ISSN : 2456-3307 OPEN ACCESS
Available Online at : www.ijsrcseit.com
doi : https://doi.org/10.32628/CSEIT25113386

Energy-Reduced Bio-Inspired 1D-CNN for Audio Emotion Recognition

Jiby Mariya Jose,
Independent Researcher,
India

ARTICLEINFO

Article History:

Accepted : 15 June 2025
Published: 20 June 2025

Publication Issue
Volume 11, Issue 3
May-June-2025

Page Number
1034-1054

ABSTRACT

This paper proposes EPyNet, a deep learning architecture designed for energy reduced audio emotion recognition.In the domain of audio based emotion recognition, where discerning emotional cues from audio input is crucial, the inte- gration of artificial intelligence techniques has sparked a transformative shift in accuracy and performance.Deep learn- ing, renowned for its ability to decipher intricate patterns, spearheads this evolution. However, the energy efficiency of deep learning models, particularly in resource-constrained environments, remains a pressing concern. Convolutional operations serve as the cornerstone of deep learning systems. However, their extensive computational demands leading to energy-inefficient computations render them as not ideal for deployment in scenarios with limited resources. Ad- dressing these challenges, researchers came up with one-dimensional convolutional neural network (1D CNN) array convolutions, offering an alternative to traditional two-dimensional CNNs, with reduced resource requirements. How- ever, this array-based operation reduced the resource requirement, but the energy-consumption impact was not studied. To bridge this gap, we introduce EPyNet, a deep learning architecture crafted for energy efficiency with a particular emphasis on neuron reduction. Focusing on the task of audio emotion recognition, We evaluate EPyNet on five pub- lic audio corpora—RAVDESS, TESS, EMO DB, CREMA D, and SAVEE.We propose three versions of EPyNet, a lightweight neural network designed for efficient emotion recognition, each optimized for different trade-offs between accuracy and energy efficiency. Experimental results demonstrated that the 0.06M EPyNet reduced energy consumed by 76.5% while improving accuracy by 5% on RAVDESS, 25% on TESS, and 9.75% on SAVEE. The 0.2M and 0.9M models reduced energy consumed by 64.9% and 70.3%, respectively. Additionally, we compared our Proposed 0.06M system with the MobileNet models on the CIFAR-10 dataset and achieved significant improvements. The

proposed system reduces energy by 86.2% and memory by 95.7% compared to MobileNet, with a slightly lower accuracy of 0.8%. Compared to MobileNetV2, it improves accuracy by 99.2% and reduces memory by 93.8%. When compared to MobileNetV3, it achieves 57.2% energy reduction, 85.1% memory reduction, and a 24.9% accuracy improvement. We further test the scalability and robustness of the proposed solution on different data dimensions and frameworks.

Keywords: Computational Efficiency, Energy Reduction, audio Emotion Detection, Lightweight Convolutional Neural Network (CNN), Artificial Intelligence on Edge, audio Databases, Hierarchical Framework.

Introduction

Audio Emotion Recognition (AER), also referred to as affective audio analysis or affective computing, is the process of automatically detecting and interpreting emotions expressed in audio signals [1], [2]. In the digi- tal age, where human-computer interactions are rapidly increasing, the ability of machines to understand human audio emotions is becoming increasingly important [3]. Humans, as lifelong learners, constantly absorb and interpret information through audio-visual data from their environment. Communication, both verbal and non-verbal, allows individuals to convey ideas, inten- tions, and experiences. The human body, an intricate system, responds to these cues, providing emotional context through tone, pitch, and rhythm in speech [4]. Unlike visual signals, which are often constrained by the line of sight, audio signals can traverse barriers, making them a unique and valuable medium for conveying emotions [5].

AER has wide-ranging applications in fields like affec- tive computing [6], social robotics [7], virtual assistants [8], psychology [9], and human-computer interaction [10], where it helps machines better understand and respond to human emotions. However, the widespread adoption of these technologies, which rely on electricity to compute[11], has also led to an increase in solutions that consume significant energy, raising concerns about their environmental impact. As much of the planet's energy still comes from non-renewable sources, this growing demand contributes to higher carbon dioxide (CO2) emissions, exacerbating global warming and the greenhouse effect [12], [13].

In this paper, we focus on Convolu- tional Neural Networks (CNNs), which are a prominent category of deep learning models with significant po- tential in various disciplines, particularly in audio-based emotion detection [14]. CNNs excel in extracting intri- cate and data-driven features due to the convolutional layer present in them [15]. The convolutional layers employ cascades of filters, known as kernels, on input signals, organised into feature maps that capture diverse features. The most widely used form of CNN is the 2D CNN [16].

Although 2D CNNs are widely used in various appli- cations, their high computational demands necessitate specialized hardware, such as GPUs, for training. This leads to increased energy-consumption on devices, especially during training [17]. This aspect limits their effectiveness in the development of edge-based solutions on platforms with limited resources and CPU-based systems [18].To address these challenges, researchers began using 1D CNNs. Unlike 2D CNNs that perform complex convolution computations on multidimensional inputs with kernels, 1D CNNs perform convolution on one-dimensional arrays with a kernel [19].

Various applications, including audio recognition [20], gesture recognition [21], bearing fault diagnosis [22], and electricity load forecasting [23], have successfully employed 1D CNNs [24]. Prior studies utilizing 1D CNNs have mainly focused on optimizing computational resources and reducing the memory footprint during inference. However, research on reducing energy consumption for 1D CNNs on CPU-based, resource-constrained devices, particularly during train- ing, which involves two-pass computation, remains limited [15]. The challenges with existing works on 1D CNN audio-based emotion recognition are as follows:

- Energy-intensive solution: The existing works with CNN are computationally intensive and thereby energy-intensive [25]. Making the training energy- intensive.
- Lacks Quality of Experience(QoE) for End-Users: Due to the resource requirements, the CNNs-based solutions were trained and deployed in the cloud [1]. Leading to issues like latency, delayed re- sponse, and increased bandwidth consumption [26]. This ultimately leads to the degradation of the qual- ity of experience for the end user.
- Existing solutions are resource inefficient: The ex- isting works with CNN are resource-intensive[27]. Making the training incompatible on resource- constrained CPU-based devices.
- Considering the above challenges, this work focuses on developing an energy-reduced and CPU-adaptable 1D- CNN design for resource-constrained edge[28] devices. The main contribution of this paper can be summarised as follows:
- Low-Powered and parameter-reduced 1D CNN ar- chitecture compatible with edge and CPU-based de- vice for audio emotion detection: We proposed a novel 1D CNN model inspired by the ecological en- ergy pyramid, aimed at significantly reducing over- heads in terms of energy consumption for audio emotion detection during training.
- Evaluation of the Proposed Solution on Benchmark Audio Datasets We assessed the effectiveness of the proposed architecture across five widely used public audio datasets: CREMA-D, EMO-DB, RAVDESS, TESS, and SAVEE, confirming the system's re- producibility. On the RAVDESS dataset, the ar- chitecture achieved an accuracy of 99.26% with 0.9 million parameters and 98.21% with 0.2 mil- lion parameters. For the TESS dataset, the model reached 99.99% accuracy with 0.9M parameters and 99.87% with 0.2M. On CREMA-D, the system recorded 98.02% accuracy with 0.9M parameters and 97.44% with 0.2M. Finally, for both EMO-DB and SAVEE, it attained 99.9% accuracy using either 0.9M or 0.2M parameters.
- Evaluation of proposed solution for energy-reduction on various audio datasets: We evaluated EPyNet and found that it outperforms existing mod- els in both recognition accuracy and energy effi- ciency. Specifically, the 0.06M model achieves a 76.5% energy reduction compared to CNN, the 0.2M model reduces energy by 64.9%, and the 0.9M model achieves a 70.3% energy reduction.
- Evaluation of proposed solution on different activation and pooling functions: We evaluated the energy consumption of different activation functions with average pooling and found that it consumed more energy compared to max pooling. ReLU showed an increase of approximately 0.06%, Swish showed around 1.77%, Sigmoid demon- strated about 19.43%, and PReLU exhibited a sig- nificant rise of approximately 61.71% in energy consumption with average pooling.
- Evaluation of proposed system with different data dimensions and frameworks: We tested our pro- posed systems on 1D and Pytorch

framework which follows a channel-first approach NCHW

Our work delves into the following research questions:

- How can 1D CNN deep learning architectures be further optimized for energy reduction without compromising performance on CPU-based devices for audio-based emotion detection applications?
- How is the applicability of the solution designed for 1D CNN on cross-domain and cross-framework?
- How does changing the number of filters within the architecture affect the performance of deep-learning solutions that utilize different pooling and activation functions?

The structure of the paper is as follows: Section 2 pro- vides a review of related literature. Sections 3 and 4 de- tail our proposed method for detecting emotions in audio signals. The experimental setup and corresponding find- ings are discussed in Section 5.

Related Works

2.1 Lightweight design techniques for CNN

In the current state of the art, parameter reduction techniques in CNN can be categorized into two based on the number of training rounds required: dual round training and single round training.

In the first approach, researchers have introduced tech- niques like knowledge distillation [29], where initially a large model is trained, and then the softmax predic- tions are combined with targets to train a smaller net- work. Some recent works [30] include the proposed pyramid distillation framework, which is built by stack- ing multiple sets of deep mutual learning.Another work

[31] proposed knowledge distillation on a pre-trained model. Another method consists of post-pruning [32] includes initially training a model and then retraining it by removing redundant weights, connections, and neurons from the trained network. Some of the re- cent work includes [33] proposed layer relevance-wise pruning for CNN, Another work [34] proposed differ- entiable network channel pruning on ResNet and Mo- bileNet V2. Another approach, post-quantization [35], reduces the precision of the computation by converting floats into integers and then performing low-precision computations. Some of the recent work includes [36] proposed decorrelating transform on weight matrices af- ter training. Another work [37] proposed applied post-quantization on the ensemble of federated learning with CNN and achieved a low model size.Another work [38] proposed a hardware-based solution on FPGA. Another work [39] combined pre-trained models with weight quantization. However, these methods can only pro- vide lightweight models for inference, and the training of these approaches requires huge computational resources.

