Proceedings of the International Conference on
Formal Methods and Foundations of Artificial Intelligence
Eszterházy Károly Catholic University
Eger, Hungary, June 5–7, 2025
pp. 102–114
DOI: 10.17048/fmfai.2025.102

Web-based facial expression recognition using hybrid deep learning

Ming Hu, Gergely Kovásznai

Eszterházy Károly Catholic University
mitntghu@gmail.com
kovasznai.gergely@uni-eszterhazy.com

Abstract. This paper present a hybrid ResNet+FPN+Transformer architecture for facial expression recognition, achieving 80.90% accuracy on FER-2013 with a browser-based implementation using TensorFlow.js for client-side inference.

We compare four model configurations: ResNet50 baseline, ResNet+FPN, ResNet+Transformer, and our full ResNet+FPN+Transformer model. Our hybrid architecture combines ResNet backbone features with Feature Pyramid Networks and transformer components to process facial features at multiple scales simultaneously. Our ResNet+FPN+Transformer model achieves 80.90% mean accuracy on FER-2013 (averaged over 5 independent training runs with different random initializations). Ablation studies confirm both FPN (+2.35%) and Transformer (+2.77%) components improve performance over the ResNet50 baseline (77.69%).

Our web application features interactive visualization tools revealing the network's decision-making process, including feature map animations and 3D neural network visualization. This browser-based implementation uses TensorFlow.js for client-side inference.

Keywords: facial expression recognition, deep learning, ResNet, transformer, feature pyramid networks, web application

AMS Subject Classification: 68T45, 68T10

1. Introduction

Facial expression recognition (FER) is crucial in human-computer interaction, emotion analysis, and various other fields. Despite advances in deep learning for facial expression recognition, challenges persist in model interpretability, accessible de-

ployment, and real-world variability.

Our main contributions include: (1) a hybrid ResNet+FPN+Transformer architecture achieving 80.90% accuracy on FER-2013 with ablation studies validating component contributions, (2) a comprehensive web application with real-time analysis and 3D network visualization, and (3) advanced training techniques addressing class imbalance challenges.

2. Related work

Facial expression recognition has evolved from traditional computer vision approaches to deep learning-based methods.

The introduction of CNNs has dramatically improved FER performance. ResNet [2] addressed the vanishing gradient problem through residual connections, while Feature Pyramid Networks [3] enhanced multi-scale feature representation. Vision Transformers [1], originally designed for NLP, have been adapted for computer vision tasks, excelling in capturing long-range dependencies.

Recent hybrid architectures combine CNNs with transformers, leveraging the strengths of both approaches. Most CNN-Transformer hybrids lack component-level ablation studies and interactive visualization tools for deployment.

3. Methodology

3.1. Hybrid architecture design

Our ResNet+FPN+Transformer model (Figure 1) integrates three complementary components to address key challenges in facial expression recognition:

ResNet Feature Extractor: A modified ResNet50 backbone extracts multi-scale features from layers C2-C5, providing robust local feature extraction with gradient flow preservation. This creates a feature pyramid capturing patterns from fine-grained details (wrinkles, texture) to high-level facial structures.

Feature Pyramid Network: FPN enables multi-scale information fusion by combining high-resolution, spatially precise features with low-resolution, semantically rich features through lateral connections. This addresses the challenge that facial expressions manifest at different spatial scales.

Transformer Encoder: A transformer encoder [6] with learnable CLS tokens captures global spatial relationships through self-attention mechanisms, modeling long-range dependencies between facial regions (e.g., eye-mouth coordination in surprise expressions).

Integration Strategy. The three components operate hierarchically: ResNet extracts local features across multiple scales, FPN fuses these multi-scale representations, and the Transformer processes the C5-level features to incorporate global

context. Specifically, C5 outputs (7×7×2048) are projected to 256 channels via 1×1 convolutions, then processed by the transformer encoder before final classification. This design leverages CNNs' locality strength, FPN's multi-scale fusion, and Transformers' global modeling in a unified framework.

