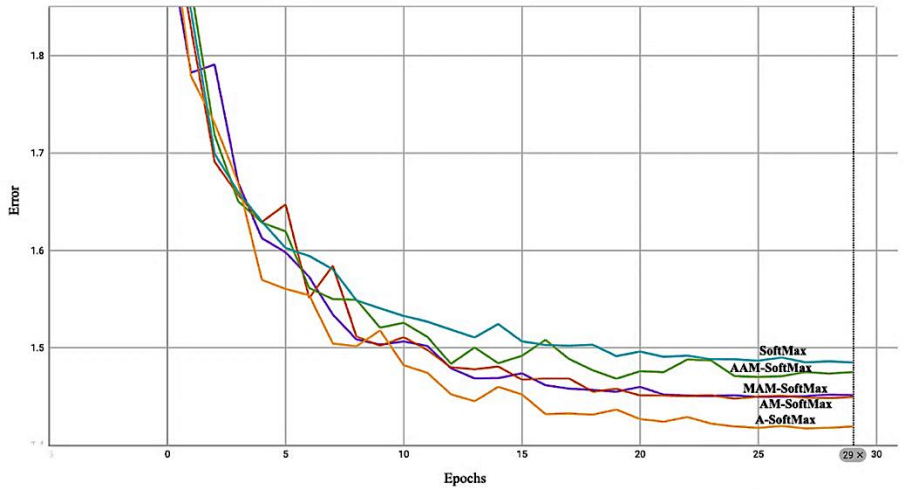

**Loss landscape**



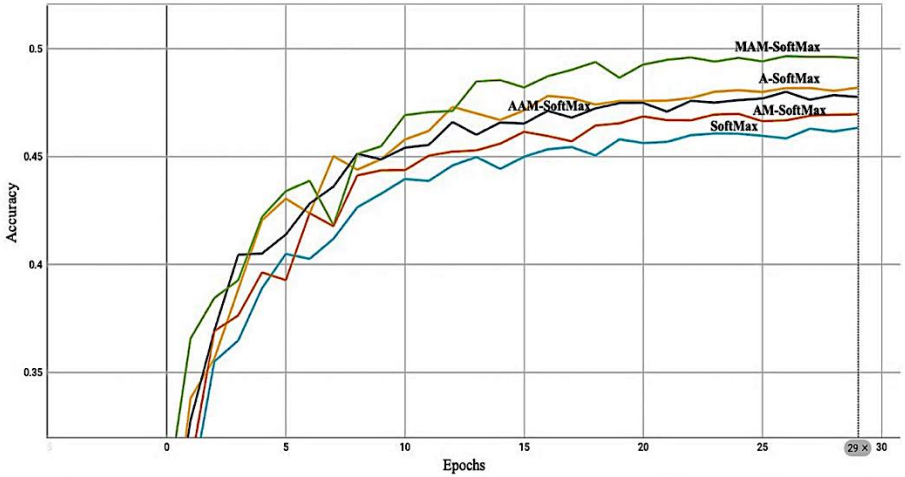
**t-SNE**



# SoftMax Evaluation Plots

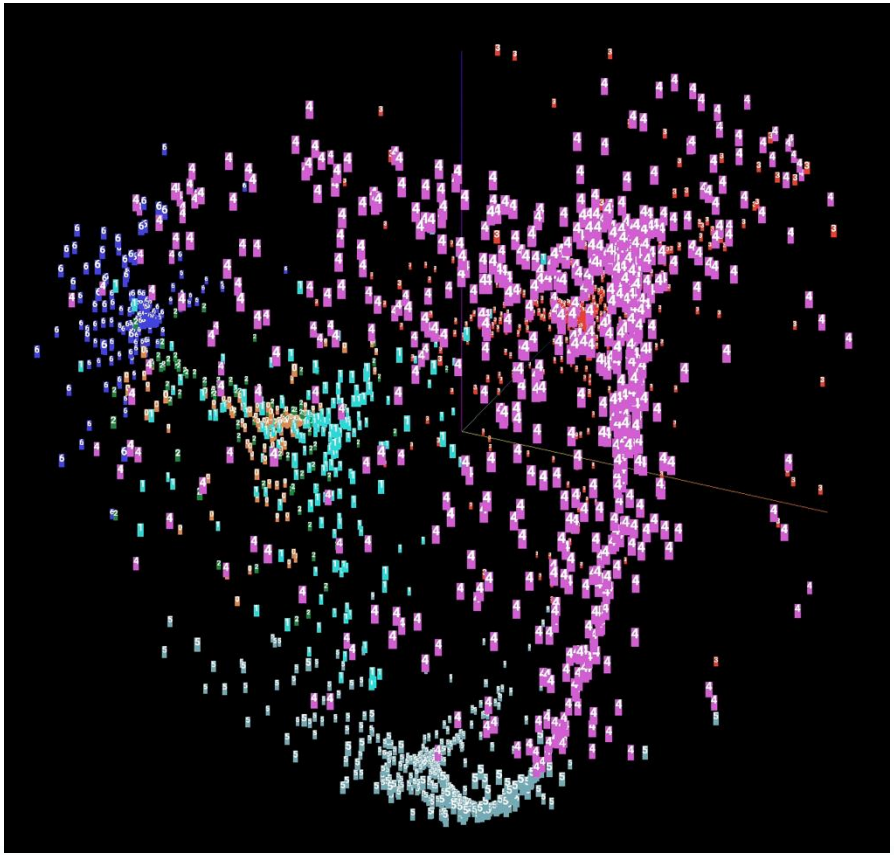



| Run | Value | Step | Relative |
|---|---|---|---|
| MAM-SoftMax(Ours)/20240319-064010/validation | 1.4515 | 29 | 15.11 min |
| Additive Margin SoftMax Loss/20240319-053701/validation | 1.4496 | 29 | 3.772 min |
| Angular Additive Margin SoftMax/20240319-055338/validation | 1.4752 | 29 | 3.809 min |
| Angular SoftMax/20240319-054704/validation | 1.4191 | 29 | 3.94 min |
| Conventional SoftMax Loss/20240319-054141/validation | 1.4849 | 29 | 3.828 min |