In the second approach, researchers introduced tech- niques like Quantization Aware Training (QAT), where the quantization is performed during training instead of after training, and hence it required no re-training. Some of the recent works include: The proposed ultra-low mixed-precision quantization techniques were specifi- cally designed for deploying the You Only Look Once (YOLO) framework. These techniques integrated 1-bit backbone and 4-bit head quantization methods, along with trainable scale and progressive network quantiza- tion strategies. Another work [40] proposed QAT in a multi-headed convolutional neural network. Another work proposed [41] an 8-bit dynamic fixed point to re- duce the memory footprint of RNN and LSTM mod- els. Another approach is convolution optimization tech- niques, which include operations like depthwise separa- ble, octave, and Winograd convolutions, which replace the expensive convolution operation with less expensive computations [42]. These approaches consume fewer re- sources as compared to dual training. However, these approaches do not always guarantee CPU compatibility [43], [44].

2.2 1D CNN

In the current state of-the art, 1D CNN is introduced to reduce the computation load of 2D CNN.Some of

the recent works include: The work [45] proposed an ap- proach to image super-resolution (SR), leveraging both 1D and 2D CNN to enhance the resolution of im- ages by reducing the model size. Another work [46] utilized 1D combined with a multi head self-attention model to improve the relation extraction of the input. Another work [47] replaced the traditional convolu- tion layers of a transformer with 1D convolution lay- ers. The work [48] presents the implementation using a 1D CNN network with leaky RELU activation. An- other work [49]proposed an explainable and lightweight 1D CNN (ELCNN) model for vibration fault diagno- sis that addresses the challenges of computational com- plexity and interpretability by optimizing feature extrac- tion and classification layers.The work [50] proposed skeleton-based human action recognition (HAR), lever- aging one-dimensional convolutional neural networks (1D-CNN) with singular value decomposition for weight compression in fully connected layers. However, these solutions are more focused on reducing the resources used. As per the best of our knowledge and the survey performed, there is a lack-of study on the energy- con- sumption of 1D-CNN [19].

2.3 CNN-based audio Emotion-Recognition

In the present advancements of the field, CNN-based audio emotion recognition can be found widely. We conducted a bibliometric analysis of 544 papers related to ”Audio Emotion Detection”, which were collected through web scraping from Google Scholar. After clean- ing and filtering the data, we focused on the highest-cited papers and analyzed them using VOSviewer to exam- ine research trends, key authors, citation patterns, and emerging topics in the field. This approach provided valuable insights into the evolution and impact of au- dio emotion detection research.

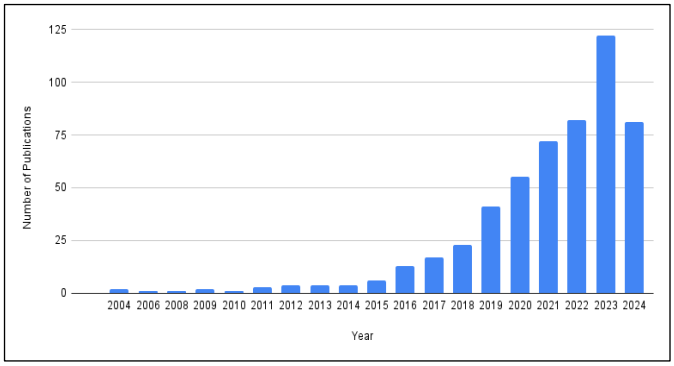

Figure 1: Trends in the number of publications on CNN-based Audio Emotion Recognition from 2014 to 2024.

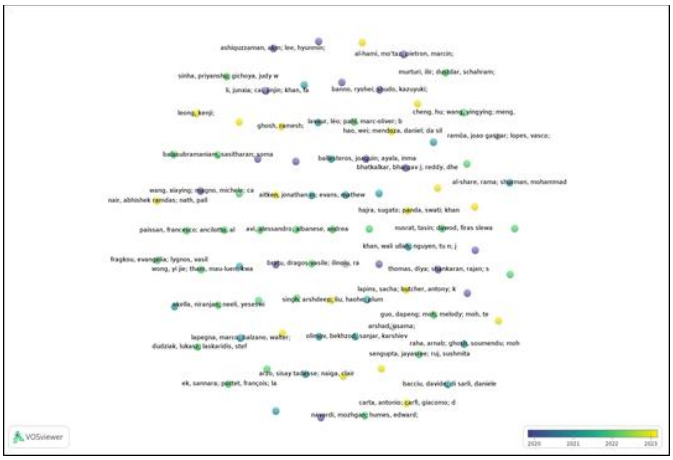

Figure 3: Trends in author contributions to Audio Emotion Detection research from 2004 to 2024.

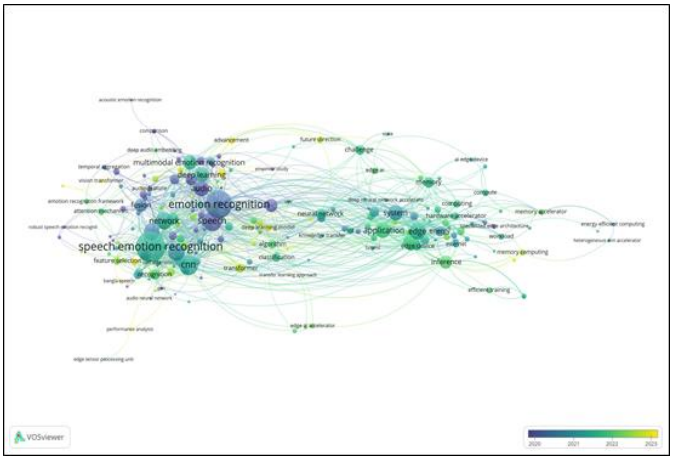

Figure 2: Trends in keyword usage in CNN-based Audio Emotion Recognition research from 2004 to 2024.

- Which authors have made the most significant con- tributions to publications in the field of Audio Emo- tion Detection between 2004 and 2024?

- What are the most frequently occurring keywords in the literature on CNN-based Audio Emotion Recog- nition?
- How has the volume of publications on CNN-based Audio Emotion Recognition evolved from 2004 to 2024, and which paper has received the highest ci- tation during this period?

Our bibliometric analysis showed that: Figure 1 illustrates a notable increase in the volume of publications on CNN-based Audio Emotion Recognition from 2014 to 2024. This rise can be attributed to significant advance- ments in technology, particularly improvements in GPU hardware, which have facilitated more efficient and scal- able deep learning models. Figure 2 The figure illustrates the evolution of key research topics in CNN-based Au- dio Emotion Recognition over the years. Initially, tra- ditional terms like ”speech emotion,” ”emotion recog- nition,” and ”audio signal processing” were dominant. However, from 2020 onwards, newer keywords such as ”deep learning,” ”transformer,” ”edge computing,” and ”EdgeAI” have gained prominence, reflecting advance- ments in technology and the integration of cutting-edge AI and edge-computing methods into the field. Figure 3 illustrates authors who have made the most significant contributions during this period.

Here are some of the recent works in CNN-based audio- based emotion detection include: the work [51] dis- cusses real-time audio extraction and prediction. It em- ploys 1D CNN, 2D CNN, and LSTM, respectively, for the task of comparing the accuracy of three different models.Another work [52] introduces an approach for depression recognition using a graph neural network (GNN) framework that captures connections within and between audio signals by incorporating gated recurrent units (GRUs). Another work [53] implemented mul- timodal emotion recognition combining 2D CNN for video and 1D CNN for audio, combined with late fu- sion. Another work [54] tackles the detection of disrup- tive situations in public transportation by framing it as an audio emotion recognition problem, utilizing a combination of CNN and SVM.The work [10] introduces a framework utilizing fuzzy logic. Another work [55] in- troduces an approach for detecting stress and anger us- ing convolutional neural networks (CNNs), integrating both handcrafted features and deep-learned representa- tions extracted from audio spectrograms. The work [56] a audio representation using autoencoders for real-time emotion recognition from audio.Another work [57] in- tegrated a dilated convolutional neural network with a multi-headed attention mechanism.However, the intro- duction of 1D-CNN into the audio-based emotion detec- tion task reduced the resource requirement as compared to 2D CNN. However, due to the large parameter size of the existing models, it consumes a good amount of energy and is difficult to deploy in resource-constrained devices [58].

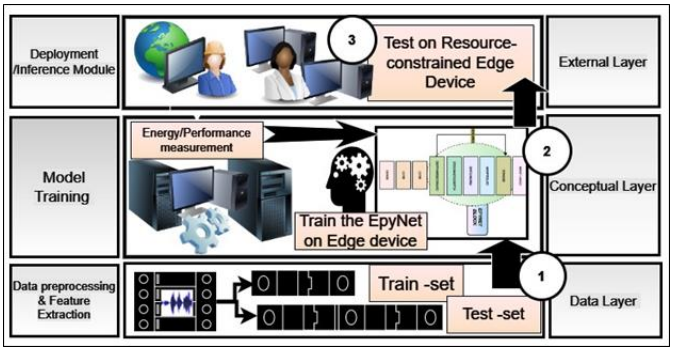

Figure 4: Low power audio emotion recognition utilizing EpyNet

Problem Formulation

In this study, we aim to design a deep learn- ing model, D, with parameters θ that minimizes energy consumption during training on a resource-constrained edge device. The model will be tailored for efficient adaptation to such environments, ensuring low energy usage while meeting specific hardware limitations. The total energy consumption E(θ) is defined as the integral of the power consumption P(θ, t) over the training duration T :

E(θ) =

T

θ, t) dt (1)

0

The design process is subject to the following constraints:

CPU cores in use ≤ 8, CPU frequency ≤ 3.0 GHz, Memory usage ≤ 8 GB,

No GPU usage.

The objective of this research is to minimize E(θ), where P(θ, t) represents the power consumed by the model D over time.