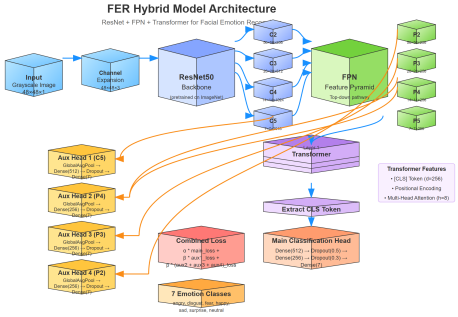

Figure 1. Architecture of the proposed ResNet-FPN-Transformer model for facial expression recognition.

3.2. Design rationale

Facial expressions manifest at multiple spatial scales and require modeling long-range dependencies between facial regions. FPN addresses the multi-scale challenge through lateral connections combining high-resolution spatial details with low-resolution semantic features. The transformer encoder captures global dependencies through self-attention, enabling the model to jointly consider coordinated facial movements (e.g., eye-mouth relationships in surprise) rather than treating regions independently.

3.3. Training techniques

To train our model, we apply the following techniques.

Focal Loss: To address severe class imbalance in FER-2013 (disgust: 1.8% vs happy: 29.3%), we implement focal loss [4] which dynamically adjusts the contribution of examples based on classification difficulty.

Mixup Training: We apply mixup data augmentation [7] to synthesize new training samples through linear interpolation, creating smooth decision boundaries and improving generalization.

Advanced Augmentation: Our pipeline includes random noise injection, occlusion simulation, and motion blur to enhance model robustness.

4. Implementation details

4.1. ResNet feature extraction pipeline

The ResNet feature extractor builds upon a pre-trained ResNet50 backbone, extracting multi-scale representations from intermediate layers. The implementation creates a multi-output model accessing specific layer outputs:

- C2 output: conv2_block3_out (56×56 resolution)
- C3 output: conv3_block4_out (28×28 resolution)
- C4 output: conv4_block6_out (14×14 resolution)
- C5 output: conv5_block3_out (7×7 resolution)

To ensure compatibility with the transformer encoder, a projection layer standardizes the C5 feature dimensions to 256 channels using 1×1 convolutions. The grayscale input images are replicated across three channels to match the pre-trained ResNet50 input requirements.

4.2. Transformer encoder configuration

The transformer encoder processes the projected C5 features with the following architecture:

- **Model dimension**: 256 channels
- **Attention heads**: 8 multi-head attention mechanisms
- **Encoder layers**: 3 stacked transformer layers
- **Feedforward dimension**: 1024 neurons
- **Dropout rate**: 0.1 for regularization

The implementation includes learnable CLS tokens initialized with random normal distribution (stddev=0.02). Positional encoding preserves spatial relationships that would otherwise be lost in the self-attention mechanism.

4.3. Training configuration

The hybrid model employs several advanced training techniques to address dataset challenges:

- **Optimizer and Learning Rate**: We use AdamW optimizer with initial learning rate of 1×10^{-4} and weight decay of 1×10^{-4}. The learning rate follows a cosine annealing schedule with warm restarts. All models train for 100 epochs with batch size 128.

- **Focal Loss Implementation**: To handle the severe class imbalance, focal loss with α=0.25 and γ=2.0 dynamically adjusts example contributions based on classification difficulty.

- **Mixup Data Augmentation**: Linear interpolation between training pairs creates synthetic examples using Beta distribution sampling (α=0.2, application probability=0.5), improving generalization and creating smoother decision boundaries.

- **Multi-Head Training**: The primary classification head receives a weight of 0.7, with auxiliary heads sharing the remaining 0.3 to enable multi-scale supervision.

5. Experiments and results

5.1. Experimental setup

We evaluate our approach on the FER-2013 dataset containing 35,887 grayscale images across seven emotion categories. To assess result stability, each model configuration was trained five times with different random initializations (no fixed seeds). Training was conducted on NVIDIA Tesla V100 with 32GB VRAM, requiring approximately 42 minutes per run for the hybrid model.

5.2. Performance comparison and ablation study

Figure 2 presents the performance comparison across four model configurations. Each model was trained five times with different random initializations to assess stability.

As shown in Figure 2, the proposed ResNet+FPN+Transformer architecture achieves 80.90% accuracy, demonstrating substantial improvements over the baseline ResNet50 (77.69%). Both the FPN and Transformer components provide significant contributions: FPN adds 2.35 percentage points through multi-scale feature fusion, while the Transformer contributes 2.77 points through global context modeling.

