An Ensemble Deep Learning based Approach for Ear Recognition

Parul Chutani
Chandigarh University, Gharuan

Nitin Sharma
Chandigarh University, Gharuan

https://doi.org/10.5109/7342462

出版情報：Evergreen. 12 (1), pp.399-422, 2025-03. Interdisciplinary Graduate School of Engineering Sciences, Kyushu University, Japan
バージョン：

An Ensemble Deep Learning based Approach for Ear Recognition

Parul Chutani[1,*], Nitin Sharma[1]
[1]Chandigarh University, Gharuan, Punjab 140413 India

*Author to whom correspondence should be addressed:
E-mail: parulchutani.phd@gmail.com

(Received September 10, 2024; Revised December 18, 2024; Accepted February 16, 2025).

Abstract: The advancement of Artificial Intelligence (AI) significantly impacts biometric identification methods, with ear biometrics becoming a standout approach due to the unique and consistent features of the[1)] human ear. This study uses two datasets, UERC and IIT-D and develops three custom ear recognition models. The primary focus is on employing ensemble learning techniques to enhance the accuracy and robustness of ear recognition systems. Ensemble methods are particularly beneficial as they reduce errors and improve overall performance by combining multiple models. The individual custom models show promising results, with Model 1, Model 2, and Model 3 achieving accuracies of 97.1%, 96.9%, and 97.3% on the IIT-D dataset, and 84%, 91.1%, and 91.6% on the UERC dataset. Combining these models, the average ensemble achieves an accuracy of 97.8% on the IIT-D dataset and 95.2% on the UERC dataset. The weighted ensemble further improves the results, achieving an accuracy of 98.99% on the IIT-D dataset and demonstrating similarly enhanced performance on the UERC dataset with an accuracy of 97.8. These findings highlight the effectiveness of ensemble learning in significantly boosting the performance of ear recognition systems. Future research could delve into more advanced ensemble techniques, employ larger and more diverse datasets, and evaluate the practical applicability of ear biometrics in various conditions.

Keywords: Ear Recognition; Convolutional Neural Network (CNN); Deep Learning; Ensemble Deep Learning

1. Introduction

In the process of technological evolution, there is hardly another powerful force than machine learning. It enhances the ability of systems to acquire intelligence by using data-driven algorithms. There are a lot of applications, from recognition of handwritten character[2,3)], faces[4)], license plate characters[5,6)], and extends into more sophisticated domains, such as medical fields, including the detection of lung cancer[7,8)] and other intricate medical diagnostics.

Biometrics, a key weapon in modern security, relies on the power of machine learning to identify people based on their unique traits[9)]. Biometrics emerges as a shining example, leveraging deep learning's adeptness to unveil the hidden language of human features[3)] , covering the way for foolproof identification. Unlike physiological or behavioural traits, biometrics leverages to identify and validate people by utilizing deep learning capabilities.

Out of all the features that have been looked at, the human ear is one of the most promising candidates[10)] . Its uniqueness, stability over time, and complexity make it a great choice for biometric applications and improvements in security protocols and authentication techniques. The present paper represents a detailed exploration of ear biometrics; it considers the use of machine learning and biometric identification and reveals the latent potential residing in the complicated curves of the human ear.

Since the given type of biometrics has a stable structure at birth, the ear proves an undeniably reliable and non-invasive identifying modality[11)]. Moreover, it can be performed at a distance and in complicated conditions, including those when facial covers are used. It requires an insignificant amount of memory and processing and is also insusceptible to alterations in emotional conditions. In addition to that, it perfectly complements other modalities of biometrics[10,12)].

1.1 Motivation

What is ensemble learning: An approach to machine learning in which multiple models are connected together to increase prediction quality and overall reliability by blending different independent models[13,14)]. In ensemble approaches, the predictions of models like decision trees or neural networks are aggregated, thereby improving resilience and minimizing the shortcomings of each individual model[15,16)]. Since the method produces more accurate and comprehensive results than single models, it

can be used in a wide range of diverse real-world settings because it reduces overfitting, improves model accuracy, and is more stable than single models. This study has demonstrated how ensemble learning can be used to create biometric systems that are more reliable through significant improvements in ear biometric accuracy.

2. Background & Related Work

The authors Ravishankar Mehta et al. introduce a method to improve ear recognition systems by combining transfer learning with lightweight CNNs[17]. They leverage pre-trained models like VGG19, VGG16, and DenseNet201 along with their custom CNN models (Model 4, Model 5, and Model 6) to enhance accuracy and robustness. Model 4 consists of three convolutional layers, Model 5 has several convolutional layers, and Model 6 includes two convolutional layers. Each model incorporates ReLU activation functions, MaxPooling layers, and dense output layers for classification. The ensemble technique generates the final output by averaging the predictions from these individual models. Evaluated on the IITD-II dataset for ear recognition, the model's accuracy is assessed using Precision, Recall, F-Score, and Receiver Operating Characteristic (ROC) metrics. The combined approach of pre-trained and custom CNN models significantly improves ear recognition accuracy, with performance demonstrated through accuracy and loss plots of individual models and the ensemble technique[18].

Authors - Hammam Alshazly et al. in their study[19] in 2020, presented a highly effective approach to ear recognition in unconstrained situations. They investigate diverse transfer learning models on the large EarVN1.0 dataset, which includes over 87,000 ear photographs from 2,500 individuals. The authors propose a two-step fine-tuning technique that employs top-tier deep CNN architectures like VGGNet[20], AlexNet[21], Inception[22], ResNet[23], and ResNeXt[24]. This entails fine-tuning the last few layers and then the entire net- work, which results in huge speed increases. This work emphasizes the potential of deep CNNs with transfer learning for ear recognition in unconstrained conditions, achieving a state-of-the-art (SOTA) rank-1 recognition accuracy of 93.45% utilizing a single ResNeXt101[25] model.

In 2021, H. Alshazly et al.[26], suggestion provides the ear recognition systems with another way of employing deep residual networks (ResNet). In this paper, the authors provide the results of the experiments carried out on four benchmark datasets of ear images which included both constrained and unconstrained images. Of the authors, five variants of ResNet architecture with greater depth and three variants of learning strategies are proposed. To enhance the system's overall performance, ensembles of heterogeneous ResNet models are con- structed using transfer learning techniques. ResNet-based models performed better on all four benchmark datasets than existing SOTA methods, with recognition rates up to 99.5% on AMI, 99.3% on AMIC, 99.2% on WPUT, and 98.5% on AWE. As a result of the study, several ResNet models of varying depths were combined to form ensembles, thereby improving the system's overall performance for recognizing ears.

Sharkas presents two novel techniques for ear recognition in personal identification in this experiment from 2022[27]. An ensemble classifier is employed in the first method, which employs traditional machine learning concepts such as ear im- age segmentation, statistical feature extraction via the discrete curvelet transform and classification using ensemble classifiers. Utilizing pre-trained networks (AlexNet, GoogleNet[28], ResNet50[29]) for end-to-end classification, feature extraction, and ensemble classifier-based categorization are subsequent elements in the second strategy, which leverages the power of deep learning. Evaluation of the AMI and IIT Delhi ear databases reveals the superior performance of deep learning approaches. Interestingly, non-segmented images consistently perform better than segmented images in categorizing ear images, highlighting their potential to be used for biometric security.

In 2018, Zhang et al.'s study addressed the difficulties in ear verification in uncontrolled environments by introducing USTB-Helloear. This new large ear database can capture photos even with partial occlusions, lighting changes or position shifts[30]. To represent ear features, they propose using a powerful neural network called a CNN. This CNN can be adjusted and improved using the USTB-Helloear database[31]. During the CNN training, they used two techniques, softmax loss and center loss. These techniques help create compact and distinct features that can be used to recognize ears, even when they're not fully visible. Additionally, they combine three CNNs with different scales of ear images to create multiple representations of ears for verification. This combination technique improves performance by about 1%, as shown by the experiments, highlighting the usefulness of the modified CNN model. The study also stresses the importance of aligning ears correctly before identification, especially when the ears are partly covered or their position changes.

A study conducted by Ashish Oberoi et al[32] introduces a six-layer deep CNN architecture for ear biometric identification which overcomes the restrictions of these persistent methods due to the COVID-19 pandemic. It is proposed that ear is a reliable biometric feature based on its structure is stable over time. As a further development, a CNN extracts features automatically from ear images and reports a detection performance of 97.36% on the IITD ear dataset. It comprises three main steps: preprocessing, feature extraction, and classification. The CNN uses several pooling layers along with convolutional layers to extract higher-order features, with subsequent fully connected layers to perform the classification. Non-linearity is added by activation functions such as ReLU, pools reduce dimensionality while retaining important

features.

The authors[33] propose a holistic approach for ear recognition which employs Gabor-filter and an ensemble of extracted features from the previously trained CNN models, VGG19 and DenseNet161. The work tackles important problems that biometric systems face including pose variations, illumination variance and rotation-invariance in unconstrained places. The method consists of 6 main steps: pre-processing of input images, generation of Gabor ear representations by means of using multiple Gabor filters, dataset splitting into training and testing sets, execution of feature extraction, model training, and classification. Their properties of spatial localization, orientation selectivity, and spatial-frequency make Gabor filters useful for feature extraction. The models are fine-tuned using 100 output neurons to classify images belonging to 100 subjects and the predictions from the two CNNs are averaged for better results. We validate our proposed model on 1000 images and at unconstrained representation achieve 74.63% accuracy thus outperforming the existing methods. The impact of these choices is limited to the task at hand—small datasets—and demonstrates the value of transfer learning for deep learning architectures that previously limited the effectiveness of biometric recognition in practical applications.

The authors[34] performed a thorough review of ear recognition methods including descriptor-based and deep-learning methods. Their goal was to assess the performance, computational complexity, and resource usages of these models when used for unconstrained environments. Using the challenging Annotated Web Ears (AWE) dataset which encompasses a broad spectrum of ear images, the effect of covariates such as ethnicity, head rotation (yaw, roll, tilt), gender and presence/absence of occlusions and accessories on recognition performance was investigated. Almost all techniques achieve the best results, including computational time performances, and the main contributions include an exhaustive experimental study on recent state-of-the-art techniques showing that descriptor-based methods are quite robust in terms of performance; missing details on deep-neural architectures achieving overall similar speed with proper hardware but higher memory computational requirements during training. The authors also introduced a new dataset, the Annotated Web Ears Extended (AWEx), comprising 4,104 images from 346 subjects, aimed at advancing research in ear recognition. Their findings highlight that high degrees of head movement and accessories significantly affect identification performance, while mild variations have a limited impact. The paper emphasizes the importance of understanding model characteristics to inform future research and development in biometric recognition technology, addressing gaps in existing literature regarding the strengths and weaknesses of current techniques.

A new CSA-GRU framework is proposed[35] to combine CNN and GRU with self-attention mechanisms for ear biometric recognition. The study employs four datasets: IITD-I, IITD-II, AMI, and AWE, implementing a wide range of image augmentation techniques(including flipping, noise, brightness and translation)to support model robustness. To enhance the computational efficiency, the images are standardized to a fixed image size of 50 × 180 pixels. The model is trained for 50 epochs with excellent performance and metrics: recognition: 97.83%, precision: 96.11%, recall: 96.56%, F1-score: 97.34%. L2 regularization and Bayesian optimization are used to overcome the obstacles of dataset variability, overfitting, and parameter selection. CSA-GRU achieves superior performance compared to state-of-the-art approaches, with significant improvements on multiple datasets, affirming its efficacy in ear recognition.

The authors suggest the DeepBio system[36], which is a mixed model using CNNS and Bidirectional Long Short-Term Memory (Bi-LSTM) networks to identify people based on ear biometrics. To prevent overfitting, they use data augmentation methods like flipping, translating, and adding Gaussian noise. The model is trained and assessed on five datasets: IITD-I, IITD-II, AWE, AMI, and EARVN1, measuring performance with recognition rate, recall, precision, and F1-score. For feature extraction, a CNN is followed by a two-layer Bi-LSTM network, with each layer having 512 units. The Bi-LSTM output goes through dense layers, leading to a final layer with 100 classes. Overall, the model has 11,070,052 trainable parameters. The findings show that the model performs well, achieving recognition rates of 97.97%, 99.37%, 98.57%, 94.5%, and 96.87% for the different datasets. This study demonstrates the effectiveness of DeepBio in ear biometrics and points out its potential as a dependable alternative to facial recognition, especially in cases where traditional methods may not work well.