Proposed-System

4.1 Data Preprocessing

Supervised learning was employed for our classification model, which involved training and testing data con- sisting of audio samples categorized into emotions such as sad, happy, disgust, and anger, each appropriately la- beled. One-hot encoding was utilized for representing the output labels.

In audio emotion detection, where there are multiple classes of various emotions, it's essential to encode these emotions into numerical representations that can be processed by neural networks, which inherently work with numerical data rather than strings or char- acters. The data preprocessing steps used in the de- sign of the proposed audio Emotion Recognition model are elaborated upon in this section.

4.2 Feature-Extraction and Data-Preparation

The initial step in our data preprocessing phase involves extracting and selecting a specific set of acoustic features. For this purpose, we utilized Mel-Frequency Cepstral Coefficients (MFCCs) [59]–[61], Zero-crossing Rate [62]–[64], Spectral Centroid [65], [66], Spectral Rolloff [67], [68], Spectral Bandwidth [69], [70], Mel- Frequency Cepstral Delta Coefficients [71], [72]. Here, In accordance with the Nyquist-Shannon sampling the- orem [73], a sampling rate of 44 kHz was employed. Upon extracting the relevant acoustic features from the audio signals, these features were consolidated into a unified feature vector of dimensions [1, 44]. This vec- tor served as the input for the subsequent stages of the emotion recognition model. Since the task involves clas- sification, the dataset was further processed to align with the requirements of the classifier.

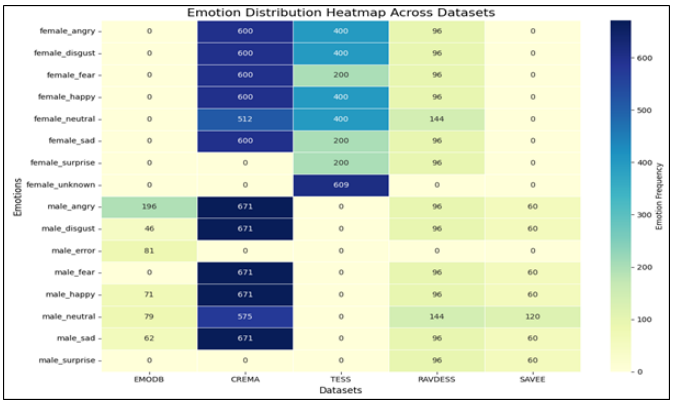

Figure 6: 75:25 ratio datasplit of the dataset

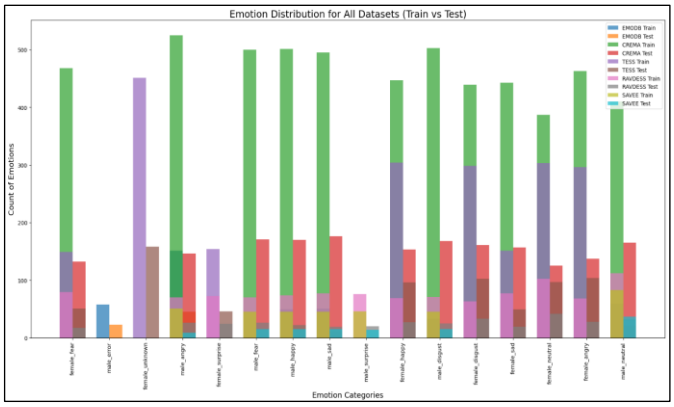

Figure 5: Heatmap Comparison of various audio datasets

- Feature-Vector Formation: We began by extracting the aforementioned features from the dataset. Since these features are categorical, we applied one- hot encoding to transform them into binary vec- tors. One-hot encoding assigns a distinct binary value (0 or 1) to each category within a feature, enabling the classifier to handle each category sep- arately and preventing any assumptions of ordinal relationships.

The mathematical formula for one-hot encoding [74] can be represented as:

OneHot Encoded Vector(c) = [0, . . . , 1] at index c (2)

OneHot Encoded Vector(c) denotes the resulting binary vector for that category.Figure 5 shows the

heatmap plot of Ravdess,EmoDB,Savee,Crema-D datasets.

- Data Split: After encoding the labels into vectors, each audio file in the dataset was represented by a vector of size [1, 45]. We then divided the dataset into training and testing sets with a 75:25 ratio [75].Figure 6 shows the data split utilized in the work.

4.3 Model Design

Our work presents a deep-learning architecture in- spired by the energy ecological pyramid [76], as depicted in Figure 7. Guided by the principle that energy can- not be created or destroyed but only converted between forms [77], we designed our CNN model accordingly, akin to the law of conservation of energy. In this anal- ogy, the CNN model, acting as a core computing unit, receives energy from a singular source. The initial layer consumes energy to produce outputs for subsequent lay- ers, akin to the ecological energy pyramid's principles. As the computational units in each layer increase, more energy is needed, accounting for some useful energy dis- sipating as heat. We introduced the Epynet block (Fig- ure 9) inspired by these ecological structures, featuring a pyramidal structure with logarithmically reduced fil- ters in each layer. Our study emphasizes the strategic placement of nodes and layers for improved energy ef- ficiency compared to existing approaches. We posit that the arrangement of components in a neural network sig- nificantly impacts the quality, speed, and efficiency of information processing.

Table 1: Parameter Settings of the Model

Num.	Type/Stride	Filter Shape	Input Size
0	Input	-	44 x 1
1	Conv1D/s1	256 filters of 5x1	44 x 256
2	MaxPool1D/s2	Pool size 2x1	22 x 256
3	DWConv1D/s1	72 filters of 3x1	22 x 256
4	Conv1D/s1	128 filters of 1x1	22 x 128
5	MaxPool1D/s2	Pool size 2x1	11 x 128
6	DWConv1D/s1	216 filters of 3x1	11 x 128
7	Conv1D/s1	64 filters of 1x1	11 x 64
8	LSTM	32 units	11 x 32
9	LSTM	16 units	16
10	Dense	8 units	16
11	Dense	8 units	8
12	Dense	14 units (classes)	8

Dtrain, Dtest = Split(D, train ratio) (3) Here, D denotes the original dataset containing N instances. The function Split() divides the dataset D into two subsets, Dtrain and Dtest, according to the specified train ratio.

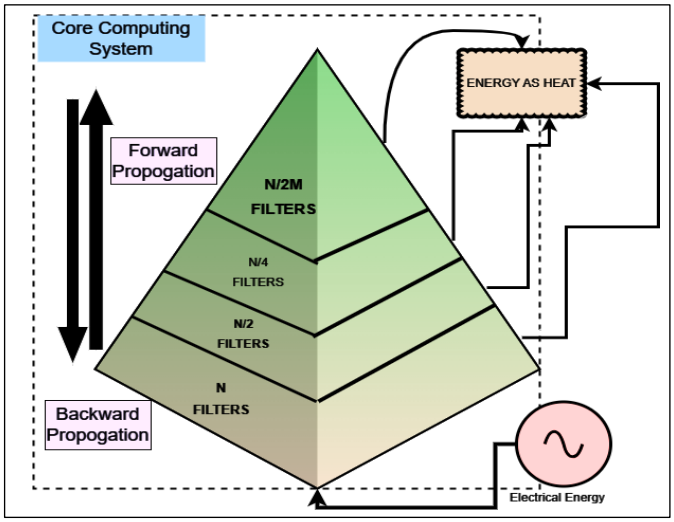

Figure 7: EPyNet block architecture (bottom-up abstract representa- tion) inspired by the ecological energy pyramid. Where each convolu- tion layer has N/2m neurons. Where m is the index of the round. Here, the core computing system represents CPU and memory. Assuming Epynet block is present in main memory.

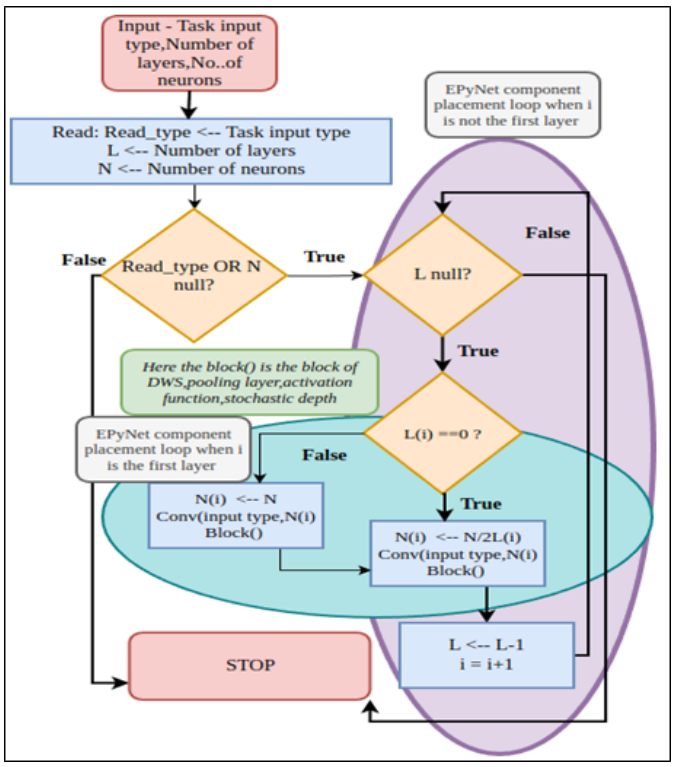

Figure 8: Component placement flow chart representation of EPynet

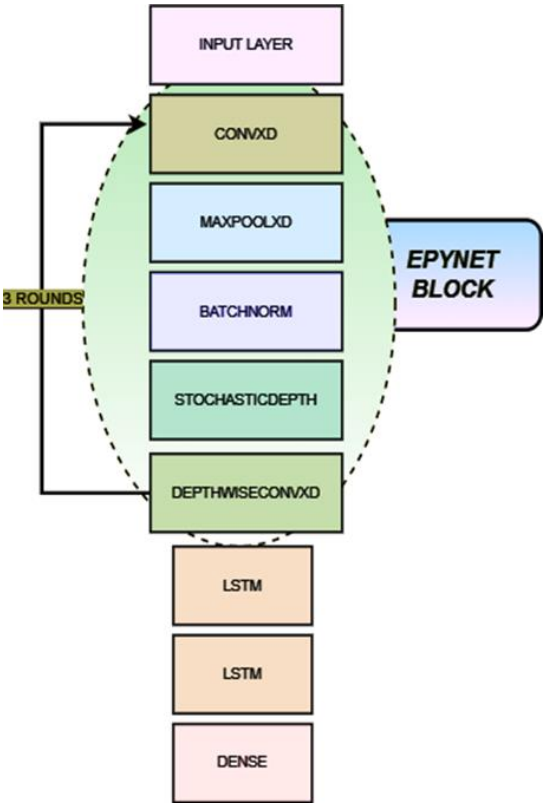

Figure 9: EPyNet block embedded network architecture

4.4 EPyNet Architecture

Figure 7 shows the proposed model is capable of re- ducing the parameters of the entire network to obtain more 1D information without increasing the computation burden or the number of channels due to the smaller number of filters in each layer. Here we represent the detailed flow as shown in Figure 8.