Component Comparison. Direct comparison between ResNet+FPN (80.04%) and ResNet+Transformer (80.46%) reveals that the Transformer provides a slightly higher individual improvement (+0.42%). This suggests that while both components are valuable, global context modeling has a marginally stronger impact than multi-scale features for overall accuracy. However, the similar magnitude of improvements (2.35% vs 2.77%) indicates that both components address important but complementary aspects of the problem.

Synergistic Effect. Our full hybrid architecture achieves 3.21 percentage points improvement over the baseline. While the combined improvement is less than the theoretical sum of individual components (2.35% + 2.77% = 5.12%),

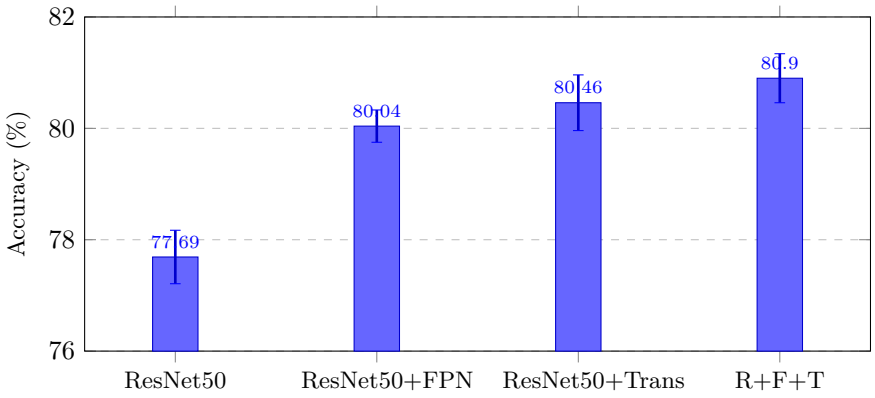

Figure 2. Ablation study results on FER-2013 dataset. Mean accuracy over 5 independent runs with error bars showing standard deviation ($\pm 1\sigma$).

this reflects the natural interaction between components where some features overlap. The integration successfully leverages the complementary strengths of both FPN and Transformer, as demonstrated by the superior per-class performance across all emotion categories. The low standard deviations across all configurations (0.29%–0.50%) indicate stable training across different initializations.

5.3. Per-class performance analysis

To understand how different architectural components affect recognition across emotion categories, we analyze per-class performance for all model variants in Figure 3.

Figure 3 reveals distinct performance patterns across emotion categories, demonstrating how different architectural components contribute to recognition of specific expressions.

ResNet+FPN Performance. The FPN component demonstrates balanced improvements across most categories, particularly excelling at fear (47.31%), surprise (86.04%), and neutral (90.89%). This suggests that multi-scale feature extraction effectively captures subtle facial details crucial for these expressions, such as the fine-grained texture patterns around the eyes in fear and the overall facial relaxation characteristic of neutral expressions.

ResNet+Transformer Performance. The Transformer component shows exceptional performance on angry expressions (78.15%), achieving an 8.61% improvement over the FPN-based model. This indicates that global context modeling is particularly beneficial for expressions characterized by complex spatial relationships, where the coordination between multiple facial regions (furrowed brows, tightened lips, and tensed jaw) must be jointly considered. However, the Transformer shows reduced performance on fear (38.71%, -8.60% vs FPN), suggesting