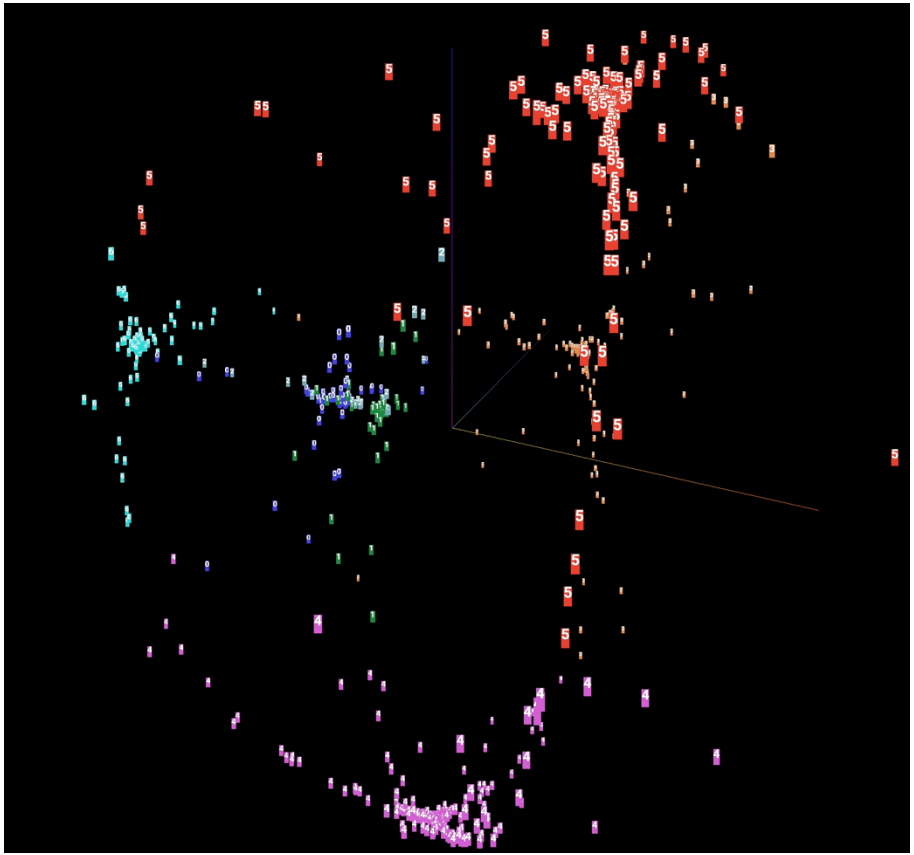



| Run | Value | Step | Relative |
|---|---|---|---|
| Additive Margin SoftMax Loss/20240319-053701/validation | 0.4696 | 29 | 3.772 min |
| Angular Additive Margin SoftMax/20240319-055830/validation | 0.4776 | 29 | 3.782 min |
| Angular SoftMax/20240319-054704/validation | 0.4819 | 29 | 3.94 min |
| Conventional SoftMax Loss/20240319-054141/validation | 0.4633 | 29 | 3.828 min |
| MAM-SoftMax(Ours)/20240319-050007/validation | 0.4956 | 29 | 15.59 min |

# 3D Learned Feature Visualization

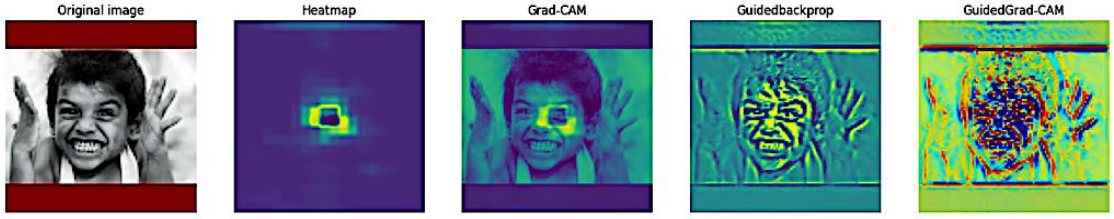


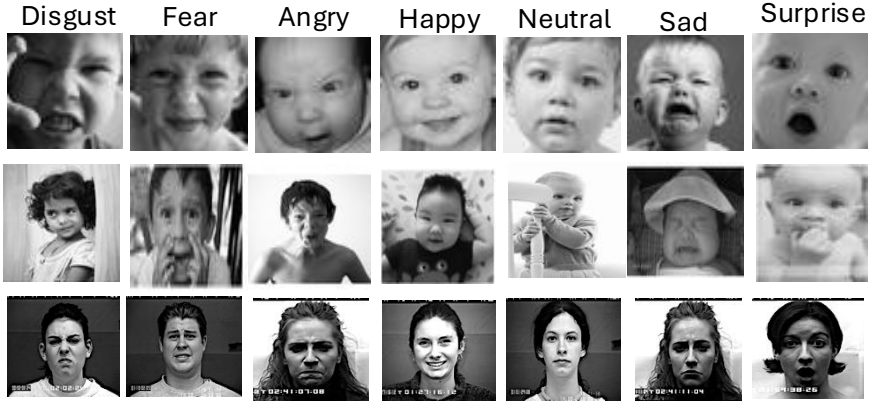
Standard SoftMax



MAM SoftMax

# Framework Evaluation on Facial Expression Recognition




**FER-2013**

| | Model | Accuracy (%) |
|---|---|---|
| 1 | **EmoSynthNet,** | **93.79** |
| 2 | PAtt-Lite [43], | 92.50 |
| 3 | Ensemble ResmaskingNet with 6 other CNNs [44], | 76.82 |
| 4 | EmoNetXt [45], | 76.12 |
| 5 | Segmentation VGG-19 [46], | 75.97 |
| 6 | Local Learning Deep+BOW [2], | 75.42 |
| 7 | LHC-Net [47], | 74.42 |
| 8 | LHC-NetC [47], | 74.28 |
| 9 | Residual Masking Network [44], | 74.14 |
| 10 | ResNet18 With Tricks [48], | 73.70 |