In this section, we discuss in detail the design of a lightweight CNN model and an EpyNet block proposed for audio emotion. The objective is to reduce the model size through parameter reduction while maintaining clas- sification efficiency and energy reduction during training on CPU and resource-constrained devices. The EpyNet block used in the model is a combination of depth-wise convolution and stochastic depth. Here we employed depthwise before the convolution, and the parameter dif- ference in terms of ratio is shown below:

$$\frac{P_{Dconv,henconv}}{P_{Conv\_then\_DS}} = \frac{C_{in} \times K + C_{in} \times C_{out}}{C_{in} \times C_{out} \times K + C_{out} \times K} \quad (4)$$

Let Cin represent number of input channels, Cout rep- resents the number of output channels, and K repre- sents the kernel-size.From the equation, it can be inferred that the number of parameters in our proposed architec- ture is very low due to the use of a depthwise layer before the convolution layer.

Our proposed Epynet block is composed of five layers ar- ranged sequentially: a convolutional layer, max pooling, batch normalization, followed by stochastic depth and depthwise convolution layers, and then another convolu- tional layer. These sequences are repeated throughout the architecture. Figure 9 illustrates the detailed flowchart of the Epynet design, while Table 1 outlines the network's architecture. For audio emotion recognition, we replaced X in ConvXd with 1. The first convolutional layers con- tain 1024 filters, the second set has 512 filters, and the third convolutional layers use 128 filters. We employ a hierarchical reduction in the number of filters to decrease computational complexity at each layer. Each filter has a size of 5 × 5. The architecture utilizes

three Epynet modules, applying filters of varying sizes as depicted in Figure 9. Parameters in the Epynet blocks are initial- ized randomly. The ReLU activation function is applied after every convolutional layer. Instead of a flattening layer, a depthwise convolution layer is used at the output of the Epynet block, significantly reducing the number of trainable parameters. Finally, a fully connected (FC) layer with a SoftMax activation function is used to carry out the classification task. We incorporated a pyramidal architecture by adding N neurons in each layer, where m represents the round index. Based on the importance of each neuron, only the neuron with the highest sig- nificance was forwarded to the next round. To achieve this, we utilized stochastic depth. Here, we assumed a stochastic depth probability of 0.5 for moving to the next layer, a value chosen based on previous studies [78]– [80]. The model remains lightweight, containing approx- imately 0.9 million parameters when N = 1024, 0.2 mil- lion with N = 512, and 0.06 million with N = 256, which is substantially fewer than many state-of-the-art deep CNNs used for audio emotion recognition. A de- tailed set of experiments and comparisons are provided below.

Experiments and Results

The data preprocessing and training experiments were conducted on a hardware platform equipped with an Intel Core i9 processor run- ning Ubuntu 20.04 64-bit OS. Python 3.6 was used for coding the model, and the TensorFlow deep learning framework was used to construct the model structure. For the experiments, three model variations were considered with initial filter sizes (N) of 1024, 512, and 256, corresponding to models with 0.9 million (0.9M), 0.2 million (0.2M), and 0.06 million (0.06M) parameters, respectively.

These variations were in- cluded to analyze how the model's energy consumption changes with different numbers of filters.

Energy measurement during the experiments was conducted using PyRAPL, a toolkit available to assess the en- ergy consumption of a host machine while executing Python code [81].

Detailed descriptions of the conducted experiments and their results are provided in the following subsections.

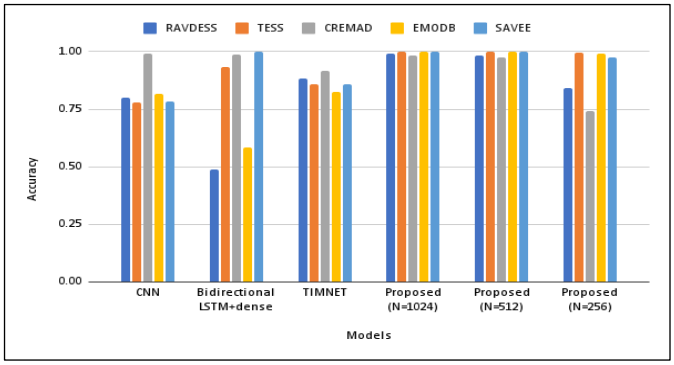

Figure 10: Evaluation of proposed EpyNet for accuracy on different datasets

Table 2: Dataset Compared-Where R indicates RAVDESS [82],T indi- cates TESS [83] ,C indicates CREMA-D [84],S indicates SAVEE [85] and E indicates EMODB dataset [86]

Data	E	C	T	S	R
Samples	535	7442	2940	480	7200
Size(GB)	0.04	0.47	0.44	0.0205	0.45

5.1 Testing the Proposed EpyNet for Audio emotion de- tection task

In this section, we aim to validate the effectiveness of our proposed method for audio emo- tion detection.We constructed an experimental setup with three different versions of the model (N = 1024, N = 256, and N = 512) filter size and utilized datasets from CREMA-D, RAVDESS, TESS, EMO-DB, and SAVEE obtained from Kaggle. The parameter settings employed for the detection task are illustrated in Table 1. The results presented in Figure 11 and Figure 10 demonstrate that compared to other networks like TIMNET [87], Bidirectional LSTM+dense [88], and CNN [89], our proposed models achieve higher accuracy with lighter parameter sizes and reduced energy consumption.

Our findings indicate that dataset size directly influences model energy consumption. Table 2 illustrates dataset sizes and their respective memory requirements. For instance, the model with N = 128 filters has a smaller size and consumes relatively less energy compared to other versions of the proposed model. However, when dealing with large datasets such as CREMA-D, even this model exhibits higher energy consumption. This suggests that the energy consumed by the application is dependent on the size of the dataset. Furthermore, from Figure 11, it is observed that models with fewer convolution operations per layer consume less energy. For instance, TIMNET, despite its complex architecture, achieves lower energy consumption per dataset due to a smaller number of convolution operations. Therefore, we conclude from this study that the overall energy consumption of the model is influenced by the frequency of complex operations and the size of the dataset.

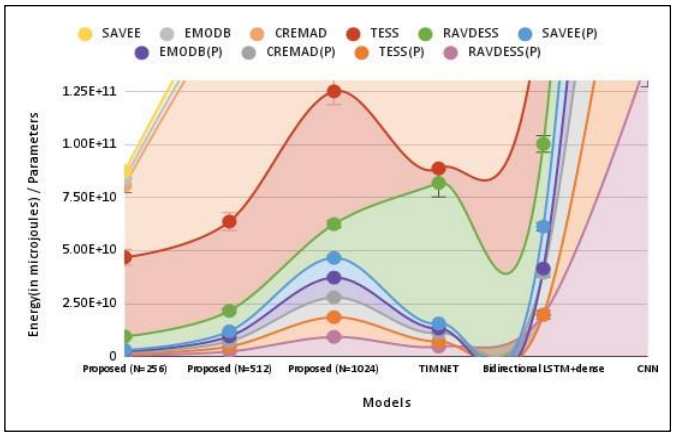

Figure 11: Evaluation of proposed EpyNet for energy-consumed on different datasets

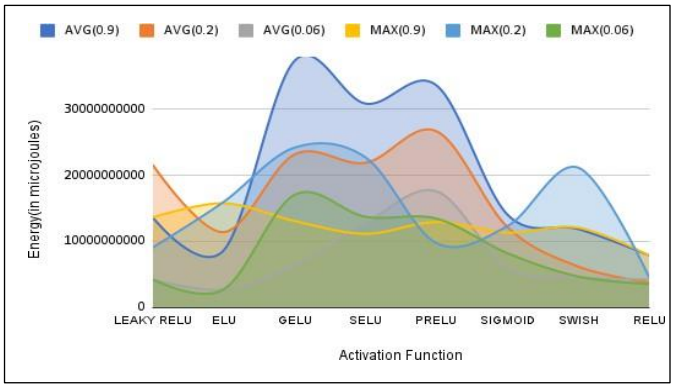

Figure 12: Evaluation of proposed EpyNet for different Pooling and activation functions on energy consumption

5.2 Evaluation of different pooling functions and acti- vation functions

This study investigated the impact of filter size on con- volution operation, affecting model size and parameter count, and subsequently influencing classification accu- racy and energy consumption. We compared network classification accuracy across different filter sizes, acti- vation functions, and pooling techniques, as depicted in Figure 12 and Figure 13.