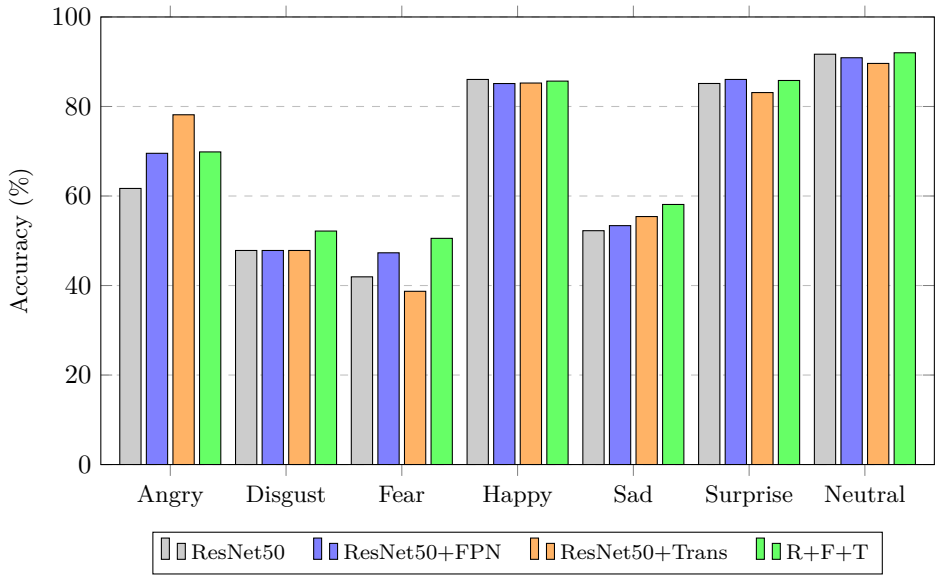

Figure 3. Per-class accuracy comparison across different architectures on FER-2013 dataset. Results averaged over 5 independent runs.

that purely global features may miss fine-grained local details critical for this emotion.

Complementary Strengths. Direct per-class comparison reveals that FPN and Transformer excel at different emotion categories. FPN outperforms Transformer on fear (+8.60%), surprise (+2.93%), and neutral (+1.27%), while Transformer excels on angry (+8.61%) and sad (+2.03%).

Full Model Advantages. Our complete ResNet+FPN+Transformer architecture successfully integrates these complementary strengths. The full model achieves best overall performance on neutral (92.00%) and fear (50.54%), demonstrates improved robustness on underrepresented classes like disgust (52.17% vs 47.83% baseline), and maintains more balanced performance across all emotion categories. Notably, while neither FPN nor Transformer alone improves disgust recognition, their combination yields a 4.34 percentage point improvement, suggesting that the integration enables the model to better handle challenging minority classes.

The confusion matrix in Figure 4 reveals the classification patterns of our model. Fear and surprise expressions show some confusion due to similar visual characteristics such as widened eyes. Sad and neutral expressions also demonstrate moderate confusion, while happy expressions show the least confusion with other categories.

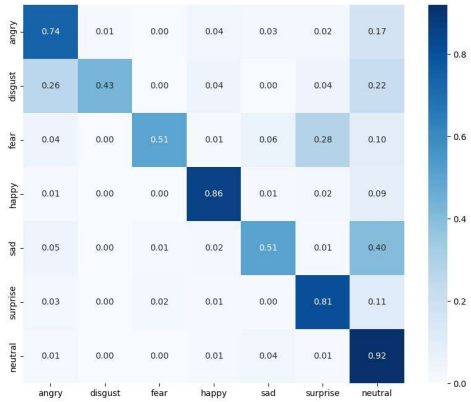

Figure 4. Confusion matrix for ResNet-FPN-Transformer model showing classification patterns.

5.4. Discussion

Dataset limitations

The FER-2013 dataset, while widely used as a benchmark, presents several inherent limitations that constrain the interpretation of our results:

Low Resolution and Grayscale: The 48×48 grayscale images limit the model's ability to capture fine-grained facial details and color-based cues that may be relevant for expression recognition in higher-quality images.

Cultural Bias: FER-2013 predominantly contains Western facial expressions, potentially limiting generalization to cross-cultural contexts where expression interpretation may differ.

Label Noise: The crowdsourced annotation process introduces label inconsistencies, as subjective interpretation of subtle expressions varies across annotators.

Performance positioning

While some recent approaches report higher accuracies on FER-2013 through specialized techniques such as ensemble methods, extensive data augmentation, or larger model architectures, our work prioritizes practical deployment considerations. Our 80.90% accuracy demonstrates effective integration of multi-scale features and global context modeling, while maintaining advantages in privacy (client-side inference), interpretability (3D visualization), and accessibility (browser-based deployment without specialized hardware).

6. Web application

We developed a web application that provides an integrated platform for facial expression recognition with interactive visualization capabilities.