**RAF-DB**

| | Model | Accuracy (%) |
|---|---|---|
| 1 | **EmoSynthNet** | **91.94** |
| 2 | DDAMFN [49], | 91.35 |
| 3 | ViT_base + MAE [50], | 91.07 |
| 4 | TransFER [51], | 90.91 |
| 5 | EAC [8], | 90.35 |
| 6 | DAN [52], | 89.70 |
| 7 | RUL (ResNet-18) [3], | 88.98 |
| 8 | PSR (VGG-16) [53], | 88.98 |
| 9 | MViT [54], | 88.62 |
| 10 | EfficientFace [55], | 88.36 |

**CK+**

| | Model | Accuracy (%) |
|---|---|---|
| 1 | **EmoSynthNet** | **100.00** |
| 2 | PAtt-Lite [43], | 100.00 |
| 3 | ViT +SE [58], | 99.8 |
| 4 | FAN [10], | 99.7 |
| 5 | FDRL [59], | 99.54 |
| 6 | FN2EN [10], | 98.6 |
| 7 | ST network [60], | 98.50 |
| 8 | Nonlinear eval on SL+SSL Puzzling (B0) [61], | 98.23 |
| 9 | DeepEmotion [62], | 98 |
| 10 | IF-GAN [63], | 97.52 |

# Conclusion

### Introduced the DCNN network in the first part

- Model designed to process data effectively by focusing on key features.
- Structure includes an analysis path for latent space generation and a synthesis path to improve data handling and localization.
- Performance improved with an auxiliary classifier that processes images across different feature maps to select useful feature vectors.

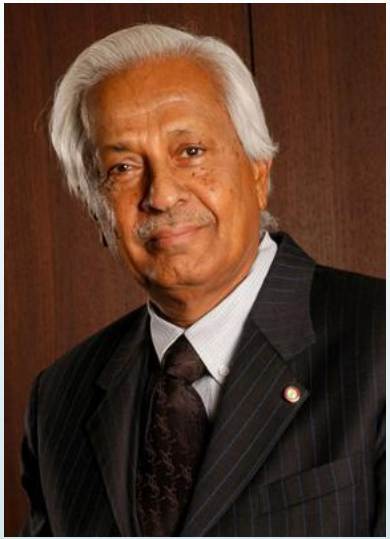
### Introduced the SoftMax Loss in the second part

- The proposed SoftMax loss aims to overcome the standard SoftMax loss function's limitations, which under learns complex classes and overfits simpler ones.
- Its loss function uses regularization, while the SoftMax function employs an adaptive soft-margin.
- The enhancement maximizes separation between dissimilar features and brings similar ones closer together.

### Evaluated the framework for FER domain in the third part

- Once of the primary goals of Facial Expression Recognition (FER) framework should detect distinctive features with limited labeled samples.
- Deep Metric Learning (DML) methods enhance FER system performance by distinguishing features between classes and maintaining consistency within classes.
- These modifications improve training stability and ensure consistent model performance.
- This work aims to develop a learning methodology in facial expression learning that emphasizes distinct features, enabling better generalization.