The authors[37] provided an exhaustive survey on ear biometrics highlighting the utilization of deep learning-based approaches where CNNs are notably performed for ear identification. The study also investigated different feature extraction methods (as Zernike Moments, local binary patterns, Gabor filters, Haralick texture moments), as well as compared them to classical machine learning methods, for example, decision trees, naive Bayes,

Table 1. Review of Ear Recognition Methods

Journal & Year	Technique Used	Dataset Used	Findings & Results
Neurocomputing (2023)	Comprehensive survey of ear recognition methods, including 2D, 3D, and deep learning approaches	Evaluation of 34 databases, including 2D, 3D, and video ear datasets	Reviewed ear recognition methods, highlighted the need for robust models, and outlined performance improvements, challenges, and a taxonomy of approaches
Applied Computational Intelligence and Soft Computing, (2022)	CNN and handcrafted feature extraction	Various datasets including UERC, IIT Delhi Ear Database, and others (73 application papers reviewed)	CNN models outperform traditional methods, with a 22% performance improvement; achieving 100% accuracy in well-illuminated and 97% in low-illumination conditions.
CMES (2024)	Hybrid CNN and Bi-LSTM	IITD-I, IITD-II, AWE, AMI, EARVN1	Demonstrated the effectiveness of ear biometrics for human identification. Recognition rates: **IITD-I**: 97.97%, **IITD-II**: 99.37%, **AMI**: 98.57%,
Evolving Systems (2024)	CSA-GRU (CNN + GRU + Self-Attention)	IITD-I, IITD-II, AMI, AWE	Proposed a hybrid ear recognition framework, achieving 99.24%–99.81% recognition rates and high precision (96.11%), recall (96.56%), and F1-score (97.34%).
Evolving Systems (2024)	Ensemble model combining transfer learning with lightweight CNNs (VGG19, VGG16, DenseNet201)	IITD-II dataset (793 images from 221 distinct subjects)	The proposed ensemble model improves ear recognition accuracy, effectively handling small datasets and outperforming state-of-the-art methods.
IEEE Access (2020)	Deep CNNs with fine-tuning strategies	EarVN1.0	Proposed a two-step fine-tuning strategy, achieving rank-1 recognition accuracy above 90%, with ResNeXt101 reaching 93.45%.
IEEE Access (2021)	Deep Residual Networks (ResNet) with transfer learning strategies	Four benchmark ear datasets (e.g., AWE, IIT Delhi-1, IIT Delhi-2, USTB)	Domain adaptation is crucial, with deeper networks outperforming ImageNet features. The model showed significant improvements in recognition rates and achieved top performance metrics (R1, R5, AUC).
Multimedia Tools and Applications (2022)	1. Discrete Curvelet Transform with Ensemble Classifiers 2. Deep Learning (End-to-End using AlexNet, GoogLeNet, ResNet50)	1. AMI Ear Image Database (564 images from 94 subjects), 2. IIT Delhi Ear Database (Raw and segmented images)	- AMI Database: 99% accuracy (end-to-end), 99.45% (ensemble classifiers) - IIT Delhi Database: Best accuracy of 93.9% with AlexNet features reduced using PCA (95% variance).

Journal & Year	Technique Used	Dataset Used	Findings & Results
IET Biometrics, (2018)	CNNs with Spatial Pyramid Pooling (SPP) and Centre Loss	USTB-Helloear database (612,661 images from 1,104 subjects in uncontrolled conditions)	Ensemble procedure improved performance by about 1%; verification accuracy varied by input size (e.g., 97.4% for ensemble).
IJRASET, (2021)	Six-layer deep CNN architecture for ear biometrics	Ear image database divided into quarters for analysis	Performance metrics indicate effective feature extraction and classification, with various thresholds yielding different recall and precision rates.
International Journal of Environmental Research and Public Health, (2022)	Gabor filters for feature extraction combined with an ensemble of pre-trained deep CNN models (VGG19 and DenseNet161) Descriptor-based and deep-learning techniques	AMI (A dataset for ear recognition)	The proposed Gabor-based ensemble model achieved an accuracy of 74.63% in unconstrained conditions, outperforming state-of-the-art methods.
Neural Computing and Applications,	Descriptor-based and deep-learning techniques	Annotated Web Ears (AWE) and AWEx	DL models can match descriptor-based speed with proper hardware but require more training resources. Comprehensive evaluation of recognition models reveals strengths and weaknesses.
Machine Vision & Applications (2021)	Deep CNNs, LogDet divergence-based Metric Learning (LDMLT), Discriminant Correlation Analysis (DCA)	AWE (Academic Widespread Ear) and USTB II (University of Science and Technology of Beijing II)	The method demonstrates improved recognition rates, with ResNet showing better performance on the AWE database and VGG-S slightly outperforming others on the USTB II database.
Applied Sciences (2024)	DCGAN + Mean-CAM-CNN	MAI, AWE	The proposed approach outperforms existing methods, particularly on the **AWE** dataset, using **Mean-CAM** to focus on important features and **DCGAN** to enhance image quality. **MAI**: Rank-1 = 100%, **AWE**: Rank-1 = 76.25%
Evolving Systems (2023)	Ensemble of three lightweight CNN models	IITD-II dataset	The ensemble approach improves recognition accuracy and stability compared to individual models, achieving an accuracy of up to 96.83% with significant performance enhancement noted.

K-nearest neighbours, and support vector machines. Abdel-Hamid et al. performed an extensive survey on the topic of ear biometrics, focusing on the use of deep learning approaches, CNNs for ear recognition.

In this work, the authors survey ear recognition technologies together with 2D, 3D, and combined approaches, while classifying the existing approaches and databases. They give a chronological account of ear recognition development, surveying over 110 papers and 34 datasets from the late 1990s through early 2022. Accordingly, this paper describes the key components in the ear recognition system to include detection, normalization, feature extraction, and classification, while also placing focus on deep learning methods which have led to substantial improvements on difficult datasets. They provide a taxonomy to classify the types of approaches available and the strengths and limitations of each approach. They highlight the weaknesses of contemporary datasets, which were frequently gathered in controlled situations, and stress the necessity of strong, up-to-the-minute models for unrestricted contexts

They developed a new method for oral identification using a deep CNN utilizing additional triplet constraints from log-det divergence, Soubra et al. In order to improve ears images representation, they utilize pre-trained models, naturally they can extract discriminative features using VGG and ResNet on ear images. Specifically, we leverage DCA to perform a dimensionality reduction fusion of the last two layers of deep features, thus maintaining high class correlations as much as possible. The authors show that their approach yields effective recognition performance across several datasets include AWE and USTB II, performing better than existing methods in the literature. This research work significantly highlights the contribution of deep feature fusion and metric learning techniques in ear recognition.

The authors applied the DCGAN + Mean-CAM-CNN framework for ear recognition task on MAI and AWE data sets and compared these results against traditional techniques. The effect of preprocessing was found to improve recognition results for both datasets, especially enhancing Rank-1 recognition rates. Using the MAI dataset, the proposed method displayed Rank-1 recognition efficiency at 100% (perfect), while the corresponding metrics of other methods satisfied between ~73% and 99%. In terms of the AWE dataset, our method achieved 76.25% Rank-1, which is highly competitive due to the dataset having a more complex and unconstrained nature. The achieved performance compared very favourably against other deep learning and static approaches which, in many cases, failed to generalize well seeing that the AWE dataset was rather difficult resulting in very low recognition rates. Mean-CAM helped to visualize the CNN focus on the most important part of the image rather than considering the whole picture which could degrade performance due to irrelevant content. The outcomes also noted an increase in image quality where the dark spots of images were lighted and gray-scaled images were colorized. These innovations helped the DCGAN + Mean-CAM-CNN method perform better than nearly half of all the compared methods, emphasising its strong performance throughout all datasets with different levels of difficulty.

This paper[38] introduces an improved system for recognizing ears by creating three diverse lightweight CNN models and subsequently fusing them through both average and weighted average ensemble methods. A joint ensemble boosts the recognition accuracy significantly as follows computing a robust weight for each of the sub-models we obtain accuracy up-to 96.83%.

Table 1 presents a clear summary of the literature review, offering a handy reference for the key studies and methods discussed.

2.1 Key Contributions:

This paper aims to advance ear biometrics through several significant contributions, detailed in the following points:

1. **Development of Custom CNN Models:** Three CNN models specifically constructed for ear recognition are shown in this research. These models were created to efficiently record and comprehend the essential for precise identification. This guarantees that the models can accurately distinguish between different ear forms and characteristics.
2. **Training:** UERC and IIT-D are two distinct datasets that were used for ex- tensive training and testing of the models. Using this method, it was verified that the models functioned effectively with various ear image kinds. To get the best results with every dataset, adjustments were done during the training process.
3. **Use of Ensemble Learning:** Use of Ensemble Learning: Following training, the predictions made by each individual CNN model were combined using ensemble learning techniques. Weighted and average ensemble methods were used in the ensemble process. By carefully choosing each model's weight, the weighted ensemble technique increased performance by strengthening the integration of predictions.
4. **Comparison with Existing Results:** The results obtained from the IIT-D dataset were compared with those reported in[17]. This comparison highlights the effectiveness of the developed models and the enhancements achieved through the proposed techniques, acknowledging the valuable contributions of prior research in the field.

The rest of the paper is structured as follows: Section 3 details the experimental setup, including the datasets used, preprocessing steps, and model architectures. The training process and hyperparameter tuning are covered in Section

4. Section 5 presents the results and analysis, including metrics-based comparative evaluations of the proposed models and ensemble techniques. The implementation details for the ensemble approach, along with observations from different datasets, are provided in Section 6. Finally, Section 7 concludes the paper with key findings and suggests future research directions.

3. Experimental Setup

3.1 Dataset Used

All three models outlined earlier were trained on two datasets. Their individual performance was assessed first, followed by an evaluation of the ensemble learning results, providing a comprehensive overview of their effectiveness.

The IIT Delhi[39] ear image database consists of two versions. The original version had 471 photos from 121 individuals, each of whom contributed at least three ear images. It was compiled between October 2006 and June 2007. The JPEG formatted pictures, which were taken using a touchless setup, have a resolution of 272*204 pixels. Additionally, automatically cropped and normalized ear pictures (50*180 pixels) are included in this edition. This enlarged collection has 754 unique ear photos, which come from 221 different individuals in total. In contrast to the initial edition, this bigger dataset is distinguished by it is greater volume and diversity[39,40].

The UERC 2019[41] challenge employed three distinct datasets for ear recognition tasks. The primary dataset, Part A, contained 3,300 ear images from 330 unique individuals, with each subject represented by 10 images. This dataset was annotated with details such as occlusion levels, rotation angles, the presence of accessories, gender, and side information, facilitating the development of specialized recognition methods To ensure the fairness of evaluation used in the test dataset which contains 1,800 images and 1,500 images in the training dataset with the testing datasets not in any case used to train the image. In part B, 804 ear images taken from 16 individuals who are the same as part A are taken to train the algorithm further. The ability of the algorithm to scale up in part C is tested by employing 7,700 ear images to 3,360 different human beings. 1,500 images is used in training the same as part A images. The test datasets applied are 1,800 images from Part A.

The earVN1.0 dataset is a collection of ear images intended for the development and testing of ear identification algorithms. It has a total of 8,288 ear photos spread across 164 folders. These photos are in JPEG format and were acquired with a touchless setup to ensure consistency and quality for biometric recognition activities. The dataset is intended for use in the construction, training, and assessment of ear-based recognition algorithms, and it includes a wide range of photos from numerous persons.

The training and test datasets had to be kept apart, and the challenge organizers completely maintained this requirement. They could disqualify any team that used test images during training based on their final analysis. To ensure the evaluation process's integrity, the training and test datasets were carefully divided. In total, there were 2,304 images from 166 people used to train and evaluate the models[42].

3.2 Data Preprocessing & Augmentation Approaches

Data augmentation means making different versions of images by doing things like turning them, flipping them side to side, changing how bright or dark they are, correcting their brightness in a certain way, making them look blurry using a certain kind of filter, and making them bigger or smaller in a random way. These variations expose the model to different perspectives of each image, allowing it to learn more about various features, thereby enhancing its performance. In preparation for CNN model training, images from both the IIT Delhi (IIT-D) and UERC datasets were standardized to a size of 128*128 pixels, ensuring uniformity and consistency across the dataset. After standardization, data augmentation techniques were applied, including rotation limited to angles between -15 and 15 degrees, horizontal flipping, random changes to brightness and contrast, random gamma correction, Gaussian blur with a probability of 0.5, and random scaling with a scale limit between 0.4 and 0.9. After doing the augmentation of the IIT-D and UERC datasets 10 times, and the EARVN1.0 dataset 3 times, the class distributions for all three datasets are shown in Fig. 1 and 2.