6.1. Client-side architecture

Our web application implements a fully client-side architecture using TensorFlow.js, ensuring privacy by processing facial data entirely in the browser. The system comprises three main modes:

Image Analysis: Users upload images for static expression prediction with confidence visualization and feature map analysis.

Real-Time Recognition: WebRTC-based camera access enables continuous facial expression analysis with frame processing control for responsive performance.

3D Network Visualization: An interactive Three.js-based visualization allows users to explore the neural network architecture, with nodes representing different layer types and connections showing data flow.

6.2. Web deployment architecture

6.2.1. TensorFlow.js model conversion

The trained model undergoes conversion to TensorFlow.js format [5] for browser deployment. The conversion process includes weight quantization, layer optimization, and format adaptation to web-compatible tensor operations.

6.2.2. Client-side inference pipeline

The browser-based inference implements efficient preprocessing and prediction. The preprocessing pipeline includes image resizing to 48×48 pixels using bilinear interpolation, RGB to BGR channel conversion for ResNet compatibility, ResNet mean normalization ([103.939, 116.779, 123.68]), and batch dimension expansion for model input.

Memory Management: The system implements tensor disposal and parallel execution for efficient real-time processing.

Performance Optimization: Real-time processing uses parallel execution for prediction and feature map extraction through `Promise.all()`, maintaining responsive user interaction while processing facial expressions.

6.2.3. WebRTC integration

Camera access utilizes WebRTC APIs for cross-browser compatibility. The implementation includes frame rate control to balance processing load with visual responsiveness. Error handling manages camera access permissions and device compatibility issues.

6.3. Interactive visualization features

6.3.1. 3D neural network rendering

Figure 5 demonstrates the 3D network visualization interface. Users can rotate, zoom, and interact with the network structure to understand the data flow and component relationships. Different geometric shapes represent various layer types, with color coding indicating layer functions and activation levels.

The Three.js-based visualization[1] creates interactive representations of the neural network architecture:

Geometric Layer Mapping: Different layer types receive distinct visual representations:

- Convolutional layers: Cube geometries with dimensions reflecting kernel sizes
- Pooling layers: Pyramid shapes indicating dimensionality reduction
- Dense layers: Sphere geometries scaled by neuron count
- Transformer layers: Octahedron shapes distinguishing attention mechanisms

Spatial Layout Algorithm: Node positioning implements hierarchical arrangements based on network depth. The algorithm calculates appropriate spacing to maintain visual clarity while preserving logical data flow relationships.

Interactive Controls: User interaction includes mouse/touch rotation and zooming controls, click events revealing detailed layer information, and animation controls demonstrating forward pass data flow.

6.3.2. Feature map visualization system

The feature map visualization in Figure 6 provides insights into model decision-making by displaying activation patterns across network layers. The system shows how early layers capture edges and textures, while deeper layers focus on emotion-specific abstractions. The animated progression helps users understand the hierarchical feature learning process.

The feature map extraction system processes intermediate network activations. The system extracts activations from multiple network layers during inference and selects representative channels (4 channels per layer) for visualization.

[1] R. Cabello: Three.js – JavaScript 3D library, https://threejs.org/

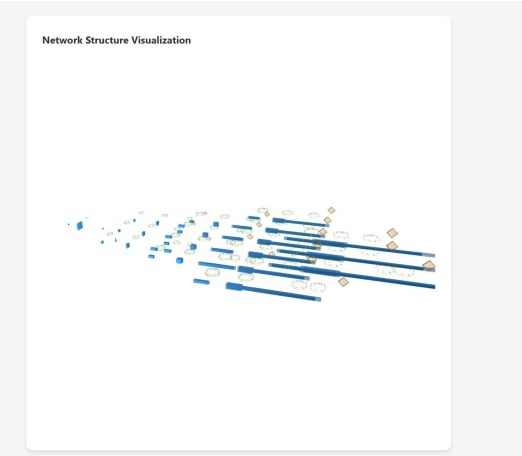

Figure 5. Interactive 3D visualization.