# Acknowledgment of Professors' Support in Thesis Completion



**Dr. M.N.S Swamy**

**Dr. M. Omair Ahmad**

**Dr. William E. Lynch**

# References:


[1] "Papers with Code - FER2013 Benchmark (Facial Expression Recognition)." Accessed: Feb. 13, 2023. [Online]. Available: https://paperswithcode.com/sota/facial-expression-recognition-on-fer2013

[2] M.-I. Georgescu, R. T. Ionescu, and M. Popescu, "Local Learning with Deep and Handcrafted Features for Facial Expression Recognition," IEEE Access, vol. 7, pp. 64827–64836, 2019, doi: 10.1109/ACCESS.2019.2917266.

[3] Y. Zhang, C. Wang, and W. Deng, "Relative Uncertainty Learning for Facial Expression Recognition," in Advances in Neural Information Processing Systems, Curran Associates, Inc., 2021, pp. 17616–17627. Accessed: Nov. 13, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/9332c513ef44b682e9347822c2e457ac-Abstract.html

[4] H. Wang, H. Huang, Y. Hu, M. Anderson, P. Rollins, and F. Makedon, "Emotion detection via discriminative kernel method," in Proceedings of the 3rd International Conference on PErvasive Technologies Related to Assistive Environments, in PETRA '10. New York, NY, USA: Association for Computing Machinery, Jun. 2010, pp. 1–7. doi: 10.1145/1839294.1839303.

[5] N. Jain, S. Kumar, A. Kumar, P. Shamsolmoali, and M. Zareapoor, "Hybrid deep neural networks for face emotion recognition," Pattern Recognit. Lett., vol. 115, pp. 101–106, Nov. 2018, doi: 10.1016/j.patrec.2018.04.010.

[6] A. P. Fard and M. H. Mahoor, "Ad-Corre: Adaptive Correlation-Based Loss for Facial Expression Recognition in the Wild," IEEE Access, vol. 10, pp. 26756–26768, 2022, doi: 10.1109/ACCESS.2022.3156598.

[7] "Facial Expression Recognition in the Wild via Deep Attentive Center Loss | IEEE Conference Publication | IEEE Xplore." Accessed: Nov. 13, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9423267

[8] Y. Zhang, C. Wang, X. Ling, and W. Deng, "Learn From All: Erasing Attention Consistency for Noisy Label Facial Expression Recognition." arXiv, Sep. 20, 2022. doi: 10.48550/arXiv.2207.10299.

[9] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random Erasing Data Augmentation," in The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, AAAI Press, 2020, pp. 13001–13008. doi: 10.1609/AAAI.V34I07.7000.

[10] H. Ding, S. K. Zhou, and R. Chellappa, "FaceNet2ExpNet: Regularizing a Deep Face Recognition Net for Expression Recognition," in 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), May 2017, pp. 118–126. doi: 10.1109/FG.2017.23.

[11] A. H. Farzaneh and X. Qi, "Facial Expression Recognition in the Wild via Deep Attentive Center Loss," in 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Jan. 2021, pp. 2401–2410. doi: 10.1109/WACV48630.2021.00245.

[12] "Pyramid With Super Resolution for In-the-Wild Facial Expression Recognition | IEEE Journals & Magazine | IEEE Xplore." Accessed: Apr. 07, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/9143068

[13] "Classifying emotions and engagement in online learning based on a single facial expression recognition neural network | IEEE Journals & Magazine | IEEE Xplore." Accessed: Nov. 14, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9815154

[14] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, Y. Bengio and Y. LeCun, Eds., 2015. Accessed: May 12, 2024.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[16] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Jul. 2017, pp. 2261–2269. doi: 10.1109/CVPR.2017.243.

[17] "[PDF] Network In Network | Semantic Scholar." Accessed: May 12, 2024. [Online]. Available: https://www.semanticscholar.org/paper/Network-In-Network-Lin-Chen/5e83ab70d0cbc003471e87ec306d27d9c80ecb16.