After applying a train-test split ratio of 75:25, the dataset distribution for each dataset is summarized in the table 2.

Table 2. Dataset distribution after Train-test

Dataset	Split Ratio	Training Data	Test Data
IIT-D Dataset	75:25	6542	2181
UERC Dataset	75:25	19008	6336
EAR VN1.0 Dataset	75:25	24864	8288

3.3 Model Architectures

After looking at a lot of past research and trying out many different kinds of CNN models, some important points were found.

A rigorous model selection technique has been used to methodically determine the best customized Convolutional Neural Network (CNN) model architecture. In order to eliminate poor models and preserve high-performing designs that generalize well across datasets, the framework comprises a multi-stage evaluation procedure. Figure 3 shows how the process flows.

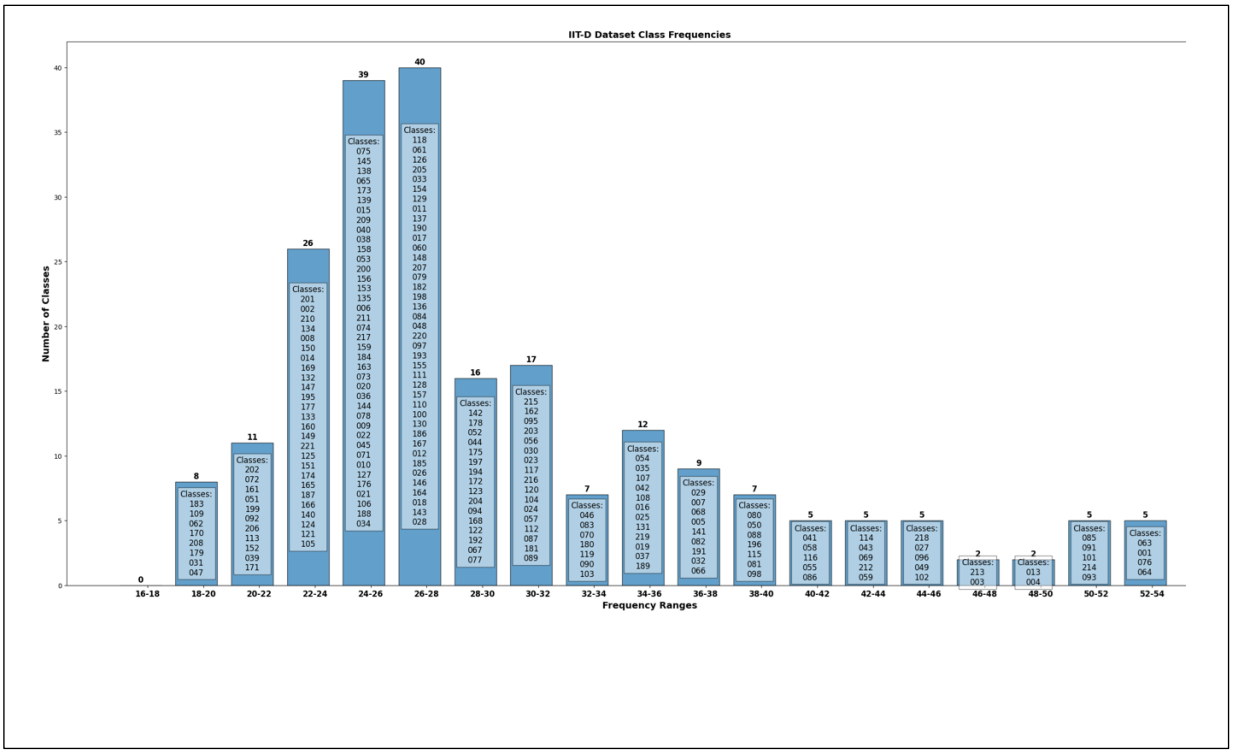

Fig. 1: Augmented Dataset Class Distribution for IIT-D dataset

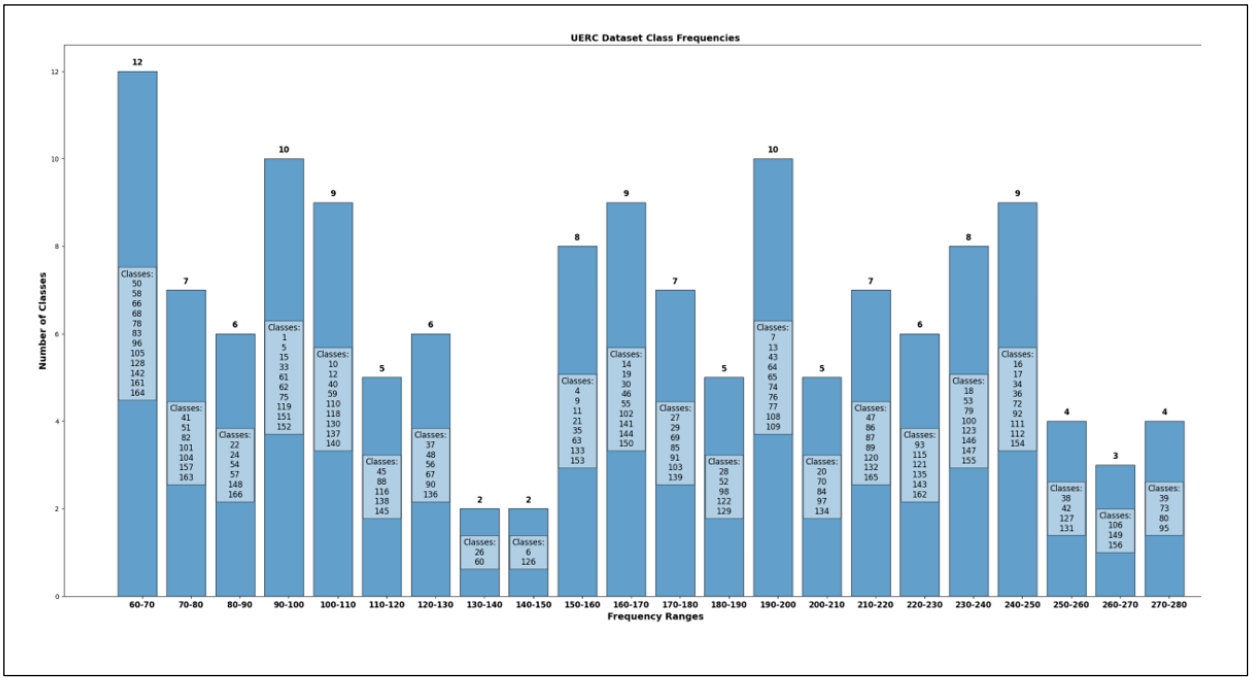

Fig. 2: Augmented Dataset Class Distribution for UERC dataset

Multiple CNN models are designed and prepared as the first step in the selection process. Numerous architectural arrangements and combinations of hyperparameters are investigated, such as: Convolutional Layer Variations**,** Pooling Techniques**,** Comparison of non-linear activation functions, including Sigmoid and ReLU, among others, Loss Functions, Selecting an Optimizer**:** Assessing optimizers (such as Adam, SGD, and RMSProp) to efficiently reduce model loss.

Utilizing a two-step evaluation, the framework first trains the models and evaluates on Dataset 1, only advancing those that pre-mature on the pre-set threshold to Phase 2, where the models are validated on Dataset 2 so that they ensure robustness and generalizability in the second stage of procedure, effectively mitigating the risk of overfitting across the board.

These experiments helped in understanding how changing the model can affect how well it learns and does its job. Based on these learnings, three CNN models are being proposed in this paper, each with its unique setup. These setups vary in terms of the number of hidden layers, convolution neurons, and dropout layers included. In this,

exploration, the main aim was to understand the delicacies of model behaviour - how these variations would impact their learning abilities, generalizing capacity, and effectiveness to tackle complex problems. By exploring the design choices in greater detail, the aim was to uncover the key factors distinguishing a good model from a great one. Figure 4 offers a comprehensive view of the diverse models developed for this study.

3.4 Hyperparameters

Key configuration parameters that guide the training process of a model are hyperparameters, including the learning rate, batch size, number of epochs, and optimizer selection. It is essential to ensure that hyperparameters are properly tuned to prevent problems such as delayed convergence, underfitting, or overfitting.

Careful consideration was given to the number of training sessions (epochs) to ensure the proposed models

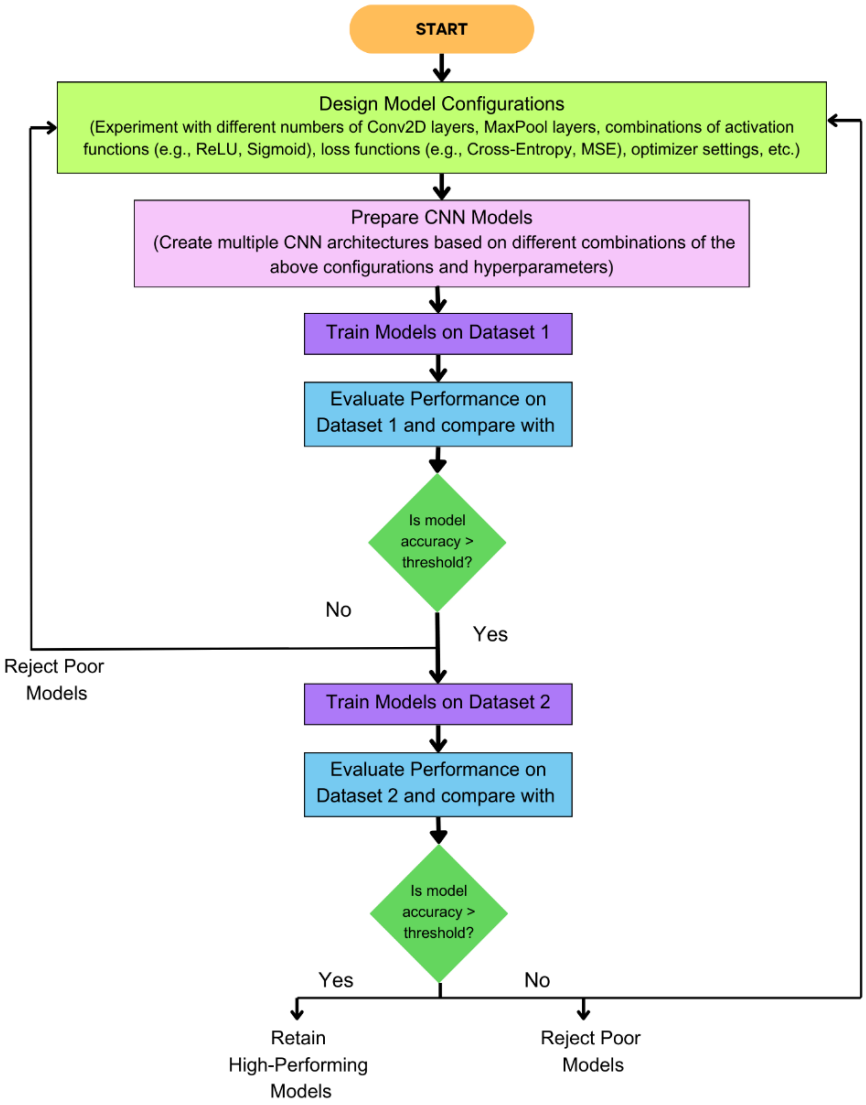

Fig. 3: Systematic model selection for CNN architectures.