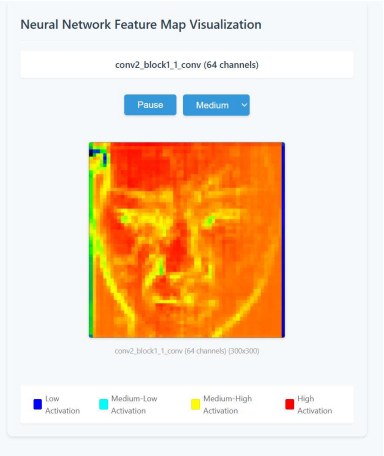

Figure 6. Feature maps.

Visualization Rendering. The system displays feature maps as animated heatmaps, showing the progression from edge detection in early layers to emotion-specific abstractions in deeper networks. Color-coded intensity maps reveal which facial regions activate different network components.

Real-time Animation. Feature map updates synchronize with inference operations, providing immediate visual feedback about network decision-making processes. The animation sequence demonstrates how facial features propagate through the network hierarchy.

6.4. Performance optimization

The complete system implements several optimization techniques:

Adaptive Rendering: Visualization quality adjusts based on device capabilities, maintaining smooth interaction across different hardware configurations.

Lazy Loading: Components initialize only when needed, reducing initial application load times and memory usage.

Efficient Resource Management: WebGL contexts and Three.js objects undergo proper cleanup to prevent resource leaks during extended usage sessions.

6.5. Ethical considerations

Privacy Protection. Our fully client-side architecture processes all facial data locally in the user's browser without server transmission, providing inherent privacy

advantages over cloud-based systems. No facial images or extracted features leave the user's device.

Fairness and Bias. Facial analysis systems may exhibit performance disparities across demographic groups. Future work should evaluate our model's fairness across age, gender, and ethnicity to ensure equitable performance.

Appropriate Use. We emphasize that emotion recognition technology should complement rather than replace human judgment, particularly in sensitive applications such as mental health assessment or surveillance contexts.

7. Conclusion

We presented a hybrid architecture combining multi-scale features and global context modeling for facial expression recognition, with client-side deployment and interactive visualizations. Future work should address cross-cultural validation and temporal modeling for video analysis.

While our system demonstrates several innovations, we acknowledge key limitations: (1) evaluation on a single, imbalanced dataset with inherent quality constraints, (2) performance gap (approximately 4-5 percentage points) compared to state-of-the-art methods, and (3) the need for further validation on diverse, real-world data to assess practical robustness across demographic groups and environmental conditions.

Key innovations include the hybrid architecture design, comprehensive training methodology addressing class imbalance, and interactive visualizations bridging the gap between technical AI implementations and human understanding. The 3D network visualization and feature map animations provide valuable educational insights into neural network behavior.

Future work includes extending to cross-cultural expression analysis, incorporating temporal information for video sequences, and developing mobile-optimized versions through model compression techniques.

References

[1] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby: *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*, in: International Conference on Learning Representations (ICLR), Published at ICLR 2021; originally posted to arXiv in 2020, 2021, DOI: 10.48550/arXiv.2010.11929.

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

[3] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie: *Feature Pyramid Networks for Object Detection*, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936–944, DOI: 10.1109/CVPR.2017.106.

[4] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár: *Focal Loss for Dense Object Detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence 42.2 (2020), DOI registered in 2018, published in 2020, pp. 318–327, DOI: 10.1109/TPAMI.2018.2858826.

[5] D. Smilkov, N. Thorat, Y. Assogba, A. Yuan, N. Kreeger, P. Yu, K. Zhang, S. Cai, E. Nielsen, D. Soergel, S. Bileschi, M. Terry, C. Nicholson, S. N. Gupta, S. Sirajuddin, D. Sculley, R. Monga, G. Corrado, F. B. Viégas, M. Wattenberg: *TensorFlow.js: Machine Learning for the Web and Beyond*, in: Proceedings of the 2nd SysML Conference, 2019, 2019, DOI: 10.48550/arXiv.1901.05350.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin: *Attention is All You Need*, in: Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 5998–6008, DOI: 10.48550/arXiv.1706.03762.

[7] H. Zhang, M. Cisse, Y. N. Dauphin, D. Lopez-Paz: *mixup: Beyond Empirical Risk Minimization*, in: International Conference on Learning Representations (ICLR), Originally posted to arXiv in 2017, 2018, DOI: 10.48550/arXiv.1710.09412.