[18] C. Szegedy et al., "Going deeper with convolutions," presented at the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Jun. 2015, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

[19] N. S. Punn and S. Agarwal, “Inception U-Net Architecture for Semantic Segmentation to Identify Nuclei in Microscopy Cell Images,” ACM Trans. Multimed. Comput. Commun. Appl., vol. 16, no. 1, p. 12:1-12:15, Feb. 2020, doi: 10.1145/3376922.
[20] “U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Dec. 19, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9446143
[21] “(4) (PDF) A Multigrid Tutorial, 2nd Edition.” Accessed: Dec. 19, 2022. [Online]. Available: https://www.researchgate.net/publication/220690328_A_Multigrid_Tutorial_2nd_Edition
[22] R. Szeliski, “Fast surface interpolation using hierarchical basis functions,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 12, no. 6, pp. 513–528, Jun. 1990, doi: 10.1109/34.56188.
[23] A. Nawaz, U. Akram, A. A. Salam, A. R. Ali, A. Ur Rehman, and J. Zeb, “VGG-UNET for Brain Tumor Segmentation and Ensemble Model for Survival Prediction,” in 2021 International Conference on Robotics and Automation in Industry (ICRAI), Oct. 2021, pp. 1–6. doi: 10.1109/ICRAI54018.2021.9651367.
[24] “The Real-World-Weight Cross-Entropy Loss Function: Modeling the Costs of Mislabeling | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Jan. 31, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/8943952
[25] R. Machlev, Y. Levron, and Y. Beck, “Modified Cross-Entropy Method for Classification of Events in NILM Systems,” IEEE Trans. Smart Grid, vol. 10, no. 5, pp. 4962–4973, Sep. 2019, doi: 10.1109/TSG.2018.2871620.
[26] B. Shan and Y. Fang, “A Cross Entropy Based Deep Neural Network Model for Road Extraction from Satellite Images,” Entropy, vol. 22, p. 535, May 2020, doi: 10.3390/e22050535.
[27] P. Li et al., “An improved categorical cross entropy for remote sensing image classification based on noisy labels,” Expert Syst. Appl., vol. 205, p. 117296, Nov. 2022, doi: 10.1016/j.eswa.2022.117296.
[28] X. Ma, H. Huang, Y. Wang, S. R. S. Erfani, and J. Bailey, “Normalized loss functions for deep learning with noisy labels,” in Proceedings of the 37th International Conference on Machine Learning, in ICML’20, vol. 119. JMLR.org, Jul. 2020, pp. 6543–6553.
[29] “Dynamically Weighted Balanced Loss: Class Imbalanced Learning and Confidence Calibration of Deep Neural Networks | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Jan. 31, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9324926
[30] P. Goyal, “Shallow SegNet with bilinear interpolation and weighted cross-entropy loss for Semantic segmentation of brain tissue,” in 2022 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), Mar. 2022, pp. 361–365. doi: 10.1109/SPICES52834.2022.9774193.
[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal Loss for Dense Object Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.
[32] “Large-margin softmax loss for convolutional neural networks | Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48.” Accessed: Apr. 06, 2024. [Online]. Available: https://dl.acm.org/doi/10.5555/3045390.3045445
[33] “Angular Margin Softmax Loss and Its Variants for Double Compressed AMR Audio Detection | Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security.” Accessed: Feb. 01, 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/3437880.3460414 .
[34] T. Kobayashi, “Large Margin In Softmax Cross-Entropy Loss,” presented at the British Machine Vision Conference, 2019. Accessed: Feb. 01, 2023. [Online]. Available: https://www.semanticscholar.org/paper/Large-Margin-In-Softmax-Cross-Entropy-Loss-Kobayashi/0c88506b5e6091906e656dfeb9ee1060946d4aab
[35] “Additive Margin Softmax for Face Verification | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Feb. 01, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/8331118
[36] D. Zhou et al., “Dynamic Margin Softmax Loss for Speaker Verification,” in Interspeech 2020, ISCA, Oct. 2020, pp. 3800–3804. doi: 10.21437/Interspeech.2020-1106.
[37] S. Zhou, C. Chen, G. Han, and X. Hou, “Double Additive Margin Softmax Loss for Face Recognition,” Appl. Sci., vol. 10, p. 60, Dec. 2019, doi: 10.3390/app10010060.
[38] “[PDF] Gaussian Mixture Convolution Networks | Semantic Scholar.” Accessed: Apr. 23, 2024. [Online]. Available: https://www.semanticscholar.org/paper/Gaussian-Mixture-Convolution-Networks-Celarek-Hermosilla/ce108373d3458c2f27524483221aaad772c3edf3
[39] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” presented at the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Jun. 2018, pp. 4510–4520. doi: 10.1109/CVPR.2018.00474.
[40] “[PDF] EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks | Semantic Scholar.” Accessed: May 12, 2024. [Online]. Available: https://www.semanticscholar.org/paper/EfficientNet%3A-Rethinking-Model-Scaling-for-Neural-Tan-Le/4f2eda8077dc7a69bb2b4e0a1a086cf054adb3f9
[41] P. J. P. and A. Sethi, WaveMix: Resource-efficient Token Mixing for Images. 2022.