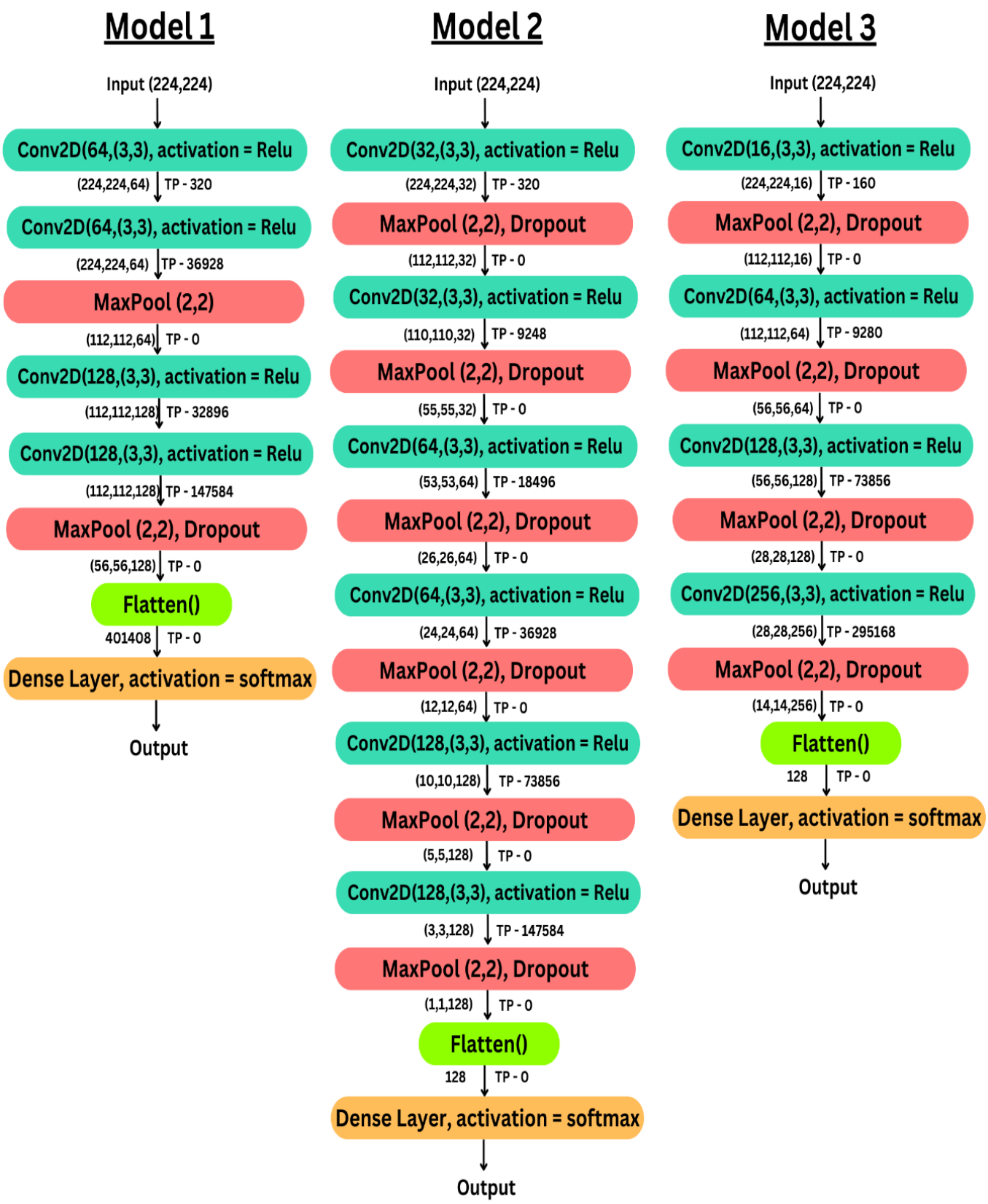

Fig. 4: Customized models

learned effectively without overfitting. After experimenting with various optimizers and activation functions, it was found that the Adam optimizer consistently yielded the best results. This optimizer dynamically adjusts. the learning rate for each parameter during training, facilitating faster convergence.

The Adam optimizer updates parameters using the following formulas:

$$m_t = \beta_1 m_{\{t-1\}} + (1 - \beta_1) g_t \quad (1)$$

$$v_t = \beta_2 v_{\{t-1\}} + (1 - \beta_2) g_t^2 \quad (2)$$

$$\widehat{m_t} = \{m_t\}/\{1 - \beta_1^t\} \quad (3)$$

$$\widehat{v_t} = v_t/1 - \beta_2^t \quad (4)$$

$$\theta_t = \theta_{\{t-1\}} - (\eta/\sqrt{v_t} + \epsilon)\widehat{m_t} \quad (5)$$

Where m_t and v_t are the first and second moment estimates, β_1 and β_2 are decay rates, g_t is the gradient at time step t, $\widehat{m_t}$ and $\widehat{v_t}$ are bias-corrected moment estimates, η is the learning rate, ϵ is a small constant to prevent division by zero, and θ_t is the parameter vector.

Categorical cross-entropy was chosen as the model performance evaluation metric because it works well for multi-class classification problems. The difference between the actual and anticipated class distributions is effectively measured by this loss function.

The categorical cross-entropy loss function is defined as:

$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) \quad (6)$$

where N is the number of classes, y_i is the actual label, and $\hat{y}_i$ is the predicted probability for class i.

To prevent the models from fixating too much on particular data patterns, dropout regularization was implemented with four dropout rates. The first technique we explored was dropout, which randomly sets a fraction of input units to 0 at each update during training time.

The dropout regularization can be expressed as:

$$h_i' = h_i . mask_i / p \quad (7)$$

where h_i is the input unit, $mask_i$ is a binary mask vector, and p is the probability of keeping a unit.

The other technique we used for improving performance was the fine-tuning of training with four different learning rates. Across all 16 combinations, we thoroughly analysed the effects of these hyperparameters on training and reported the results in the paper, demonstrating how they influenced model convergence and performance.

3.5 Evaluation Metrices

Evaluation of each model's predictions was done after conducting an experiment with all the other classifiers. Further, all the predictions were combined with the average ensemble approach to see if it would have an effect on the results. The results from the ensemble were then evaluated accordingly to the individual predictions.

A weighted ensemble approach has been taken while adjusting the parameters to reflect individual model predictions accurately.

The experiment's results were evaluated with three key metrics: accuracy, precision, and the F1 score. The corresponding results were analysed thoroughly to evaluate the performance of each of the models in question.

3.6 Software & Hardware

Python was used as the programming language and the TensorFlow and Keras frameworks to create deep learning models. To enhance the models' capacity to generalize and identify patterns in novel, unseen data, the Albumentations library[43] was utilized to intentionally expand the dataset size by randomly transforming the photos through flips, rotations, and brightness modifications. The training was held on a machine that had a powerful 16-core, 12-generation Intel Core i9 processor with 64GB of RAM and an NVIDIA Tesla T4 GPU with 16GB of RAM to speed up the procedure. Model performance increased as a result of the training process's optimization for effective resource use.

4. Training

This section describes the CNN models constructed for the current investigation, including their training setups and outcomes. Using previously published model architectures, preprocessing procedures, and data augmentation strategies, training and validation were carried out on two datasets: IIT-D and UERC.

4.1. Training on IIT-D Dataset

To find the best parameters for the suggested CNN models, a thorough hyperparameter search was done on the IIT-D dataset.

The models were trained using four dropout rates (0.1, 0.2, 0.3, 0.5) and four learning rates (0.005, 0.001, 0.0005, 0.0001), resulting in 16 combinations. The training process was evaluated based on four key metrics: training accuracy, training loss, validation accuracy, and validation loss. All 16 combinations were systematically trained, and the results for each case, including the mentioned metrics, are detailed in Table 3.

After tuning the hyperparameters, the following configurations were determined as optimal based on the highest validation accuracy achieved during the training process:

- Model 1: Dropout rate of 0.1 and learning rate of

0.001.
- Model 2: Dropout rate of 0.2 and learning rate of 0.001.
- Model 3: Dropout rate of 0.1 and learning rate of 0.0001.

4.2. Training on UERC Dataset

Similar to the process followed for the IIT-D dataset, all hyperparameters of the UERC dataset were thoroughly tuned. To determine the optimal dropout rates and learning rates for different CNN model configurations, the same range of dropout rates and learning rates were tested. The results are shown in Table 4.

The optimal hyperparameters for the UERC dataset were identified based on the highest validation accuracy, ensuring that each model was finely tuned for the dataset are:

- Model 1: Dropout rate of 0.5 and learning rate of 0.001.
- Model 2: Dropout rate of 0.2 and learning rate of 0.005.
- Model 3: Dropout rate of 0.5 and learning rate of 0.0001.

Similarly results for EAR VN 1.0 are shown in table 5

5. Results & Analysis

5.1 Metrics-Based Comparative Analysis of IIT-D Dataset

In this analysis, the performance of three machine learning models (Model 1, Model 2, and Model 3) trained with different Dropout Rates (DR) and learning rates (LR) on the IIT-D test dataset is evaluated. The dropout strategy involves randomly removing neurons from a training set in order to prevent overfitting.

5.1.1 Accuracy

When it comes to lower Dropout Rates (DR) of 0.1 and 0.3, Model 3 constantly delivers the maximum accuracy. Table 6 shows its accuracy is persistently higher than 0.8 for all DR values, demonstrating an adequate ability of recognizing with precision. When the Dropout Rate rises, Model 1's accuracy slightly increases and ranges from 0.759 to 0.845. The significant drop in accuracy for Model 2 at higher Dropout Rates suggests it might be overfitting the data at lower Dropout Rates.

5.1.2 Precision

All models exhibit a similar trend in precision. As the Dropout Rate increases, the precision values generally go up as shown in Table 6. This suggests that with more data being dropped out during training, the models become better at identifying true positives but might miss some relevant instances. Model 3 maintains a slight edge in precision over the other models for most Dropout Rates. Model 1 and Model 2 have comparable precision scores.

5.1.3 F1-Score

Table 6 represents the comparisons of F1 scores across different configurations, Model 1 consistently exhibits robust performance, ranging from 0.939 to 0.965. Comparatively, Model 2 displays a broader range of F1 scores, which mirror its accuracy and precision trends. As for Model 3, its F1-score values are very close to those of both Model 1 and Model 2, with some variability depending on the hyperparameter settings. Compared with its counterparts, Model 2 and Model 3, Model 1 shows better F1-score performance and is more consistent.

5.1.4 Implications

Based on various hyperparameter settings, Model 1 is generally more accurate, precise, and has a higher F1-score than Model 2 and Model 3. There is greater variability in Model 2, whereas Model 3 has a comparable performance to Model 1, but slightly lower stability in some cases.

5.2 Metrics-Based Comparative Analysis of UERC Dataset

This study compares the effectiveness of three machine learning models (Models 1, 2, and 3), which were trained with varying Dropout Rates (DR) and Learning Rates (LR), on the UERC test dataset.

5.2.1 Accuracy

Table 7 shows when Dropout Rates (DR) range from 0.1 to 0.5, Model 3 consistently obtains the maximum accuracy. Its great ability to recognize effectively is demonstrated by the fact that its accuracy stays above 0.88 for all DR levels.

But Model 2 exhibits a drop in accuracy as it was for IIT-D dataset also, especially at higher Dropout Rates, indicating a possible overfitting issue. Between 0.759 and 0.845, Model 1 shows a small improvement in accuracy with increasing Dropout Rate.

5.2.2 Precision

A similar pattern of precision is evident when the dropout Rate increases, with the level of precision increasing as the rate rises as shown in Table 7. Model 3 maintains a slight edge in precision over the others for most Dropout Rates, indicating its effectiveness in identifying true positives. Model 1 and Model 2 show comparable precision scores across different DR configurations.

Table 3. Dropout rate and learning rate analysis for IIT-D dataset.

(a)Model 1

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.995	2.664	0.937	0.709	1	0.0001	0.945	0.448	0.993	0.028	0.9211	0.578	0.992	0.0314	0.945	0.35
0.001	1	0.001	0.946	0.491	0.994	0.023	0.941	0.476	0.996	0.016	0.932	0.51	0.997	0.012	0.952	0.373
0.0005	0.999	0.001	0.94	0.532	1	0.648	0.944	0.436	1	7.84	0.945	0.386	0.996	0.014	0.948	0.3414
0.0001	1	3.899	0.938	0.717	0.997	0.0126	0.927	0.652	0.993	0.026	0.932	0.493	0.993	0.028	0.938	0.399

(b)Model 2

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.992	0.0306	0.926	0.368	0.984	0.0516	0.928	0.338	0.967	0.107	0.83	0.762	0.962	0.168	0.717	3.341
0.001	0.991	0.028	0.948	0.393	0.983	0.052	0.953	0.245	0.969	0.107	0.925	0.288	0.964	0.221	0.752	3.995
0.0005	0.992	0.027	0.937	0.423	0.985	0.0504	0.925	0.392	0.968	0.108	0.915	0.374	0.962	0.172	0.672	3.129
0.0001	0.982	0.646	0.938	0.339	0.986	0.047	0.95	0.263	0.964	0.116	0.804	0.745	0.962	0.224	0.767	2.796

(c)Model 3

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	1	1.814	0.934	0.566	0.996	0.0116	0.919	0.543	0.996	0.016	0.924	0.4325	0.992	0.033	0.651	1.696
0.001	1	2.575	0.937	0.5012	0.996	0.014	0.927	0.506	0.998	0.007	0.805	1.17	0.991	0.39	0.771	1.146
0.0005	0.998	0.005	0.927	0.514	0.016	0.995	0.907	0.595	0.99	0.039	0.77	1.145	0.987	0.0512	0.909	0.492
0.0001	1	0.001	0.937	0.497	1	4.294	0.941	0.498	0.993	0.039	0.838	0.886	0.985	0.049	0.828	0.841

Table 4. Dropout rate and learning rate analysis for UERC dataset.

