Script

SOTA Models (1)

* EmoNeXt built on Facebook’s ConvNeXt architecture
* The STN module aligns faces by correcting for rotation, scale, and position.
  + It predicts how to adjust the image (**localisation network**)
  + Builds a map for the change (**grid generator**)
  + Applies it (**sampler**) — helping the model focus on key facial areas.
* SE blocks help the model focus on important features by learning which channels matter most.
  + After down sampling, they compress each feature map (**squeeze**)
  + Estimate feature importance with a small network (**excitation**)
  + Rescale the maps accordingly (**scale**)
* Self-attention regularisation helps the model focus more evenly across features.
  + Works by reducing how much the attention weights differ from their average — promoting balance.
  + The attention itself compares parts of the input using similarity scores.
  + The final loss combines the usual cross-entropy loss with this regularisation to improve generalisation

SOTA Model (2)

* ResEmoteNet built on a traditional convolutional neural network backbone enhanced with residual connections to address the vanishing gradient problem. Helps the model maintain performance.
* The model starts with three convolutional layers that extract features step by step — like edges, textures, and shapes. Each layer is followed by batch normalization to stabilize learning, and max pooling to reduce image size and make the model more robust.
* Next, the SE block helps the model focus on the most important feature channels. It does this by compressing each feature map, learning which ones matter most, and then reweighting them to boost key signals.
* Then we use three residual blocks, which include shortcut connections. These help the model train more effectively by avoiding problems like vanishing gradients in deeper networks.
* After the residual layers, we apply Adaptive Average Pooling. Unlike regular pooling, it gives us a fixed size output no matter the input size, which is useful for handling different image dimensions.
* Finally, the pooled features are passed to a classifier, which predicts the emotion shown in the image.

SOTA Model (3)

* Baseline model to compare other models to since its not created with FER in mind.
* Utilises residual connections similar to ResEmoteNet.
* ResNet50 has 50 layers, striking a balance between depth, accuracy, and computational efficiency, widely used architectures for generic image recognition tasks.
* Architecture Explained:
  + Starts with a large 7×7 convolutional layer and a max pooling layer, reducing spatial dimensions of the input while retaining key information.
  + Residual blocks consist of three layers:
    - 1×1 convolution for reducing dimensions
    - 3×3 convolution for capturing spatial features
    - 1×1 convolution to restore the original channel size.
  + Instead of relying on pooling layers for down sampling, ResNet50 uses a stride of 2 within the convolutional layers, keeping the model efficient while maintaining important features.
  + Finally a global average pooling layer aggregates spatial information, leading into the final fully connected layer that outputs the class prediction.
* ResNet50’s design offers high representational power with manageable computational cost, making it ideal for large-scale image classification problems, and a popular backbone for many modern vision tasks.

SOTA Model (4)

* DDAMFN consists of two major components:
* Mixed Feature Network (MFN)
  + Extracts foundational features
  + Adapted from MobileFaceNet
  + Composed of Non-Residual & Residual Blocks
  + Different kernel sizes within each block to capture features at multiple scales.
  + PReLU Activation: Chosen over ReLU to better extract features.
  + Network Depth Tuning: Adjusted to avoid overfitting.
  + Coordinate Attention: Focuses the model on important spatial locations in the feature maps.
* Dual-Direction Attention Network (DDAN)
  + Refines those features using attention mechanisms to focus on key facial regions.
  + Horizontal & vertical attention maps (1 from each DDA).
  + Replaces average pooling with a **linear GDConv layer** to assign different importance to spatial positions.
  + Combines attention maps element-wise and selects the most informative one to highlight key facial regions.
  + Uses an **attention loss** (Mean Squared Error between attention heads) to ensure each head focuses on different areas.
  + The output feature map (7×7×512) is processed by a linear GDConv layer and reshaped into a 512-dimensional vector.
  + A fully connected layer then predicts the final class (emotion).

SOTA Model (5)

* PAtt-Lite starts with a lightweight version of MobileNetV1 (up till block 9). This allows it to extract useful features from images while keeping the model small and efficient.
* It uses a patch extraction block made of three convolutional layers.
  + The first two are depthwise separable convolutions that divide the feature map into four patches and learn higher-level features.
  + The third is a pointwise convolution to refine those features. This setup boosts accuracy and keeps the model compact
* To avoid overfitting, the model uses Global Average Pooling. GAP reduces each feature map to a single average value, cutting down parameters and helping generalisation.
* Finally, an attention classifier improves focus. It includes a self-attention layer (via dot product) placed between two fully connected layers. This helps the model highlight the most important parts of the image for better classification.
* Overall, PAtt-Lite is designed to be efficient yet effective, with a small parameter count and enhanced performance on challenging facial emotion recognition subsets.

Model Results (1)

* SOTA Model
  + DDAMFN++
    - Overall accuracy outperformed other models
    - Strong cross-domain generalization (82.8% on CK+)
* JAFFE performance
  + Poor performance overall
    - Limited generalisation for datasets with narrow demographics
    - Importance of training data diversity
    - Raises ethical concerns about deploying these models in diverse real-world settings.
    - Models performed significantly better on CK+

Model Results (2)

* Efficiency and Practicality
  + ResNet-based models (e.g., ResNet50, ResEmoteNet):
    - Trained faster (under 2.5 hours) while still performing well.
    - Example: ResNet50 reached 83.3% in just 1 hour on RAF-DB.
  + DDAMFN++ is more accurate but takes 4+ hours to train.
  + RAF-DB consistently led to shorter training times and higher accuracy, likely due to well-balanced class distribution.
* Reproducibility and Consistency
  + Compared results with Papers With Code benchmarks:
    - Notable performance drops seen in PAtt-Lite and DDAMFN++ on FER2013 vs FER+.
    - Suggests sensitivity to label variations and training splits.
  + EmoNeXt and ResEmoteNet also underperformed relative to published benchmarks.
  + DDAMFN++ showed better alignment with reported results, especially outside FER2013.
  + Highlights the importance of consistent labels and careful evaluation practices.

Ethical Consideration

No need for script