In this section, we examined the effects of different pooling methods—specifically max pooling and aver- age pooling—combined with various activation func- tions on the energy consumption of 1D CNNs.The acti- vation functions evaluated in this study include Sigmoid, Leaky Rectified Linear Unit (LeakyReLU), Gaussian Er- ror Linear Unit (GeLU), Exponential Linear Unit (ELU), Scaled Exponential Linear Unit (SeLU), Parametric Rec- tified Linear Unit (PReLU), Swish, and Rectified Lin- ear Unit (ReLU). Figures 12 and 13 illustrate that, for a model with 0.06 million parameters, max pooling paired with GeLU activation resulted in the highest energy con- sumption, while average pooling with PReLU activation had the lowest. When the model complexity increased to N = 256 filters, max pooling consumed the most energy with ELU activation, whereas average pooling peaked with PReLU activation. Further increasing the model size to N = 1024 filters showed that average pooling

com- bined with GeLU activation led to greater energy con- sumption, and max pooling peaked with ELU activation. These results suggest that as model complexity grows, the choice of pooling and activation functions has a no- table effect on energy usage. The increase in energy con- sumption may be due to greater memory and CPU uti- lization during computations, leading to more frequent context switching between CPU and memory resources [90]. Based on these findings, we infer that model size is generally proportional to energy consumption; however, models with fewer parameters do not necessarily guaran- tee lower energy use. The energy demand is substantially influenced by the specific combination of activation and pooling functions applied. Moreover, the selection of these functions also affects model accuracy, even when datasets and architectures remain constant.

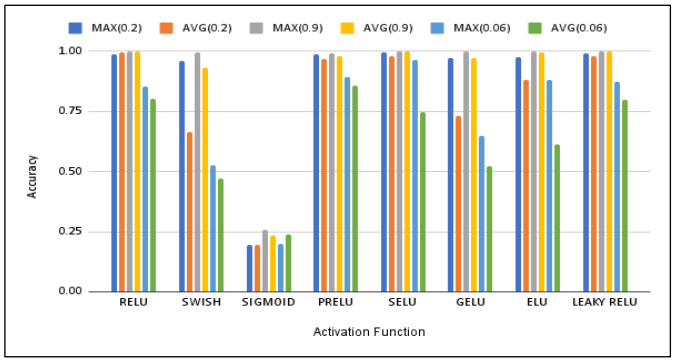

Figure 13: Evaluation of proposed EpyNet for different Pooling and activation functions on accuracy

5.3 Evaluation on different Loss Functions

To evaluate the performance of our proposed model and investigate the effect of various loss functions, we trained the model using the RAVDESS dataset. The results of this evaluation are presented in Figure 14.Figure14 depicts the 0.9M size model for the metrics training and validation loss, accuracy, and energy.Figure16 and Figure 18 shows the perfor- mance of the model for 0.2M and 0.06M respectively. We observed that the proposed model's performance across various loss functions indicated that Categorical Crossentropy was the most effective, achieving high accuracy and low loss, which converged smoothly after approximately 30 epochs. But in terms of energy, Cate- gorical Crossentropy required slightly more energy but provided the highest accuracy and best generalization. Figure 15, Figure16 and Figure19 illustrates the cpu,wall and system time.

Table 3: Performance Comparison of Various Models on Google Cloud CPU

Method	Memory size	Parameter	FLOPS	CPU Time	System Time	Wall Time
MobileNetV1	12.36 MB	3239114	6478228	1min 57s	11.3 s	2min 49s
MobileNetV2	8.66 MB	2270794	4541588	13min 4s	34 s	10min 42s
MobileNetV3small	3.60 MB	944890	1889780	6min 33s	18.8 s	6min 38s
EfficientNetB0	15.50 MB	4062381	8124762	4min 11s	11.4 s	4min 50s
EfficientNetB1	25.13 MB	6588049	13176098	34min 18s	1min 24s	28min 23s
EfficientNetB2	29.69 MB	7782659	15565318	37min 27s	1min 27s	30min 43s
EfficientNetB3	41.19 MB	10798905	21597810	5min 26s	13 s	5min 49s
EfficientNetB4	67.49 MB	17691753	35383506	6min 22s	13.7 s	6min 55s
EfficientNetB5	108.85 MB	28534017	57068034	7min 24s	14.5 s	9min 5s
EfficientNetB6	156.34 MB	40983193	81966386	9min 5s	18.4 s	10min 15s

Method	Memory size	Parameter	FLOPS	CPU Time	System Time	Wall Time
EfficientNetB7	244.61 MB	64123297	128246594	11min 26s	21.6 s	14min 1s
EfficientNetV2B0	22.63 MB	5932122	11864244	24min 28s	1min 1s	19min 1s
EfficientNetV2B1	26.49 MB	6943934	13887868	5min 8s	12.3 s	5min 43s
EfficientNetV2B2	33.51 MB	8783464	17566928	5min 18s	12.1 s	5min 44s
EfficientNetV2B3	49.39 MB	12945992	25891984	6min 7s	12.9 s	6min 12s
EfficientNetV2S	77.61 MB	20344170	40688340	1h 12min 30s	2min 20s	53min 39s
EfficientNetV2M	202.80 MB	53163198	106326396	10min 31s	16.9 s	12min 5s
Proposed (N=512)	2.32 MB	608842	1217684	4min 42s	17.1 s	5min 53s
Proposed (N=256)	286.25 KB	73280	146560	2min 55s	12.9 s	2min 58s
Proposed (N=1024)	3.06 MB	802682	1605364	6min 45s	17.8 s	9min 36s

5.4 Evaluation on different datasets

To explore the transferability of our proposed model and examine the impact of different datasets on activa- tion functions, we conducted training using three distinct datasets: RAVDESS, SAVEE, and EMODB. We evalu- ated these datasets by changing the activation functions and filter sizes, as depicted in Figures 21 and 20.

With the RAVDESS dataset, we observed that the N=128 model exhibited higher energy consumption with ELU and PReLU activations compared to the N=256 model. This indicates that certain activation functions may intro- duce computational overhead and increase energy con- sumption, despite reducing model complexity. Con- versely, in the SAVEE dataset, the N=256 model con- sumed less energy than the N=1024 model for both ReLU and ELU activations. Similar trends were ob- served with the EMODB dataset, where the N=256 model showed higher energy consumption compared to the N=1024 model for both ReLU and ELU activations. This suggests that employing small datasets with smaller models may not always yield energy-reduced solutions.

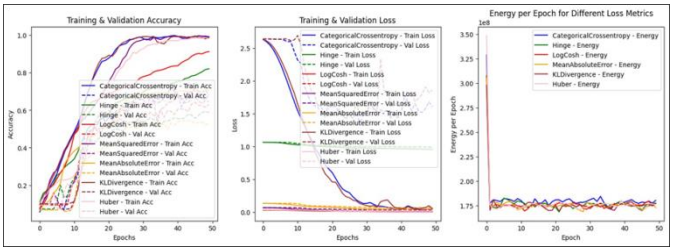

Figure 14: Comparison of Accuracy, Loss, and Energy consumed vari- ous loss functions for proposed 0.9M model

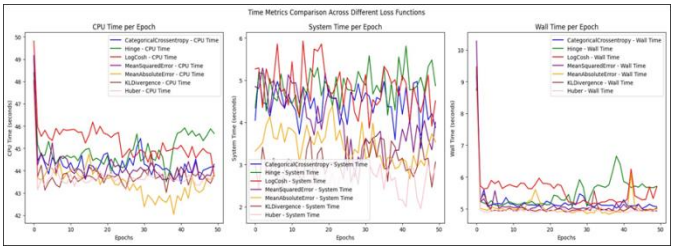

Figure 15: Comparison of CPU Time, System Time, Wall Time various loss functions for proposed 0.9M

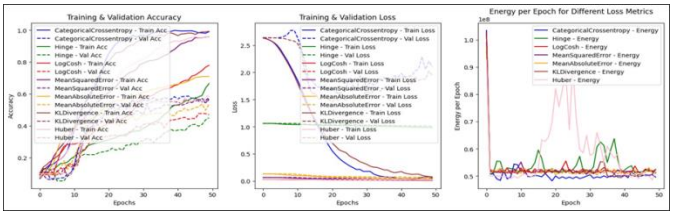

Figure 16: Comparison of Accuracy, Loss, and Energy consumed vari- ous loss functions for proposed 0.2M model

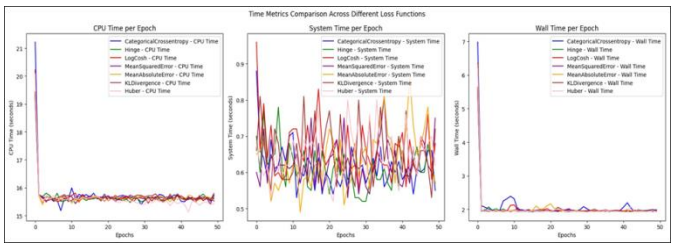

Figure 17: Comparison of CPU Time, System Time, Wall Time various loss functions for proposed 0.2M model

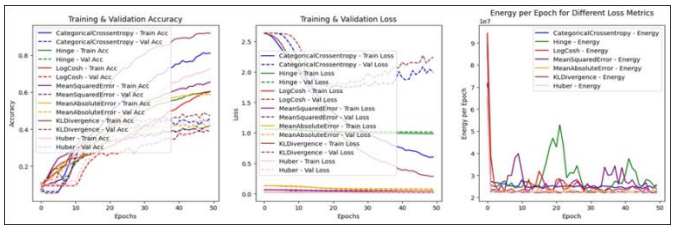

Figure 18: Comparison of Accuracy, Loss and Energy consumed vari- ous loss functions for proposed 0.06M model

5.5 Evaluation on a different data dimension

To assess the performance of our proposed architec- ture on two-dimensional input, we carried out a compar- ative analysis using the CIFAR-10 dataset [91], which is designed for object recognition tasks. The dataset con- tains 60,000 color images of size 32×32, categorized into 10 distinct object classes. It is a subset of the larger 80- million tiny-images dataset, featuring 6,000 images for each class.We employed the CIFAR-10 dataset to assess the applicability of our proposed 1D architecture in the context of 2D systems. To explore the compatibility of our model with different platforms, we conducted exper- iments on a Google Cloud CPU. In our proposed archi- tecture Figure 9), we replaced the XD with 2D and re- moved the LSTM block since no time-dependent input was present.