(a)Model 1

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.995	0.019	0.767	2.591	0.998	0.005	0.776	2.557	0.996	0.013	0.779	2.567	0.996	0.014	0.789	2.245
0.001	0.996	0.015	0.704	2.949	0.997	0.011	0.761	2.748	0.012	0.997	0.781	2.358	0.996	0.014	0.802	2.003
0.0005	0.998	0.006	0.791	2.521	0.997	0.0134	0.748	2.694	0.994	0.302	0.728	2.835	0.997	0.012	0.793	2.073
0.0001	0.998	0.009	0.768	2.539	0.996	0.015	0.76	3.319	0.995	0.018	0.799	2.329	0.995	0.018	0.798	1.878

(b)Model 2

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.972	0.093	0.892	0.729	0.924	0.263	0.904	0.535	0.795	0.706	0.854	0.644	0.904	0.895	0.845	2.345
0.001	0.975	0.082	0.896	0.693	0.923	0.261	0.897	0.568	0.805	0.682	0.842	0.6758	0.888	1.111	0.767	5.029
0.0005	0.974	0.082	0.905	0.681	0.921	0.263	0.891	0.575	0.799	0.684	0.85	0.646	0.9421	0.2567	0.827	1.8523
0.0001	0.975	0.085	0.898	0.714	0.917	0.277	0.892	0.583	0.799	0.684	0.852	0.629	0.947	0.162	0.839	9.11

(c)Model 3

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.994	0.021	0.87	1.161	0.998	0.006	0.886	1.105	0.995	0.016	0.903	0.799	0.989	0.034	0.9	0.643
0.001	0.995	0.017	0.889	0.946	0.997	0.01	0.891	1.052	0.995	0.018	0.902	0.797	0.992	0.024	0.9	0.705
0.0005	0.997	0.007	0.878	1.59	0.998	0.006	0.889	1.03	0.997	0.009	0.898	0.754	0.989	0.034	0.901	0.585
0.0001	0.996	0.015	0.895	0.927	0.997	0.01	0.892	0.913	0.995	0.016	0.904	0.748	0.99	0.003	0.901	0.649

Table 5: Dropout rate and learning rate analysis for EAR VN1.0 dataset.

(a)Model 1

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.944	1.368	0.85	2.161	0.948	1.96	0.853	2.105	0.935	1.916	0.853	2.399	0.959	2.034	0.844	1.643
0.001	0.935	1.425	0.849	1.946	0.947	1.91	0.852	2.052	0.945	2.018	0.849	2.297	0.942	2.024	0.837	1.705
0.0005	0.947	1.408	0.838	1.759	0.958	1.986	0.859	2.03	0.947	2.009	0.851	1.754	0.939	2.034	0.838	2.584
0.0001	0.936	1.398	0.805	2.027	0.937	1.941	0.852	1.913	0.935	2.016	0.862	2.448	0.929	2.003	0.824	2.641

(b)Model 2

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.941	2.318	0.825	2.087	0.940	1.951	0.841	2.102	.945	1.906	0.843	2.241	0.909	2.140	0.834	1.623
0.001	0.938	2.405	0.817	1.854	0.939	1.909	0.842	2.054	0.935	2.038	0.839	2.314	0.942	2.047	0.827	1.675
0.0005	0.933	2.298	0.802	1.755	0.947	1.964	0.849	2.102	0.937	2.019	0.840	1.842	0.939	2.120	0.831	2.504
0.0001	0.932	1.908	0.841	2.121	0.934	1.944	0.848	1.943	0.938	2.026	0.834	2.438	0.949	2.324	0.834	2.614

(c)Model 3

lr	Dropout 0.1				Dropout 0.2				Dropout 0.3				Dropout 0.5			
	Train		Val		Train		Val		Train		Val		Train		Val	
	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
0.005	0.954	2.085	0.835	1.987	0.942	1.962	0.842	2.122	0.935	1.906	0.842	2.041	0.937	1.940	0.824	1.423
0.001	0.936	2.056	0.847	1.954	0.937	1.941	0.837	2.034	0.932	2.018	0.840	2.014	0.932	2.033	0.847	1.473
0.0005	0.948	2.077	0.832	1.845	0.946	2.044	0.839	2.142	0.947	2.121	0.836	1.542	0.935	2.114	0.834	2.424
0.0001	0.929	2.106	0.831	2.095	0.931	1.994	0.848	1.943	0.941	1.926	0.833	2.236	0.947	2.334	0.836	2.315

Table 6. Performance evaluation of custom models with different dropout rates and learning rates on IIT-D test dataset.

	ACC Model 1	ACC Model 2	ACC Model 3	Precision Model 1	Precision Model 2	Precision Model 3	F1-Score Model 1	F1-Score Model 2	F1-Score Model 3
DR 0.1 LR 0.001	0.971	0.952	0.961	0.974	0.962	0.966	0.969	0.952	0.96
DR 0.1 LR 0.005	0.958	0.949	0.954	0.964	0.959	0.959	0.958	0.95	0.953
DR 0.1 LR 0.0001	0.948	0.945	0.973	0.958	0.953	0.977	0.948	0.944	0.972
DR 0.1 LR 0.0005	0.957	0.959	0.9495	0.965	0.966	0.957	0.958	0.959	0.948
DR 0.2 LR 0.001	0.953	0.969	0.946	0.931	0.974	0.956	0.952	0.968	0.946
DR 0.2 LR 0.005	0.956	0.942	0.9403	0.962	0.953	0.952	0.956	0.942	0.94
DR 0.2 LR 0.0001	0.939	0.953	0.965	0.951	0.961	0.972	0.939	0.953	0.965
DR 0.2 LR 0.0005	0.962	0.946	0.938	0.968	0.958	0.951	0.961	0.946	0.938
DR 0.3 LR 0.001	0.951	0.946	0.832	0.962	0.959	0.878	0.952	0.946	0.831
DR 0.3 LR 0.005	0.953	0.943	0.829	0.964	0.956	0.876	0.953	0.944	0.829
DR 0.3 LR 0.0001	0.965	0.934	0.944	0.971	0.951	0.954	0.965	0.934	0.943
DR 0.3 LR 0.0005	0.963	0.839	0.95	0.968	0.907	0.959	0.963	0.843	0.951
DR 0.5 LR 0.001	0.959	0.7391	0.941	0.968	0.8353	0.951	0.959	0.7394	0.941
DR 0.5 LR 0.005	0.962	0.7197	0.819	0.968	0.7706	0.874	0.962	0.7211	0.813
DR 0.5 LR 0.0001	0.961	0.7675	0.707	0.966	0.8546	0.86	0.96	0.7754	0.724
DR 0.5 LR 0.0005	0.953	0.6931	0.865	0.962	0.7537	0.912	0.952	0.6893	0.868

Table 7. Performance evaluation of custom models with different dropout rates and learning rates on UERC test dataset.

	ACC Model 1	ACC Model 2	ACC Model 3	Precision Model 1	Precision Model 2	Precision Model 3	F1-Score Model 1	F1-Score Model 2	F1-Score Model 3
DR 0.1 LR 0.001	0.759	0.905	0.905	0.788	0.912	0.901	0.76	0.905	0.891
DR 0.1 LR 0.005	0.786	0.897	0.896	0.817	0.903	0.898	0.79	0.896	0.889
DR 0.1 LR 0.0001	0.808	0.901	0.901	0.831	0.908	0.911	0.807	0.901	0.905
DR 0.1 LR 0.0005	0.829	0.907	0.906	0.846	0.912	0.904	0.83	0.906	0.89
DR 0.2 LR 0.001	0.8	0.91	0.909	0.842	0.917	0.906	0.811	0.909	0.898
DR 0.2 LR 0.005	0.829	0.911	0.91	0.845	0.916	0.908	0.83	0.91	0.899
DR 0.2 LR 0.0001	0.811	0.907	0.907	0.851	0.912	0.904	0.82	0.907	0.897
DR 0.2 LR 0.0005	0.799	0.905	0.905	0.819	0.911	0.911	0.801	0.905	0.903
DR 0.3 LR 0.001	0.813	0.874	0.875	0.846	0.889	0.916	0.818	0.875	0.909
DR 0.3 LR 0.005	0.807	0.881	0.88	0.828	0.891	0.908	0.808	0.88	0.902
DR 0.3 LR 0.0001	0.817	0.888	0.887	0.841	0.898	0.913	0.818	0.887	0.905
DR 0.3 LR 0.0005	0.762	0.882	0.883	0.804	0.895	0.912	0.764	0.883	0.904
DR 0.5 LR 0.001	0.839	0.833	0.836	0.858	0.869	0.916	0.841	0.836	0.909
DR 0.5 LR 0.005	0.84	0.852	0.839	0.863	0.869	0.919	0.841	0.839	0.913
DR 0.5 LR 0.0001	0.838	0.861	0.862	0.855	0.879	0.923	0.839	0.862	0.917
DR 0.5 LR 0.0005	0.845	0.839	0.844	0.86	0.876	0.92	0.846	0.844	0.915

5.2.3 F1-Score

It is clear from the Table 7 Model 1 consistently shows robust performance in F1- score, ranging from 0.811 to 0.965 across different configurations. F1 scores for Model 2 are more diverse, reflecting its accuracy and precision trends. The F1-score values for Model 3 are closely aligned with those of Models 1 and 2, with some variations depending on hyperparameter settings. In general, Model 1 performs better and more consistently than Models 2 and 3.

5.2.4 Implications

Model 1 do better than Model 2 and Model 3 in terms of accuracy, precision, and F1-score on the UERC dataset when different hyperparameter values are taken into account. Model 2 has greater unpredictability, while Model 3 performs similarly to Model 1, but occasionally with somewhat less consistency.

Similarly, results for test data of EAR VN1.0 is presented in Table 8.

5.3 Evaluation of all 3 models based on training & test results

This study evaluates the designs and effectiveness of three distinct CNNs in the task of ear identification. A meticulous analysis of each model was conducted to determine the optimal hyperparameter configurations, including systematically varied dropout rates and learning rates. The detailed comparison of these three custom models is provided in Table 7.

5.3.1 Model 1

Model 1, comprising over 79 million trainable parameters, operates as a robust CNN designed for detailed image analysis. It decomposes visual input into four layers, extracting fine features such as textures, edges, and spatial relationships. Then, the model compresses this data, concentrating it on the most important details to split each examined image into one of a previously learned. It is also significant that the designed model strives to reflect numerous subtleties since it has to be useful for tasks that require distinguishing images with extremely high precision.

5.3.2 Model 2

Model 2 uses a sequential neural network architecture that is designed for advanced image processing. Its design consists of a sequence of convolutional layers, max-pooling, and dropout layers, it progressively reduces spatial dimensions to enhance feature recognition. To be more exact, this model has over 9 million adjustable parameters. Overall, this model is designed to be quite complex since such complexity is essential for increased precision of patterns and data classification. In Model 1, enhancements like the integration of Squeeze-and-Excitation Blocks, which intelligently recalibrate channel-wise features, and iterative utilization of spatial measurements are well-suited to improve feature detection across various levels.

5.3.3 Model 3

Model 3 is a CNNs using a sequential neural network which has the dropouts, max-pooling's, and convolutional layers, which are incorporated into layers of CNN specifically useful for interpreting complex images. This architecture is carefully tuned to decrease the spatial dimension. At its architecture level of learning specifics, dimension reduction is beneficial for the model as it improves the ability to recognize fine details. The dense layer contains over 1.2 million parameters, which are traditionally trained deliberately assist in ensuring that the model properly understand the visual output and correctly

classify potential identities. It is important to note that compared to earlier models, Model 3 has a number of deep and more parameters for convolutional layers, which helps it enhance model representation features. This represents a specific change in the field which indicates that more complex architectures produce more accurate output in the ear identification task.