[42] S. Verma, A. Chug, and A. P. Singh, “Revisiting activation functions: empirical evaluation for image understanding and classification,” Multimed. Tools Appl., Jul. 2023, doi: 10.1007/s11042-023-16159-2.
[43] “PAtt-Lite: Lightweight Patch and Attention MobileNet for Challenging Facial Expression Recognition | Request PDF.” Accessed: May 12, 2024. [Online]. Available: https://www.researchgate.net/publication/371684300_PAtt-Lite_Lightweight_Patch_and_Attention_MobileNet_for_Challenging_Facial_Expression_Recognition.
[44] I. J. Goodfellow et al., “Challenges in Representation Learning: A Report on Three Machine Learning Contests,” in Neural Information Processing, M. Lee, A. Hirose, Z.-G. Hou, and R. M. Kil, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2013, pp. 117–124. doi: 10.1007/978-3-642-42051-1_16.
[45] “EmoNeXt: an Adapted ConvNeXt for Facial Emotion Recognition | IEEE Conference Publication | IEEE Xplore.” Accessed: Apr. 07, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10337732
[46] S. Vignesh, M. Savithadevi, M. Sridevi, and R. Sridhar, “A novel facial emotion recognition model using segmentation VGG-19 architecture,” Int. J. Inf. Technol., vol. 15, no. 4, pp. 1777–1787, Apr. 2023, doi: 10.1007/s41870-023-01184-z.
[47] “[PDF] Local Multi-Head Channel Self-Attention for Facial Expression Recognition | Semantic Scholar.” Accessed: May 12, 2024. [Online]. Available: https://www.semanticscholar.org/paper/Local-Multi-Head-Channel-Self-Attention-for-Facial-Pecoraro-Basile/cde1045b1e9f3b3939ae54df6ae8a2e3079b6d15.
[48] “Fer2013 Recognition - ResNet18 With Tricks | Papers With Code.” Accessed: Aug. 14, 2023. [Online]. Available: https://paperswithcode.com/paper/fer2013-recognition-resnet18-with-tricks
[49] “A Dual-Direction Attention Mixed Feature Network for Facial Expression Recognition. - Document - Gale Academic OneFile.” Accessed: May 12, 2024. [Online]. Available: https://go.gale.com/ps/i.do?id=GALE%7CA764265120&sid=sitemap&v=2.1&it=r&p=AONE&sw=w&userGroupName=anon%7Ebe10a316&aty=open-web-entry
[50] J. Li, J. Nie, D. Guo, R. Hong, and M. Wang, “Emotion Separation and Recognition from a Facial Expression by Generating the Poker Face with Vision Transformers.” arXiv, Jun. 09, 2023. doi: 10.48550/arXiv.2207.11081.
[51] F. Xue, Q. Wang, and G. Guo, “TransFER: Learning Relation-aware Facial Expression Representations with Transformers,” 2021 IEEECVF Int. Conf. Comput. Vis. ICCV, pp. 3581–3590, Oct. 2021, doi: 10.1109/ICCV48922.2021.00358.
[52] Z. Wen, W. Lin, T. Wang, and G. Xu, “Distract Your Attention: Multi-Head Cross Attention Network for Facial Expression Recognition,” Biomim. Basel Switz., vol. 8, no. 2, p. 199, May 2023, doi: 10.3390/biomimetics8020199.
[53] H. Vo, G.-S. Lee, H.-J. Yang, and S. H. Kim, “Pyramid with Super Resolution for In-The-Wild Facial Expression Recognition,” IEEE Access, vol. PP, pp. 1–1, Jul. 2020, doi: 10.1109/ACCESS.2020.3010018.
[54] “MViT: Mask Vision Transformer for Facial Expression Recognition in the wild.” Accessed: Apr. 08, 2024. [Online]. Available: https://www.researchgate.net/publication/352244399_MViT_Mask_Vision_Transformer_for_Facial_Expression_Recognition_in_the_wild
[55] Z. Zhao, Q. Liu, and F. Zhou, “Robust Lightweight Facial Expression Recognition Network with Label Distribution Training,” Proc. AAAI Conf. Artif. Intell., vol. 35, no. 4, Art. no. 4, May 2021, doi: 10.1609/aaai.v35i4.16465.
[56] K. Wang, X. Peng, J. Yang, D. Meng, and Y. Qiao, “Region Attention Networks for Pose and Occlusion Robust Facial Expression Recognition,” IEEE Trans. Image Process., vol. 29, pp. 4057–4069, 2020, doi: 10.1109/TIP.2019.2956143.
[57] “[PDF] Occlusion Aware Facial Expression Recognition Using CNN With Attention Mechanism | Semantic Scholar.” Accessed: Apr. 08, 2024. [Online]. Available: https://www.semanticscholar.org/paper/Occlusion-Aware-Facial-Expression-Recognition-Using-Li-Zeng/b5bb7e12a15b57b4d307e742da127a74596d0c7c
[58] K. Kpalma, “Learning Vision Transformer with Squeeze and Excitation for Facial Expression Recognition,” arXiv (Cornell University), Jul. 2021, Accessed: May 12, 2024. [Online]. Available: https://www.academia.edu/108303243/Learning_Vision_Transformer_with_Squeeze_and_Excitation_for_Facial_Expression_Recognition.