To evaluate the efficacy and adaptability of our proposed EPynet block on resource-constrained devices, we compared our architecture with widely used archi- tectures designed for resource-constrained devices, such as MobileNet V1 [93], V2 [94], and V3 [95], as de- picted in Figure 22. Additionally, we further com- pared our model with 20 other deep learning architec- tures trained on the TensorFlow framework, with results presented in Table 3. Our experimental findings demon- strated that our proposed architecture achieves better ac- curacy with fewer parameters and faster computation times compared to the other architectures analyzed. Con- sequently, based on this study, we conclude that our ar- chitecture, which outperforms models specifically tai- lored for resource-constrained devices, is well- suited for deployment in such environments. Furthermore, this study also highlights that the fundamental deep learn- ing architecture plays a significant role in achieving optimal-performance, regardless of the input-dimension.

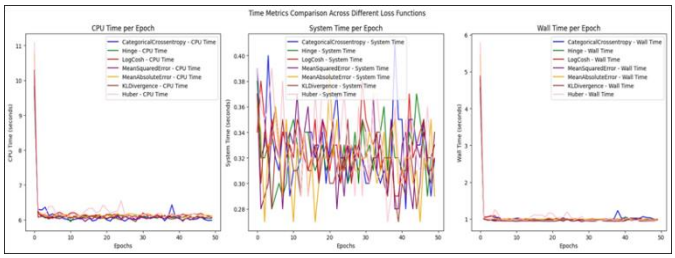

Figure 19: Comparison of CPU Time,System Time,Wall Time various loss functions for proposed 0.06M model

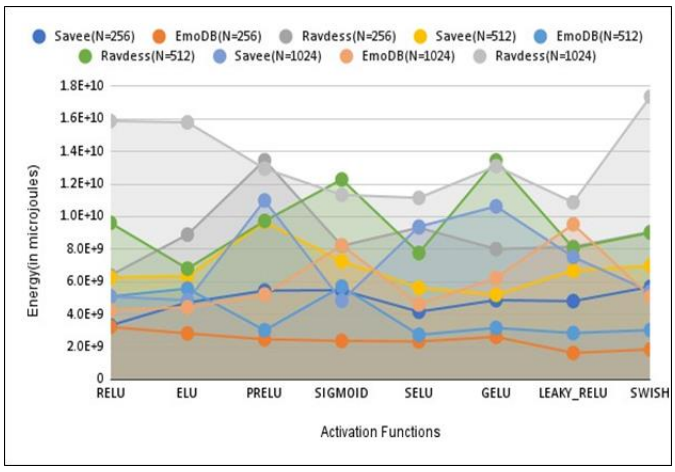

Figure 20: Comparative analysis of activation functions on the EMODB, RAVDESS, and SAVEE datasets for analysing energy con- sumption.

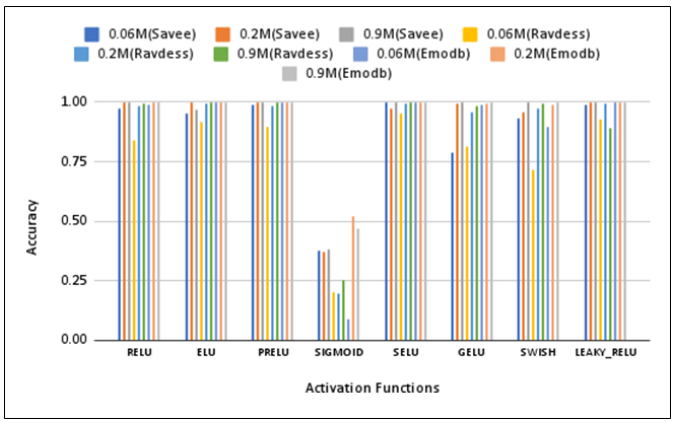

Figure 21: Comparative analysis of activation-functions on EMODB, RAVDESS and SAVEE dataset for accuracy study

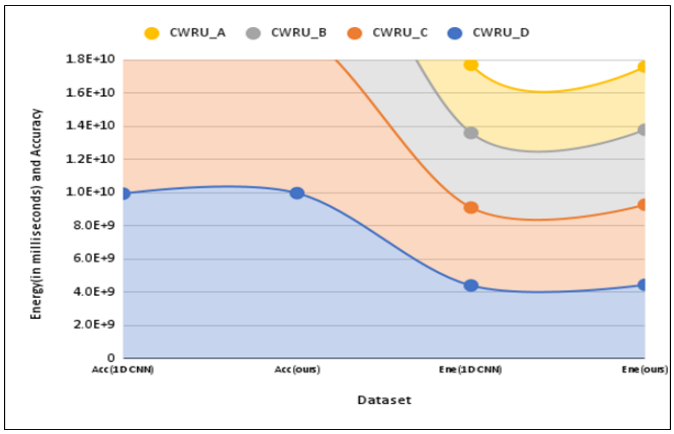

Figure 22: Comparison of the proposed architecture (ours) with the 1D CNN architectures[92]for energy consumption and accuracy (scaled) study for bearing fault classification

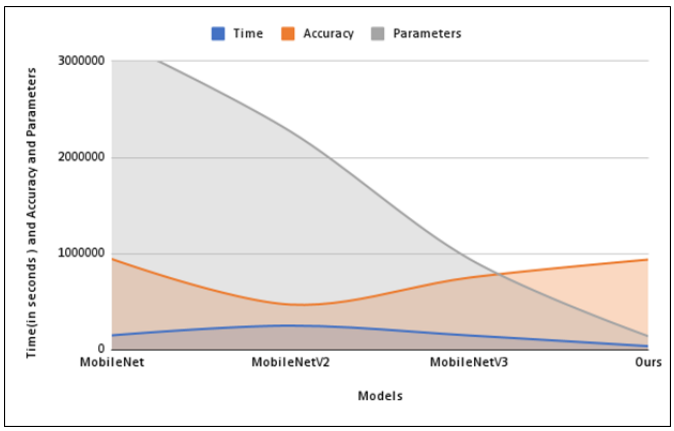

Figure 23: Comparison of the proposed architecture(Ours) with mo- bilenet architectures for model size(parameters in millions),Time in seconds scaled to 10X and Accuracy scaled to 50X for representation purpose

5.6 Evaluation on a different framework

The preceding sections' results demonstrated the ef-fectiveness of our proposed network on the TensorFlow framework for 1D audio signals and 2D inputs when run- ning on Intel i9 and Google Cloud CPUs. To explore the adaptability of our model to the PyTorch framework and other 1D signal types, we conducted training using the CWRU Bearing Dataset on the AMD Ryzen 5 CPU. Bearings, integral to rotating machinery, are challeng- ing to diagnose due to their compact size. Recent stud- ies [96]–[98] have applied deep learning architectures to classify bearing faults. In this study, we tested the suitability of our architecture for this application, per- forming experiments on PyTorch v2.0.1 with the CWRU Bearing Dataset [99]. We evaluated the model's accu-racy and compared it with recent work [92]. The results, depicted in Figure 23, revealed that with each dataset (CWRU Bearing Datasets A, B, C, and D), our pro- posed system outperformed 1D CNN in terms of accu- racy while maintaining reduced-energy.

Conclusion

This paper proposes EPynet, an energy-reduced 1D CNN architecture designed for edge-compatible deep learning applications, with a specific focus on audio emotion detection. The study addresses the challenges of energy consumption for deep learning models on CPUs and resource-constrained edge devices. Implementing a neuron reduction technique bio-inspired from the ecolog- ical energy pyramid, resulted in significant reductions in model parameters while maintaining high accuracy. The experimental evaluations conducted across diverse pub- lic audio datasets, using different frameworks and vary- ing data dimensions on Intel i9, AMD Ryzen 5, and Google Cloud CPUs, demonstrate the energy-reduced and high accuracy of our proposed architecture. How-ever, we performed all the experiments in controlled en- vironments. And in the future, we propose to deploy it on real-time edge devices to evaluate the real-time behavior of the proposed architecture.

Furthermore, we also aim to deploy the solution on multiple devices and analyze how the architecture responds in collaborative scenarios.

Acknowledgements

This research work received partial support from the GOOGLE Cloud Research Credits Program under EDU Credit 324579202.

References

[1]. Y. U¨ . SO¨ NMEZ and A. VAROL, "In-depth investigation of speech emotion recognition studies from past to present the im- portance of emotion recognition from speech signal for ai," In- telligent Systems with Applications, p. 200 351, 2024.

[2]. R. Guo, H. Guo, L. Wang, M. Chen, D. Yang, and B. Li, "Devel- opment and application of emotion recognition technology—a systematic literature review," BMC psychology, vol. 12, no. 1, p. 95, 2024.

[3]. H. R. Kirk, B. Vidgen, P. Ro¨ttger, and S. A. Hale, "The ben- efits, risks and bounds of personalizing the alignment of large language models to individuals," Nature Machine Intelligence, pp. 1–10, 2024.

[4]. H. Ouhaichi, D. Spikol, and B. Vogel, "Research trends in mul- timodal learning analytics: A systematic mapping study," Com- puters and Education: Artificial Intelligence, p. 100 136, 2023.

[5]. Y. Wu, J. Han, Z. Jian, and W. Xu, "Human voice sensing through radio-frequency technologies: A comprehensive re- view," IEEE Transactions on Instrumentation and Measure- ment, 2024.

[6]. A. A. Anthony and C. M. Patil, "Speech emotion recognition systems: A comprehensive review on different methodologies," Wireless Personal Communications, vol. 130, no. 1, pp. 515–525, 2023.

[7]. U. Maniscalco, A. Minutolo, P. Storniolo, and M. Esposito, "Towards a more anthropomorphic interaction with robots in museum settings: An experimental study," Robotics and Au- tonomous Systems, vol. 171, p. 104 561, 2024.

[8]. J. Yu, A. Dickinger, K. K. F. So, and R. Egger, "Artificial intelligence-generated virtual influencer: Examining the effects of emotional display on user engagement," Journal of Retailing and Consumer Services, vol. 76, p. 103 560, 2024.