To sum up, one should note that each model looks at ear identification quite differently. Model 1's approach is through-in-depth image examination. Model 2, in turn, focuses on extensive image processing. Further, Model 3 deals with intricate image understanding. By using the benefits of these novel ensemble learning approaches in the study, it is possible to enhance the quality of the results in ear identification. In addition, one may note that the number of convolutional layers and the effect of image size on the layers and the total number of trainable parameters at the layer has also been analysed. Hence, it is possible to gain a deeper insight into the essence of models. Similarly, a description of the behaviour of each model has been provided in Table 9.

6. Ensemble Techniques

Ensembling is a method that combines the best features of multiple models by using their predictions, and typically it is way more useful than only a single model. The most prominent benefit is that ensembling opportunities meaningfully enhance generalization and reduce overfitting, because they impartially balance the biases and variances of individual models. Other benefits encompass enhanced robustness, increased stability, and capacity to deal with intricate data patterns. To implement ensembling, multiple models' predictions are put into two prevalent strategies – average ensembling and weighted average ensembling. The former strategy calculates an average of all models' predictions, which smoothes the influence of outliers. The latter method attunes a relative weight to models' predictions relative to their performance, which is the mechanism that enables superior models to have a slightly larger influence.

These methods are straightforward yet effective, enhancing the accuracy and resilience of the predictions for the IIT-D datasets.

6.1 Average Ensembling

The Average Ensembling technique involves averaging the predictions from multiple models to combine them. The predictions made by each model are given equal weight, which helps to offset any peculiarities or inaccuracies.

$$P_{avg} = \frac{1}{n}\sum_{i=1}^{n} P_i \tag{8}$$

6.2 Weighted Average Ensembling

Using the Weighted Average Ensembling technique, models are given varying weights according to their prior performance. A weighted sum of the predictions from each model is the final prediction.

$$P_{wavg} = \sum_{i=1}^{n} w_i P_i \tag{9}$$

Here, w_i represents the weight of the *i*-th model and the sum of all weights is 1.

Ensembling is a powerful technique in machine learning which improves prediction accuracy by combining the strengths of multiple models. It helps to lessen individual model weaknesses, resulting in more robust and reliable outcomes.

Figure 5 provides flowchart illustrating a generic ensembling workflow. Preprocessed and augmented input datasets provide better quality and variety of data. This augmented larger dataset is used to train multiple models (Model 1, Model 2, up to Model n). Each trained model provides its own set of predictions. These predictions are then combined using two methods: an average ensemble and a weighted average ensemble. The final predictions for the IIT-D datasets are derived from these combined methods, improving overall prediction performance.

7. Implementation for ensembled approach-based model

For the IIT-D dataset, the optimal hyperparameter configurations based on validation accuracy are as follows:

- Model 1: Dropout rate of 0.1 & learning rate of 0.001.
- Model 2: Dropout rate of 0.2 & learning rate of 0.001.
- Model 3: Dropout rate of 0.1 & learning rate of 0.0001.

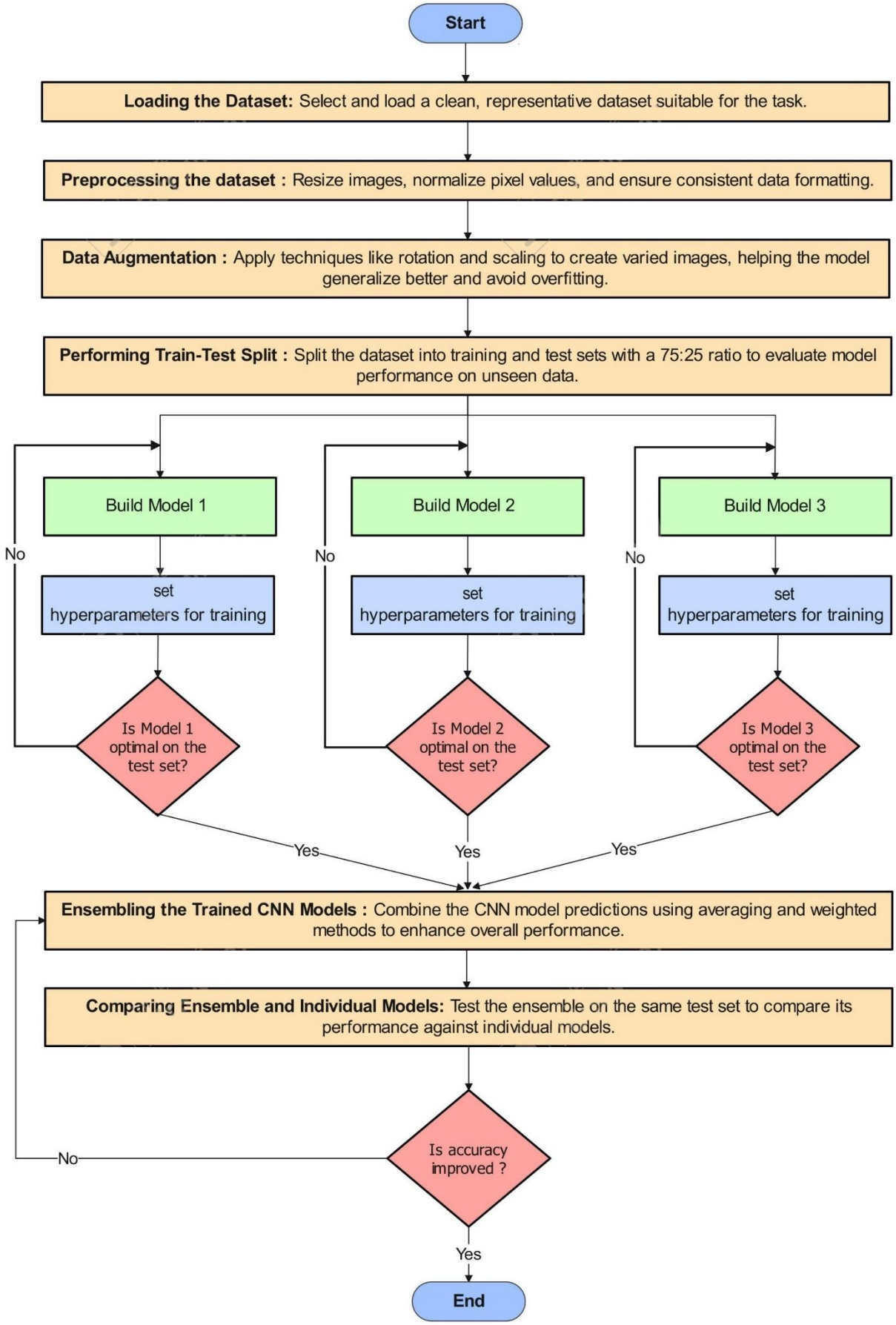

Fig. 5: Ensemble workflow

Table 8. Performance evaluation of custom models with different dropout rates and learning rates on EAR VN 1.0 test dataset.

	ACC Model 1	ACC Model 2	ACC Model 3	Precision Model 1	Precision Model 2	Precision Model 3	F1-Score Model 1	F1-Score Model 2	F1-Score Model 3
DR 0.1 LR 0.001	0.873	0.842	0.843	0.829	0.843	0.872	0.834	0.867	0.849
DR 0.1 LR 0.005	0.858	0.847	0.868	0.838	0.838	0.867	0.865	0.834	0.867
DR 0.1 LR 0.0001	0.898	0.834	0.878	0.840	0.884	0.874	0.874	0.845	0.849
DR 0.1 LR 0.0005	0.857	0.846	0.847	0.847	0.834	0.886	0.867	0.874	0.849
DR 0.2 LR 0.001	0.851	0.845	0.831	0.845	0.846	0.875	0.866	0.861	0.841
DR 0.2 LR 0.005	0.852	0.874	0.837	0.867	0.867	0.864	0.849	0.866	0.851
DR 0.2 LR 0.0001	0.833	0.865	0.834	0.859	0.849	0.875	0.877	0.854	0.870
DR 0.2 LR 0.0005	0.859	0.861	0.841	0.857	0.860	0.851	0.855	0.834	0.859
DR 0.3 LR 0.001	0.875	0.837	0.850	0.849	0.854	0.874	0.871	0.861	0.867
DR 0.3 LR 0.005	0.853	0.846	0.847	0.853	0.848	0.880	0.867	0.857	0.872
DR 0.3 LR 0.0001	0.863	0.837	0.837	0.856	0.838	0.874	0.872	0.864	0.856
DR 0.3 LR 0.0005	0.843	0.874	0.836	0.864	0.861	0.867	0.864	0.872	0.871
DR 0.5 LR 0.001	0.839	0.864	0.847	0.872	0.863	0.879	0.847	0.854	0.859
DR 0.5 LR 0.005	0.842	0.871	0.836	0.868	0.872	0.876	0.861	0.849	0.849
DR 0.5 LR 0.0001	0.841	0.857	0.842	0.858	0.876	0.869	0.839	0.871	0.839
DR 0.5 LR 0.0005	0.842	0.848	0.847	0.861	0.869	0.872	0.847	0.837	0.874

Table 9. Comparison of custom models.

Feature	Model 1 (High Complexity)	Model 2 (Balanced)	Model 3 (Efficient)
Layers	4 Conv, 2 Max Pool, 1 Dense	6 Conv, 6 Max Pool, 5 Drop, 1 Dense	4 Conv, 4 Max Pool, 3 Drop, 1 Dense
Filters	64, 64, 128, 128 (increasing)	32, 32, 64, 64, 128, 128 (balanced)	16, 64, 128, 256 (gradual)
Parameters	79,679,453 (high)	9,099,660 (moderate)	1,216,845 (low)
Focus	Intricate details, high precision	Balance of detail	Key features, efficiency
Strengths	Potential for high Accuracy	Versatile, adaptable	Resource-friendly, less overfitting
Weaknesses	Prone to overfitting, high cost	May not excel in extremes	May miss subtle patterns
Training Time	Longest	Moderate	Fastest
Inference Time	Slowest	Moderate	Fastest
Memory Requirements	Highest	Moderate	Lowest
Model Depth	Deep	Moderate Depth	Shallow
Feature Extraction	Fine-grained detail	Balanced Depth	Key features

The configurations were determined to be optimal among 16 distinct combinations of dropout rates and learning rates evaluated for each model. Figure 6 illustrates the performance of the three models, highlighting their accuracy, precision, and F1-score for each ideal configuration.

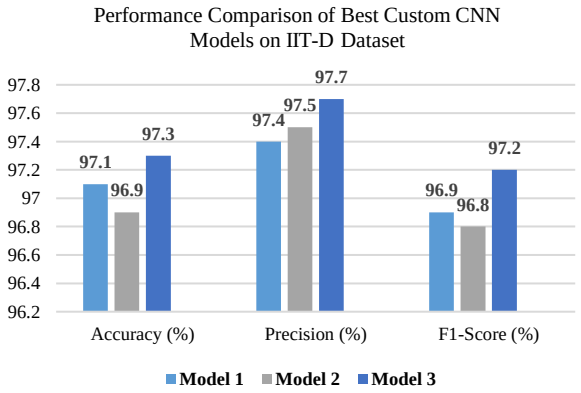

Fig. 6: Performance comparison of best custom CNN models on IIT-D dataset.

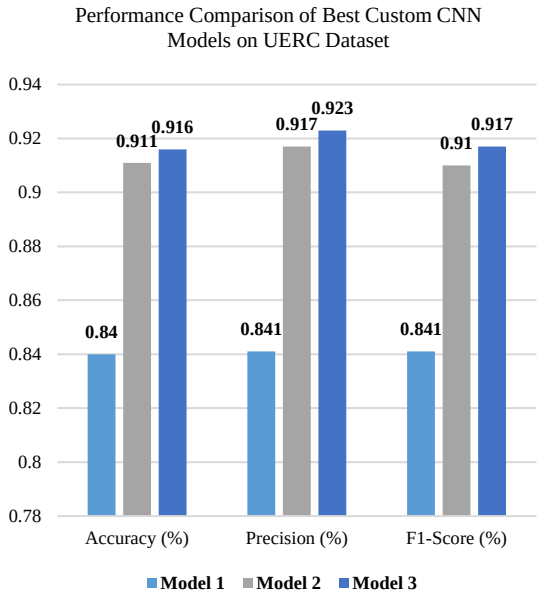

Fig. 7: Performance comparison of best custom CNN models on UERC dataset.

and, for UERC dataset, optimal hyperparameter configurations are:

- Model 1: Dropout rate of 0.5 & learning rate of 0.001.
- Model 2: Dropout rate of 0.2 & learning rate of 0.005.
- Model 3: Dropout rate of 0.5 & learning rate of 0.0001

Figure 7 displays the performance of the three models on the UERC dataset, emphasizing their accuracy, precision, and F1-score for each optimal configuration.