[59] D. Ruan, Y. Yan, S. Lai, Z. Chai, C. Shen, and H. Wang, “Feature Decomposition and Reconstruction Learning for Effective Facial Expression Recognition,” 2021 IEEECVF Conf. Comput. Vis. Pattern Recognit. CVPR, pp. 7656–7665, Jun. 2021, doi: 10.1109/CVPR46437.2021.00757.
[60] “Facial Expression Recognition Based on Deep Evolutional Spatial-Temporal Networks - PubMed.” Accessed: Apr. 08, 2024. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28371777/
[61] “Using Self-Supervised Auxiliary Tasks to Improve Fine-Grained Facial Representation | Semantic Scholar.” Accessed: Aug. 10, 2023. [Online]. Available: https://www.semanticscholar.org/paper/Using-Self-Supervised-Auxiliary-Tasks-to-Improve-Pourmirzaei-Esmaili/8c34613a9c123246802e079fa0feb0709f090087
[62] “Spatial-Temporal Recurrent Neural Network for Emotion Recognition - PubMed.” Accessed: Oct. 18, 2023. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29994572/
[63] “(PDF) Identity-Free Facial Expression Recognition Using Conditional Generative Adversarial Network.” Accessed: Apr. 08, 2024. [Online]. Available: https://www.researchgate.net/publication/351736453_Identity-Free_Facial_Expression_Recognition_Using_Conditional_Generative_Adversarial_Network

Thank you