[9]. Z. Yang, S. Zhou, L. Zhang, and S. Serikawa, "Optimizing speech emotion recognition with hilbert curve and convolu- tional neural network," Cognitive Robotics, vol. 4, pp. 30–41, 2024.

[10]. P. Kozlov, A. Akram, and P. Shamoi, "Fuzzy approach for audio-video emotion recognition in computer games for chil- dren," Procedia Computer Science, vol. 231, pp. 771–778, 2024.

[11]. J. M. Jose, "Optimizing neural network energy efficiency through low-rank factorisation and pde-driven dense layers," International Journal of Research Publication and Reviews, vol. 6, no. 1, pp. 5483–5487, Jan. 2025, IssN: 2582-7421.

[12]. R. Desislavov, F. Mart´ınez-Plumed, and J. Herna´ndez-Orallo, "Trends in ai inference energy consumption: Beyond the performance-vs-parameter laws of deep learning," Sustainable Computing: Informatics and Systems, vol. 38, p. 100 857, 2023.

[13]. N. Aslam, W. Yang, R. Saeed, and F. Ullah, "Energy transition as a solution for energy security risk: Empirical evidence from bri countries," Energy, vol. 290, p. 130 090, 2024.

[14]. L. Alzubaidi, J. Zhang, A. J. Humaidi, et al., "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," Journal of big Data, vol. 8, pp. 1–74, 2021.

[15]. G. Habib and S. Qureshi, "Optimization and acceleration of convolutional neural networks:

A survey," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 7, pp. 4244–4268, 2022.

[16]. Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convo- lutional neural networks: Analysis, applications, and prospects," IEEE transactions on neural networks and learning systems, vol. 33, no. 12, pp. 6999–7019, 2021.

[17]. Q. Zhang, M. Zhang, T. Chen, Z. Sun, Y. Ma, and B. Yu, "Re- cent advances in convolutional neural network acceleration," Neurocomputing, vol. 323, pp. 37–51, 2019.

[18]. C.-C. J. Kuo and A. M. Madni, "Green learning: Introduction, examples and outlook," Journal of Visual Communication and Image Representation, vol. 90, p. 103 685, 2023.

[19]. S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1d convolutional neural networks and applica- tions: A survey," Mechanical systems and signal processing, vol. 151, p. 107 398, 2021.

[20]. A. Mehrish, N. Majumder, R. Bharadwaj, R. Mihalcea, and S. Poria, "A review of deep learning techniques for speech pro- cessing," Information Fusion, p. 101 869, 2023.

[21]. Y. Zhou, M. Shen, X. Cui, Y. Shao, L. Li, and Y. Zhang, "Tribo- electric nanogenerator based self-powered sensor for artificial intelligence," Nano Energy, vol. 84, p. 105 887, 2021.

[22]. M. Pandiyan and T. N. Babu, "Systematic review on fault di- agnosis on rolling-element bearing," Journal of Vibration Engi- neering & Technologies, pp. 1–35, 2024.

[23]. P. Boopathy, M. Liyanage, N. Deepa, et al., "Deep learning for intelligent demand response and smart grids: A comprehensive survey," Computer Science Review, vol. 51, p. 100 617, 2024.

[24]. S. Abdoli, P. Cardinal, and A. L. Koerich, "End-to-end envi- ronmental sound classification using a 1d convolutional neural network," Expert Systems with Applications, vol. 136, pp. 252– 263, 2019.

[25]. Y. Huang, T. Ando, A. Sebastian, M.-F. Chang, J. J. Yang, and Q. Xia, "Memristor-based hardware accelerators for artificial intelligence," Nature Reviews Electrical Engineering, pp. 1–14, 2024.

[26]. A. A. Laghari, X. Zhang, Z. A. Shaikh, A. Khan, V. V. Es- trela, and S. Izadi, "A review on quality of experience (qoe) in cloud computing," Journal of Reliable Intelligent Environments, pp. 1–15, 2023.

[27]. M. R. Falahzadeh, E. Z. Farsa, A. Harimi, A. Ahmadi, and A. Abraham, "3d convolutional neural network for speech emotion recognition with its realization on intel cpu and nvidia gpu," IEEE Access, vol. 10, pp. 112 460–112 471, 2022.

[28]. J. M. Jose, "Edge intelligence: Architecture, scope and appli- cations," Journal homepage: www. ijrpr. com ISSN, vol. 2582,p. 7421,

[29]. L. Wang and K.-J. Yoon, "Knowledge distillation and student- teacher learning for visual intelligence: A review and new out-looks," IEEE transactions on pattern analysis and machine in- telligence, vol. 44, no. 6, pp. 3048–3068, 2021.

[30]. H. Yu, X. Feng, and Y. Wang, "Enhancing deep feature rep- resentation in self-knowledge distillation via pyramid feature refinement," Pattern Recognition Letters, vol. 178, pp. 35–42, 2024.

[31]. Z. Yang, Y. Zhang, D. Sui, Y. Ju, J. Zhao, and K. Liu, "Expla- nation guided knowledge distillation for pre-trained language model compression," ACM Transactions on Asian and Low- Resource Language Information Processing, vol. 23, no. 2, pp. 1–19, 2024.

[32]. Y. Tang, Y. Wang, J. Guo, et al., "A survey on transformer com- pression," arXiv preprint arXiv:2402.05964, 2024.

[33]. S.-K. Yeom, P. Seegerer, S. Lapuschkin, et al., "Pruning by ex- plaining: A novel criterion for deep neural network pruning," Pattern Recognition, vol. 115, p. 107 899, 2021.

[34]. Y.-J. Zheng, S.-B. Chen, C. H. Ding, and B. Luo, "Model compression based on differentiable network channel pruning," IEEE Transactions on Neural Networks and Learning Systems, 2022.

[35]. P. P. Ray, "A review on tinyml: State-of-the-art and prospects," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 4, pp. 1595–1623, 2022.

[36]. S. I. Young, W. Zhe, D. Taubman, and B. Girod, "Trans- form quantization for cnn compression," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 9, pp. 5700–5714, 2021.

[37]. P. V. Astillo, D. G. Duguma, H. Park, J. Kim, B. Kim, and I. You, "Federated intelligence of anomaly detection agent in iotmd-enabled diabetes management control system," Future Generation Computer Systems, vol. 128, pp. 395–405, 2022.

[38]. J. Hwang, A. S. Uddin, and S.-H. Bae, "A layer-wise extreme network compression for super resolution," IEEE Access, vol. 9, pp. 93 998–94 009, 2021.

[39]. M. Hussain, M. Fiza, A. Khalil, et al., "Transfer learning-based quantized deep learning models for nail melanoma classifica- tion," Neural Computing and Applications, vol. 35, no. 30, pp. 22 163–22 178, 2023.

[40]. A. Kumar, A. Vishwakarma, and V. Bajaj, "Multi-headed cnn for colon cancer classification using histopathological images with tikhonov-based unsharp masking," Multimedia Tools and Applications, pp. 1–20, 2024.

[41]. J. Chen, S.-W. Jun, S. Hong, W. He, and J. Moon, "Eciton: Very low-power recurrent neural network accelerator for real-time inference at the edge," ACM Transactions on Reconfigurable Technology and Systems, vol. 17, no. 1, pp. 1–25, 2024.

[42]. Y. Liu, J. Xue, D. Li, W. Zhang, T. K. Chiew, and Z. Xu, "Image recognition based on lightweight convolutional neural network: Recent advances," Image and Vision Computing, p. 105 037, 2024.

[43]. X. Ma, S. Lin, S. Ye, et al., "Non-structured dnn weight prun- ing—is it beneficial in any platform?" IEEE transactions on neural networks and learning systems, vol. 33, no. 9, pp. 4930– 4944, 2021.

[44]. S. Mittal, "A survey on optimized implementation of deep learning models on the nvidia jetson platform," Journal of Sys- tems Architecture, vol. 97, pp. 428–442, 2019.

[45]. J. Park, J. Lee, and D. Sim, "Low-complexity cnn with 1d and 2d filters for super-resolution," Journal of Real-Time Image Pro- cessing, vol. 17, no. 6, pp. 2065–2076, 2020.

[46]. A. Pourdaryaei, M. Mohammadi, H. Mubarak, et al., "A new framework for electricity price forecasting via multi-head self- attention and cnn-based techniques in the competitive electricity market," Expert Systems with Applications, vol. 235, p. 121 207, 2024.

[47]. Y. Wang, S. Zhao, H. Jiang, et al., "Diffmdd: A diffusion-based deep learning framework for mdd diagnosis using eeg," IEEE Transactions on Neural Systems and Rehabilitation Engineer- ing, 2024.

[48]. M. I. Shirazi, S. Khatir, D. Boutchicha, and M. A. Wahab, "Fea- ture extraction and classification of multiple cracks from raw vi- brational responses of composite beams using 1d-cnn network," Composite Structures, vol. 327, p. 117 701, 2024.

[49]. P. Pang, J. Tang, J. Luo, M. Chen, H. Yuan, and L. Jiang, "An explainable and lightweight improved 1d cnn model for vi- bration signals

of rotating machinery," IEEE Sensors Journal, 2024.

[50]. B. Zhang, J. Han, Z. Huang, J. Yang, and X. Zeng, "A real- time and hardware-efficient processor for skeleton-based action recognition with lightweight convolutional neural network," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 66, no. 12, pp. 2052–2056, 2019. doI: 10.1109/TCSII. 2019.2899829.

[51]. A. Sen, G. Rajakumaran, M. Mahdal, et al., "Live event de- tection for people's safety using nlp and deep learning," IEEE Access, 2024.

[52]. A. K. Das and R. Naskar, "A deep learning model for depres- sion detection based on mfcc and cnn generated spectrogram features," Biomedical Signal Processing and Control, vol. 90, p. 105 898, 2024.

[53]. C. Dixit and S. M. Satapathy, "Deep cnn with late fusion for real time multimodal emotion recognition," Expert Systems with Applications, vol. 240, p. 122 579, 2024.