7.1 Implementation of Average Ensembled Model

The average ensemble model achieved an accuracy of 97.80%, a precision of 98.09%, and an F1-score of 97.25% for IIT-D Dataset. These results indicate even higher performance compared to the UERC dataset, showing that the ensemble model was particularly effective in this context.

For the UERC, the average ensemble model dataset achieved an accuracy - 95.20%, a precision - 95.41%, and an F1-score - 95.19%. This shows strong performance across all metrics, indicating that the ensemble model performed better than any single custom model in predicting outcomes for this dataset as shown in Table 10.

Table 10. Average ensemble results comparison.

Dataset	Accuracy	Precision	F-1 Score
IIT-D	97.80%	98.09%	97.25%
UERC	95.20%	95.41%	95.19%

To assign the weights to the three models, grid search was employed to identify the optimal combination that would yield the best Accuracy, Precision, and F1-Score.

Grid search is a technique that performs an exhaustive search and finds the most optimal hyperparameters by testing all possible combinations in the pre-defined parameters grid. What it does is a gritty optimization over all combinations, while providing the reproducible result, making it effective in finding the best performing configuration in static parametric space.

Different combinations of weights were applied to the models and the computed outputs were tested on the test datasets. For each tested combination, accuracy, precision, and F1-score performance measures were recorded. This was done repeatedly to cover all the combinations of weights, and the weights were then ranked in descending order based on the performances obtained.

The best 10 combinations according to the results of the grid search are listed in the table9 for IIT-D dataset and in table 12 for UERC dataset and the process is shown in Fig. 8.

7.2 Implementation of Weighted Average Ensembled Model

For improved results, a combination of the proposed three custom models was utilized. Various combinations of weights for each model were experimented with to determine the optimal setup.

7.2.1 Observations for IIT-D Dataset

Table 11. Performance comparison of different weighted Ensemble configurations for IIT-D dataset.

Weights Assigned to Model 1, 2 & 3	Accuracy	Precision	F1-Score
(0.4, 0.2, 0.4)	**0.99000120**	**0.99370823**	**0.989943086**
(0.3, 0.5, 0.2)	0.989912879	0.991235418	0.989821965
(0.2, 0.4, 0.4)	0.989454379	0.990671896	0.989454379
(0.3, 0.4, 0.3)	0.988995873	0.990429998	0.988903181
(0.3, 0.2, 0.5)	0.988754208	0.990155841	0.988846593
(0.2, 0.5, 0.3)	0.988537368	0.990080775	0.988468382
(0.4, 0.3, 0.3)	0.988078863	0.989495383	0.98796316
(0.3, 0.3, 0.4)	0.987620358	0.989149685	0.987528061
(0.2, 0.3, 0.5)	0.987161852	0.988726176	0.987045269
(0.5, 0.3, 0.2)	0.983439810	0.985738692	0.983232999

7.2.1.1 Best Combination

The setup with weights (0.4, 0.2, 0.4) worked the best. This combination gave us the highest accuracy score of 99%, precision of 99.37% and an F1-score of 98.99%. Here, it is clear how the complementary strengths of Model 1 and Model 3 in precision and the contribution of Model 2, led to an overall increase in the accuracy-evaluated performance metrics. This technique allows for an optimal distribution across different weights, achieving a harmonious balance between accuracy and precision, thus augmenting model efficacy in practical applications.

7.2.1.2 Trends in Model Weighting

Results showed that more weight to Model 2 or a balanced weight betwixt Model 2 and Model 3 usually hit better than Model 1. This implies that when leveraging the complementary properties of each model (accuracy vs recall) we achieve better predictive performance.

7.2.1.3 Further Insights

Although the results from Table 1 are relatively similar, it is clear from further examination that there are slight variances among each combination of weights. Indeed, it can be seen that the configurations giving more weight to Models 1 and 3 tend to report better results, in terms of higher precision and F1-score, consistently. Hence, the insights suggest that it is important to adjust the weightings for each model and their relative importance, based on their individual strengths and relevant factors for the task.

7.2.2 Observations for UERC Dataset

7.2.2.1 Best Combination

The best combination is the weights, having 0.3, 0.4, and 0.3. This is because it was able to give Model 2 40% while Model 1 and Model 3 given weights of 30%. This was able to give a gain in accuracy of 97.79%, in precision of 97.86%, and 97.78% in f1-score. This shows that having Model 2 more important and the other models equal in the model was able to really help.

7.2.2.2 Trends in Model Weighting

From the analysis, configurations that balanced the weights between Model 2 and the other models tended to give better results compared to configurations that focused too much on one model. For instance, in the setup (0.3, 0.4, 0.3), having a good balance between Model 1, Model 2, and Model 3 led to higher precision and F1-score, showing their combined strengths.

Table 12. Performance comparison of different weighted ensemble configurations for UERC dataset.

Weights Assigned to Model 1, 2 & 3	Accuracy	Precision	F1-Score
(0.3, 0.4, 0.3)	**0.97790404**	**0.978577429**	**0.977772371**
(0.2, 0.4, 0.4)	0.977746212	0.978456081	0.977633437
(0.2, 0.5, 0.3)	0.976167929	0.977003046	0.976023857
(0.4, 0.2, 0.4)	0.975549767	0.976897240	0.975950926
(0.3, 0.3, 0.4)	0.975536616	0.976371272	0.975414399
(0.3, 0.5, 0.2)	0.974116162	0.974884502	0.97391917
(0.2, 0.3, 0.5)	0.971748737	0.972713709	0.971601652
(0.4, 0.3, 0.3)	0.971590909	0.972461958	0.971394144
(0.5, 0.2, 0.3)	0.965428751	0.966800453	0.970095842
(0.5, 0.3, 0.2)	0.952651515	0.954990565	0.952498238

7.2.2.3 Further Insights

Upon close examination of the table, slight differences between various weight combinations were observed. Configurations that evenly spread the weights across Model 2 and Model 3 consistently gave us higher precision and F1-scores. This highlights how important it is to adjust the weights based on the strengths of each model and what the dataset needs.

8. Results Comparison

8.1 Comparative Analysis of Proposed Models and Ensembles with Custom-Designed Models and Ensembles from Reference Paper for IIT-D Dataset

The performance of our models and ensemble methods on the IIT-D dataset was thoroughly evaluated, and the results are presented in Table 13. A detailed comparison was made between the Average Ensemble and Weighted Ensemble techniques using our proposed models, as well as results from three relevant research papers.

8.1.1 Average Ensemble (Using Proposed Models)

The Average Ensemble method, applied to our custom models, achieved an impressive accuracy of 97.8%, precision of 98.09%, and F1-score of 97.25%. These results were compared with existing work. For example, the Average Ensemble method on the IIT-D dataset reported in[17] showed an accuracy of 93.08%. Our method outperformed this previous result, indicating a significant improvement in classification performance due to our custom CNN models. Other studies[36,38] reported accuracy values of 94.82% and 97.48%, respectively, while our proposed Average Ensemble achieved an even higher performance.

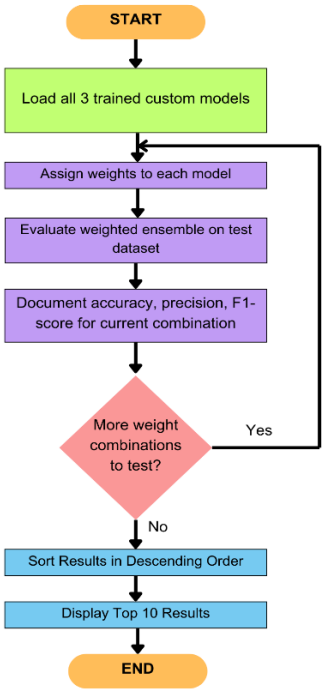

Fig. 8: Ensemble weight optimization workflow.

8.1.2 Weighted Ensemble

The Weighted Ensemble method, utilizing the weights 0.4, 0.2, and 0.4, demonstrated exceptional performance with an accuracy of 99.0%, precision of 99.37%, and F1-score of 98.99%. These results significantly surpass those of previous studies, such as the Weighted Ensemble method in[17] (accuracy 95.96%) and in[38] (accuracy 98.74%). Our Weighted Ensemble achieved the highest classification performance, further validating the effectiveness of the custom CNN models when combined through ensemble learning.

8.2 Comparative Analysis of Proposed Models and Ensembles with Custom-Designed Models and Ensembles for UERC Dataset

For the UERC dataset, we evaluated the performance of our proposed models and ensembles, as summarized in Table 14. The models considered were the custom Model 3, the Average Ensemble, and the Weighted Ensemble. For Custom Model (Model 3): The custom-designed Model 3 achieved an accuracy of 91.6%, precision of 92.3%, and F1-score of 91.7%. This model demonstrated solid performance as the best custom model, establishing a strong baseline for ear recognition tasks.

Table 13. Performance comparison on IIT-D dataset.

Method	Accuracy (%)	Precision (%)	F1-Score (%)
Average Ensemble			
Average Ensemble (Using Proposed Models)	**97.8**	**98.09**	**97.25**
Average Ensemble (IIT-D Dataset)[17]	93.0817	93.7107	93.1237
CNN model for IITD-II[36,38]	94.82	-	94.78
Average Ensemble (IIT-D Dataset) [38]	97.48	97.85	97.28
Weighted Ensemble			
Weighted Ensemble (0.4, 0.2, 0.4) (Using Proposed Models)	**99.0**	**99.37**	**98.99**
Weighted (IIT-D Dataset) [17]	95.9638	96.7402	95.9638
Weighted (IIT-D Dataset) [38]	98.74	99.00	98.86

Table 14. Performance comparison on UERC dataset

Model	Accuracy (%)	Precision (%)	F1-Score (%)
Model 1	84.0	84.1	84.1
Model 2	91.1	91.7	91.0
Best Custom Model (Model 3)	91.6	92.3	91.7
Average Ensemble	95.2	95.41	95.19
Weighted Ensemble (0.3, 0.4, 0.3)	97.79	97.86	97.78

8.2.1 Average Ensemble

The Average Ensemble method improved upon Model 3's performance, achieving an accuracy of 95.2%, which is a 3.6 percentage point improvement over Model 3. Additionally, precision and F1-score also saw improvements, reaching 95.41% and 95.19%, respectively. This shows that ensemble techniques can effectively enhance the performance of individual models by aggregating their predictions.

8.2.2 Weighted Ensemble

The most notable improvement was achieved using the Weighted Ensemble method, with weights of 0.3, 0.4, and 0.3. This ensemble configuration resulted in an accuracy of 97.79%, a 6.19 percentage point improvement over Model 3. The precision and F1-score also increased to 97.86% and 97.78%, respectively. These results underline the power of ensemble learning in refining model predictions and achieving superior classification accuracy.

9. Conclusion & Future Scope

This paper examines the potential prospects of an innovative biometric identification technique, ear biometrics, which is promising due to its distinctness, structural invariance and non-intrusiveness. Biometric systems based on the ear become a strong candidate for access control, surveillance, and forensic identification systems because of these unique characteristics. In this paper, three custom designs models are developed to solve various ear biometrics tasks. All the models have been thoroughly validated with the help of two diverse datasets: the IITD and UERC. The results demonstrated the validity of ear biometrics tasks and provided new paradigms for their implementation.

The results show that, in combination with ear biometrics, the use of ensemble learning contributes to a considerable increase in performance. As achieved by adoption of average ensemble and weighted ensemble, we accomplished significant increments in classification accuracy, precision, and F1-score. Average ensemble results using best parameters for the custom models displaying accuracy, precision, and F1-score were 97.8%, 98.09, and 97.25% respectively for IIT-D Dataset and 952%, 95.41% and 95.19% for UERC dataset; On the other hand, Top Weighted Ensemble method showed even better

results. For IIT-D dataset, accuracy, precision and F1-Score reaches to 99%, 99.37% and 98.99% for specific weights to all the custom models, whereas the values for UERC rises to 97.79%, 97.85% and 97.77%. These improvements highlight how beneficial ensemble methods are in terms of increasing the reliability and robustness of ear biometrics systems.

Work in this paper paves the way for this new and promising area of research and has shown the significant contribution that customized models and sophisticated class-based ensemble techniques can add to classification accuracy. Risks of biometric data being misused decrease especially when it comes to ear biometrics, which can potentially be used for authentication in a hands-free manner, making it the flexible and safe option for various applications. Future research can build on this work by investigating other ensemble techniques, including stacked, boosted, and hierarchical ensembles to improve the accuracy and adaptability of such systems even further. The generalizability or real-world applicability of the models will be enhanced by expanding the dataset to include more diverse demographic groups and real-life conditions.

