Kathmandu University

Department of Artificial Intelligence

Panchkhal, Kavre

Deep Learning System for Motor Fault Classification

[Code No. AICC 401]

Submitted By:

Rakshya Bhusal (3)

Melina Singh(28)

Submitted To:

Mr. Subodh Acharya

Submission Date:

March 31, 2025

Panchkhal, Kavre

Deep Learning System for Motor Fault Classification

Introduction

This project focuses on developing a deep learning-based classification system for diagnosing industrial motor faults using vibration signals. The four fault types considered are Normal (N), Bearing Fault (BF), Misalignment (MA), and Rotor Imbalance (RI). The system is built using the Machinery Fault Database (MFPT) dataset, where raw time-series vibration signals are converted into spectrograms and processed through a hybrid ResNet-50 and EfficientNet-B0 model. Despite data limitations, the final model achieved an accuracy of 81% across varying noise levels (SNRs: 0, 10, 20 dB). The system is designed for real-time inference, with predictions taking less than one second on an NVIDIA GTX 1650 GPU.

Data Acquisition and Preparation

Objective

To gather and prepare vibration signals representing the four motor fault classes for deep learning-based classification.

Process

- **Dataset**: The MFPT dataset provides 26 raw vibration signals across multiple test rig conditions. These include three Normal (N) signals and 23 Bearing Fault (BF) signals from inner and outer race faults.

- **Simulation of Missing Fault Classes**:

    - **Misalignment (MA)**: Introduced a 1 Hz sinusoidal perturbation to Normal signals.

    - **Rotor Imbalance (RI)**: Introduced a 25 Hz sinusoidal perturbation to Normal signals.

    - **Additional Bearing Fault Data**: Augmented by adding a 100 Hz perturbation to Normal signals.

- **Augmentation**: Each signal was repeated five times to expand the dataset to approximately 130 signals per class.

- **Final Dataset Composition**: Four classes (N, BF, MA, RI) with approximately 130 signals each, ensuring balanced representation for training.

Rationale

Since the MFPT dataset lacks MA and RI signals, simulated perturbations were introduced to approximate real-world conditions. The dataset expansion mitigates class imbalance and improves model generalization.

Data Preprocessing

Objective

To convert raw 1D time-series vibration signals into 2D spectrograms for convolutional neural network (CNN) training.

Process

- **Resampling**: Standardized all signals to 10 kHz with 10,000 samples for consistency.

- **Noise Augmentation**: Applied Additive White Gaussian Noise (AWGN) at SNR levels of 0, 10, and 20 dB to simulate real-world noisy environments.

- **Spectrogram Generation**: Used Short-Time Fourier Transform (STFT) with a segment size of 256 to extract time-frequency features.

- **Resizing and Normalization**:

  - Converted spectrograms to 224x224 pixels using `tf.image.resize`.

  - Standardized values to zero mean and unit variance.

  - Converted single-channel spectrograms to RGB format for compatibility with pre-trained models.

- **Final Output**: An array of shape (390, 224, 224, 3) with corresponding class labels.

Rationale

Spectrograms effectively capture fault signatures, and noise augmentation improves robustness under real-world conditions. The resizing step ensures compatibility with deep learning models.

Model Architecture

Objective

To design a hybrid model leveraging ResNet-50 and EfficientNet-B0 for feature extraction and classification.

Process

- **Feature Extraction**:
  - ResNet-50 (pre-trained on ImageNet) extracts deep spatial features.
  - EfficientNet-B0 refines these features and improves classification.
- **Key Components**:
  - A 1x1 convolution layer aligns feature dimensions.
  - A transposed convolution layer upscales features for EfficientNet-B0 processing.
  - Global Average Pooling and a dense layer (256 units) enhance feature representation.
  - A dropout layer (0.5) prevents overfitting.
  - The final softmax layer outputs probabilities for the four fault classes.

Rationale

Combining ResNet-50's deep feature extraction with EfficientNet-B0's efficiency enables accurate classification while keeping computational complexity manageable.

Training Process

Objective

To train the model for high classification accuracy across all fault types.

Process

- **Data Splitting**: 80% training (312 samples), 10% validation (39 samples), 10% test (39 samples).
- **Augmentation**: Applied random rotations, zooming, and shifts to enhance data diversity.
- **Optimization**:
  - Used Adam optimizer with a learning rate of 0.001.
  - Sparse categorical cross-entropy loss function.
  - Class weights adjusted to counteract dataset imbalance.
- **Training Setup**:
  - 30 epochs with early stopping based on validation loss.
  - Trained on an NVIDIA GTX 1650 GPU (~1–2 hours).

Rationale

Augmentation, class weighting, and early stopping prevent overfitting and ensure robust learning despite the small dataset size.

Evaluation and Results

Objective

To assess the model's performance and validate its ability to classify motor faults.

Process

- **Metrics**: Accuracy, precision, recall, and F1-score.
- **Visualizations**:
    - Accuracy and loss curves over training epochs.
    - Confusion matrix to highlight classification performance.
    - Example spectrogram predictions.
- **Final Accuracy**: **81% test accuracy**, indicating the model's ability to distinguish faults under noisy conditions.
- **Inference Speed**: Achieved real-time classification with an average inference time of **0.25 seconds** per signal.

Rationale

Despite dataset limitations, achieving 81% accuracy under noise augmentation validates the model's effectiveness. Real-time inference ensures practical industrial applicability.

Challenges and Limitations

1. **Limited Real Data**: The reliance on simulated MA and RI data may not fully capture real-world fault variations.
2. **Upsampling Artifacts**: The transition from 7x7 to 224x224 resolution may introduce distortions.
3. **Class Imbalance**: The dominance of BF signals required class balancing to improve model generalization.
4. **Model Complexity**: The hybrid architecture increases computational demand, which could be optimized further.

Conclusion and Future Work

The deep learning system successfully classifies industrial motor faults using a hybrid CNN model. While achieving **81% accuracy**, certain limitations, such as simulated data and upsampling artifacts, impacted performance. Future work will explore:

- **Richer synthetic data generation** using physics-based simulation.
- **Parallel feature extraction** with separate CNN branches for time and frequency domains.
- **Lighter architectures** to improve efficiency without sacrificing accuracy.

This study demonstrates the potential of deep learning in predictive maintenance, paving the way for more robust industrial fault detection systems.