[54]. E. Mancini, A. Galassi, F. Ruggeri, and P. Torroni, "Disrup- tive situation detection on public transport through speech emo- tion recognition," Intelligent Systems with Applications, vol. 21, p. 200 305, 2024.

[55]. S. Kapoor and T. Kumar, "Fusing traditionally extracted fea- tures with deep learned features from the speech spectrogram for anger and stress detection using convolution neural net- work," Multimedia Tools and Applications, vol. 81, no. 21, pp. 31 107–31 128, 2022.

[56]. N. Patel, S. Patel, and S. H. Mankad, "Impact of autoencoder based compact representation on emotion detection from au- dio," Journal of Ambient Intelligence and Humanized Comput- ing, pp. 1–19, 2022.

[57]. K. Mustaqeem, A. El Saddik, F. S. Alotaibi, and N. T. Pham, "Aad-net: Advanced end-to-end signal processing system for human emotion detection & recognition using attention-based deep echo state network," Knowledge-Based Systems, vol. 270, p. 110 525, 2023.

[58]. M. Maithri, U. Raghavendra, A. Gudigar, et al., "Automated emotion recognition: Current trends and future perspectives," Computer methods and programs in biomedicine, vol. 215, p. 106 646, 2022.

[59]. M. Mohan, P. Dhanalakshmi, and R. S. Kumar, "Speech emo- tion classification using ensemble models with mfcc," Procedia Computer Science, vol. 218, pp. 1857–1868, 2023.

[60]. S. P. Mishra, P. Warule, and S. Deb, "Speech emotion recog- nition using mfcc-based entropy feature," Signal, Image and Video Processing, vol. 18, no. 1, pp. 153–161, 2024.

[61]. C. Hema and F. P. G. Marquez, "Emotional speech recogni- tion using cnn and deep learning techniques," Applied Acous- tics, vol. 211, p. 109 492, 2023.

[62]. L. Yunxiang and Z. Kexin, "Design of efficient speech emotion recognition based on multi task learning," IEEE Access, vol. 11, pp. 5528–5537, 2023.

[63]. J. de Lope and M. Gran˜a, "An ongoing review of speech emo- tion recognition," Neurocomputing, vol. 528, pp. 1–11, 2023.

[64]. M. R. Ahmed, S. Islam, A. M. Islam, and S. Shatabda, "An ensemble 1d-cnn-lstm-gru model with data augmentation for speech emotion recognition," Expert Systems with Applications, vol. 218, p. 119 633, 2023.

[65]. P. Singh, M. Sahidullah, and G. Saha, "Modulation spectral features for speech emotion recognition using deep neural net- works," Speech Communication, vol. 146, pp. 53–69, 2023.

[66]. L.-M. Zhang, G. W. Ng, Y.-B. Leau, and H. Yan, "A parallel- model speech emotion recognition network based on feature clustering," IEEE Access, 2023.

[67]. S. Rajesh and N. Nalini, "Polyphonic instrument emotion recognition using stacked auto

encoders: A dimensionality reduction approach," Procedia Computer Science, vol. 218, pp. 1905–1914, 2023.

[68]. A. Bakhshi, J. Garc´ıa-Go´mez, R. Gil-Pita, and S. Chalup, "Vi- olence detection in real-life audio signals using lightweight deep neural networks," Procedia Computer Science, vol. 222, pp. 244–251, 2023.

[69]. S. P. Mishra, P. Warule, and S. Deb, "Variational mode decom- position based acoustic and entropy features for speech emotion recognition," Applied Acoustics, vol. 212, p. 109 578, 2023.

[70]. U. Bilotti, C. Bisogni, M. De Marsico, and S. Tramonte, "Mul- timodal emotion recognition via convolutional neural networks: Comparison of different strategies on two multimodal datasets," Engineering Applications of Artificial Intelligence, vol. 130, p. 107 708, 2024.

[71]. V. Singh and S. Prasad, "Speech emotion recognition system using gender dependent convolution neural network," Procedia Computer Science, vol. 218, pp. 2533–2540, 2023.

[72]. Z.-T. Liu, M.-T. Han, B.-H. Wu, and A. Rehman, "Speech emotion recognition based on convolutional neural network with attention-based bidirectional long short-term memory net- work and multi-task learning," Applied Acoustics, vol. 202, p. 109 178, 2023.

[73]. D. Sanaguano-Moreno, J. Lucio-Naranjo, R. Tenenbaum, and G. Sampaio-Regattieri, "Real-time impulse response: A methodology based on machine learning approaches for a rapid impulse response generation for real-time acoustic virtual re- ality systems," Intelligent Systems with Applications, vol. 21,p. 200 306, 2024.

[74]. A. Khurana, S. Mittal, D. Kumar, S. Gupta, and A. Gupta, "Tri- integrated convolutional neural network for audio image classi- fication using mel-frequency spectrograms," Multimedia Tools and Applications, vol. 82, no. 4, pp. 5521–5546, 2023.

[75]. M. G. Campana, F. Delmastro, and E. Pagani, "Transfer learn- ing for the efficient detection of covid-19 from smartphone au- dio data," Pervasive and Mobile Computing, vol. 89, p. 101 754, 2023.

[76]. R. Trebilco, J. K. Baum, A. K. Salomon, and N. K. Dulvy, "Ecosystem ecology: Size-based constraints on the pyramids of life," Trends in ecology & evolution, vol. 28, no. 7, pp. 423–431, 2013.

[77]. A. R. Kumar and A. M. R. Khan, "Paradigm shift of energy from past to present," 2017.

[78]. G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth," in Computer Vision– ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14, Springer, 2016, pp. 646–661.

[79]. S. Pradhan and S. Longpre, Exploring the depths of recurrent neural networks with stochastic residual learning, 2016.

[80]. C. P. Woods, "Impact of stochastic depth on deterministic and probabilistic resnet models for weather modeling," Ph.D. dis- sertation, Monterey, CA; Naval Postgraduate School, 2022.

[81]. Author(s) on pypi.org, Pyrapl.

[82]. S. R. Livingstone and F. A. Russo, "The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english," PloS one, vol. 13, no. 5, e0196391, 2018.

[83]. K. Dupuis and M. K. Pichora-Fuller, "Toronto emotional speech set (tess)-younger talker happy," 2010.

[84]. K. DONUK, "Crema-d: Improving accuracy with bpso-based feature selection for emotion recognition using speech," Jour- nal of Soft

Computing and Artificial Intelligence, vol. 3, no. 2, pp. 51–57, 2022.

[85]. N. J. Shoumy, L.-M. Ang, D. M. Rahaman, T. Zia, K. P. Seng, and S. Khatun, "Augmented audio data in improving speech emotion classification tasks," in Advances and Trends in Arti- ficial Intelligence. From Theory to Practice: 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2021, Kuala Lumpur, Malaysia, July 26–29, 2021, Proceedings, Part II 34, Springer, 2021, pp. 360–365.

[86]. F. Burkhardt, A. Paeschke, M. Rolfes, W. F. Sendlmeier, B. Weiss, et al., "A database of german emotional speech.," in In- terspeech, vol. 5, 2005, pp. 1517–1520.

[87]. J. Ye, X.-C. Wen, Y. Wei, Y. Xu, K. Liu, and H. Shan, "Tempo- ral modeling matters: A novel temporal emotional modeling ap- proach for speech emotion recognition," in ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Sig- nal Processing (ICASSP), IEEE, 2023, pp. 1–5.

[88]. J. Zhao, X. Mao, and L. Chen, "Speech emotion recognition using deep 1d & 2d cnn lstm networks," Biomedical signal pro- cessing and control, vol. 47, pp. 312–323, 2019.

[89]. D. Issa, M. F. Demirci, and A. Yazici, "Speech emotion recogni- tion with deep convolutional neural networks," Biomedical Sig- nal Processing and Control, vol. 59, p. 101 894, 2020.

[90]. H. Hong, D. Choi, N. Kim, et al., "Survey of convolutional neu- ral network accelerators on field-programmable gate array plat- forms: Architectures and optimization techniques," Journal of Real-Time Image Processing, vol. 21, no. 3, pp. 1–21, 2024.

[91]. A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)," URL http://www. cs. toronto. edu/kriz/cifar. html, vol. 5, no. 4, p. 1, 2010.

[92]. C.-J. Chang, C.-C. Chen, and B.-H. Chen, "Bearing fault diag- nosis based on an advanced method: Id-cnn-lstm," in 2023 IEEE 6th Eurasian Conference on Educational Innovation (ECEI), 2023, pp. 63–66. doI: 10.1109/ECEI57668.2023.10105356.

[93]. D. Sinha and M. El-Sharkawy, "Thin mobilenet: An enhanced mobilenet architecture," in 2019 IEEE 10th annual ubiquitous computing, electronics & mobile communication conference (UEMCON), IEEE, 2019, pp. 0280–0285.

[94]. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.

[95]. A. Howard, M. Sandler, G. Chu, et al., "Searching for mo- bilenetv3," in Proceedings of the IEEE/CVF international con- ference on computer vision, 2019, pp. 1314–1324.

[96]. J. Zarei, M. A. Tajeddini, and H. R. Karimi, "Vibration analysis for bearing fault detection and classification using an intelligent filter," Mechatronics, vol. 24, no. 2, pp. 151–157, 2014.

[97]. A. H. Boudinar, N. Benouzza, A. Bendiabdellah, et al., "Induc- tion motor bearing fault analysis using a root-music method," IEEE Transactions on Industry applications, vol. 52, no. 5, pp. 3851–3860, 2016.

[98]. S. Singh, A. Kumar, and N. Kumar, "Motor current signature analysis for bearing fault detection in mechanical systems," Pro- cedia Materials Science, vol. 6, pp. 171–177, 2014.

[99]. X. Zhang, B. Zhao, and Y. Lin, "Machine learning based bear- ing fault diagnosis using the case western reserve university data: A review," IEEE Access, vol. 9, pp. 155 598–155 608,2021.