To summarize, ear biometrics could open doors to improved password less authentication systems that avoid user discomfort associated with existing biometric systems. Considering myriad technological advancements with every passing day and extensive research underway, ear biometrics will become one of the largest scales in critical security applications to enhance security this is a sentiment echoed in the recent market research initiatives.

References

1) H. Alshazly, C. Linse, E. Barth, S.A. Idris, and T. Martinetz, "Towards explainable ear recognition systems using deep residual networks," *IEEE Access*, 9 122254–122273 (2021). doi:10.1109/ACCESS.2021.3109441.
2) P. Kumar, N. Sharma, A.R.-I.J. of Computer, and undefined 2012, "Handwritten character recognition using different kernel based svm classifier and mlp neural network (a comparison)," *CiteseerP Kumar, N Sharma, A RanaInternational Journal of Computer Applications, 2012•Citeseer*, 53 (*11*) 975–8887 (2012). https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=cb2e16204c143ec14e343c23e90c441f2447f9fe (Accessed September 7, 2024).
3) Ş. Kılıç, Y.D.-T. du Signal, and undefined 2023, "Deep learning based gender identification using ear images," *Avesis.Kayseri.Edu.TrŞ Kılıç, Y DoğanTraitement Du Signal, 2023•avesis.Kayseri.Edu.Tr*, (2023). doi:10.18280/ts.400431.
4) Z. Wang, B. Huang, G. Wang, P. Yi, and K. Jiang, "Masked face recognition dataset and application," *IEEE Trans Biom Behav Identity Sci*, 5 (*2*) 298–304 (2023). doi:10.1109/TBIOM.2023.3242085.
5) N. Sharma, … P.D.-I.J. of, and undefined 2021, "A hybrid approach for automatic licence plate recognition system," *Ingentaconnect.ComN Sharma, PK Dahiya, BR MarwahInternational Journal of Sensors Wireless Communications and Control, 2021•ingentaconnect.Com*,. https://www.ingentaconnect.com/content/ben/swcc/2021/00000011/00000001/art00010 (Accessed September 7, 2024).
6) N. Sharma, P.K. Dahiya, R. Lalit, M.A. Haq, B.R. Marwah, N. Mittal, and I. Keshta, "Deep learning and svm-based approach for indian licence plate character recognition.," *Researchgate.NetN Sharma, MA Haq, PK Dahiya, BR Marwah, R Lalit, N Mittal, I KeshtaComputers, Materials & Continua, 2023•researchgate.Net*,. doi:10.32604/cmc.2023.027899.
7) M.A. Jopek, K. Pastuszak, S. Cygert, M.G. Best, T. Wurdinger, J. Jassem, A.J. Zaczek, and A. Supernat, "Deep learning-based, multiclass approach to cancer classification on liquid biopsy data," *IEEE J Transl Eng Health Med*, 12 306–313 (2024). doi:10.1109/JTEHM.2024.3360865.
8) M. Subramanian, J. Cho, V.E. Sathishkumar, and O.S. Naren, "Multiple types of cancer classification using ct/mri images based on learning without forgetting powered deep learning models," *IEEE Access*, 11 10336–10354 (2023). doi:10.1109/ACCESS.2023.3240443.
9) A. Mahajan, and S.K. Singla, "Csa-gru: a hybrid cnn and self attention gru for human identification using ear biometrics," *Evolving Systems*, (2023). doi:10.1007/S12530-023-09555-4.
10) Ž. Emeršič, V. Štruc, and P. Peer, "Ear recognition: more than a survey," *Neurocomputing*, 255 26–39 (2016). doi:10.1016/j.neucom.2016.08.139.
11) K. Chang, K.W. Bowyer, S. Sarkar, and B. Victor, "Comparison and combination of ear and face images in appearance-based biometrics," *IEEE Trans Pattern Anal Mach Intell*, 25 (*9*) 1160–1165 (2003). doi:10.1109/TPAMI.2003.1227990.
12) Y. Moolla, A. De Kock, G. Mabuza-Hocquet, C.S. Ntshangase, N. Nelufule, and P. Khanyile, "Biometric recognition of infants using fingerprint, iris, and ear biometrics," *IEEE Access*, 9 38269–38286 (2021). doi:10.1109/ACCESS.2021.3062282.
13) E. Lobacheva, N. Chirkova, M. Kodryan, and D. Vetrov, "On power laws in deep ensembles," *Proceedings.Neurips.CcE Lobacheva, N Chirkova, M Kodryan, DP VetrovAdvances in Neural Information Processing Systems, 2020•proceedings.Neurips.Cc*,. https://proceedings.neurips.cc/paper_files/paper/2020/hash/191595dc11b4d6e54f01504e3aa92f96-Abstract.html (Accessed September 7, 2024).
14) Y. Yang, H. Lv, and N. Chen, "A survey on ensemble learning under the era of deep learning," *Artificial*

Intelligence Review 2022 56:6, 56 (*6*) 5545–5589 (2022). doi:10.1007/S10462-022-10283-5.

15) H. Kaushik, D. Singh, M. Kaur, … H.A.-I., and undefined 2021, "Diabetic retinopathy diagnosis from fundus images using stacked generalization of deep models," *Ieeexplore.Ieee.OrgH Kaushik, D Singh, M Kaur, H Alshazly, A Zaguia, H HamamIEEE Access, 2021•ieeexplore.Ieee.Org*,. https://ieeexplore.ieee.org/abstract/document/9500222/ (Accessed September 7, 2024).
16) B. Lakshminarayanan, A. Pritzel, and C.B. Deepmind, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Proceedings.Neurips.CcB Lakshminarayanan, A Pritzel, C BlundellAdvances in Neural Information Processing Systems, 2017•proceedings.Neurips.Cc*,. https://proceedings.neurips.cc/paper_files/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html (Accessed September 7, 2024).
17) R. Mehta, and K.K. Singh, "Ensemble of transfer learning and light-weight convolutional neural network model for an effective ear recognition system," *Evolving Systems*, 15 (*1*) 115–131 (2024). doi:10.1007/S12530-023-09561-6/METRICS.
18) A. Gona, M. Subramoniam, and R. Swarnalatha, "Transfer learning convolutional neural network with modified lion optimization for multimodal biometric system," *Computers and Electrical Engineering*, 108 108664 (2023). doi:10.1016/J.COMPELECENG.2023.108664.
19) H. Alshazly, C. Linse, E. Barth, and T. Martinetz, "Deep convolutional neural networks for unconstrained ear recognition," *IEEE Access*, 8 170295–170310 (2020). doi:10.1109/ACCESS.2020.3024116.
20) "VGG-net architecture explained. the company visual geometry group… | by siddhesh bangar | medium," (n.d.). https://medium.com/@siddheshb008/vgg-net-architecture-explained-71179310050f (Accessed September 10, 2024).
21) "AlexNet explained | papers with code,". https://paperswithcode.com/method/alexnet (Accessed September 10, 2024).
22) C. Szegedy, V. Vanhoucke, S. Ioffe, and J. Shlens, "Rethinking the inception architecture for computer vision,".
23) K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December 770–778 (2015). doi:10.1109/CVPR.2016.90.
24) "ResNeXt explained | papers with code,". https://paperswithcode.com/method/resnext (Accessed September 10, 2024).
25) Q. Zhang, "A novel resnet101 model based on dense dilated convolution for image classification," *SN Appl Sci*, 4 (*1*) 1–13 (2022). doi:10.1007/S42452-021-04897-7/FIGURES/12.
26) H. Alshazly, C. Linse, E. Barth, S.A. Idris, and T. Martinetz, "Towards explainable ear recognition systems using deep residual networks," *IEEE Access*, 9 122254–122273 (2021). doi:10.1109/ACCESS.2021.3109441.
27) M.S.-M.T. and Applications, and undefined 2022, "Ear recognition with ensemble classifiers; a deep learning approach," *SpringerM SharkasMultimedia Tools and Applications, 2022•Springer*, 81 (*30*) 43919–43945 (2022). doi:10.1007/s11042-022-13252-w.
28) C. Szegedy, W. Liu, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions,".
29) "(PDF) international journal of intelligent systems and applications in engineering advancements in nsfw content detection: a comprehensive review of resnet-50 based approaches,". https://www.researchgate.net/publication/374812284_International_Journal_of_INTELLIGENT_SYSTEMS_AND_APPLICATIONS_IN_ENGINEERING_Advancements_in_NSFW_Content_Detection_A_Comprehensive_Review_of_ResNet-50_Based_Approaches (Accessed September 10, 2024).
30) Y. Zhang, Z. Mu, L. Yuan, C.Y.-I. Biometrics, and undefined 2018, "Ear verification under uncontrolled conditions with convolutional neural networks," *Wiley Online LibraryY Zhang, Z Mu, L Yuan, C YuIET Biometrics, 2018•Wiley Online Library*, 7 (*3*) 185–198 (2018). doi:10.1049/iet-bmt.2017.0176.
31) Y. Zhang, Z. Mu, L. Yuan, C. Yu, and Q. Liu, "USTB-helloear: a large database of ear images photographed under uncontrolled conditions," *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10667 LNCS 405–416 (2017). doi:10.1007/978-3-319-71589-6_35.
32) R. Ayoub, and A. Oberoi, "Study of ear biometrics based identification system using machine learning," *Academia.EduR Ayoub, A Oberoiacademia.Edu*,. https://www.academia.edu/download/78504568/Study_of_Ear_Biometrics_Based_Identification_System_Using_Machine_Learning.pdf (Accessed December 3, 2024).
33) R. Mehta, G. Ujjwal, … S.S.-… C. on D., and undefined 2023, "Rotation invariant 2d ear recognition using gabor filters and ensemble of pre-trained deep convolutional neural network model," *Ieeexplore.Ieee.OrgR Mehta, G Ujjwal, SJ Shilpa, S Vityazev, KK Singh2023 25th International Conference on Digital Signal Processing, 2023•ieeexplore.Ieee.Org*,. https://ieeexplore.ieee.org/abstract/document/10113436/ (Accessed December 3, 2024).
34) Ž. Emeršič, B. Meden, P. Peer, and V. Štruc,

"Evaluation and analysis of ear recognition models: performance, complexity and resource requirements," *Neural Comput Appl*, 32 (*20*) 15785–15800 (2020). doi:10.1007/S00521-018-3530-1.

35) A. Mahajan, and S.K. Singla, "Csa-gru: a hybrid cnn and self attention gru for human identification using ear biometrics," *Evolving Systems*, 15 (*4*) 1197–1218 (2024). doi:10.1007/S12530-023-09555-4.
36) A. Mahajan, S.S.-C.-C.M. in, and undefined 2024, "DeepBio: a deep cnn and bi-lstm learning for person identification using ear biometrics.," *Search.Ebscohost.ComA Mahajan, SK SinglaCMES-Computer Modeling in Engineering & Sciences, 2024•search.Ebscohost.Com*,. https://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=15261492&AN=180106537&h=PQMJwq74ppxjDz6P2IXmgsPuYs%2FxZdLCpoDWFLJWDzLGb7UQ2ahSKqqgxJ9ST%2B2qbvteuWao8g6erPmn%2BmQQQA%3D%3D&crl=c (Accessed December 3, 2024).
37) A. Booysens, and S. Viriri, "Ear biometrics using deep learning: a survey," *Applied Computational Intelligence and Soft Computing*, 2022 (2022). doi:10.1155/2022/9692690.
38) R. Mehta, and K.K. Singh, "An efficient ear recognition technique based on deep ensemble learning approach," *Evolving Systems*, 15 (*3*) 771–787 (2024). doi:10.1007/S12530-023-09505-0.
39) A. Kumar, and C. Wu., "ITD-ii: ear database, 2012. http://www4.comp.polyu.edu.hk/~csajaykr/myhome/database_request/ear/.," (Accessed December 17, 2024)
40) A. Kumar, C.W.-P. Recognition, and undefined 2012, "Automated human identification using ear imaging," *ElsevierA Kumar, C WuPattern Recognition, 2012•Elsevier*,. doi:10.1016/j.patcog.2011.06.005.
41) "Ear recognition research | university of ljubljana," (n.d.). http://awe.fri.uni-lj.si/uerc/ (Accessed September 10, 2024).
42) Ž. Emeršič, A.K.S. V., B.S. Harish, W. Gutfeter, J.N. Khiarak, A. Pacut, E. Hansley, M.P. Segundo, S. Sarkar, H. Park, G.P. Nam, I.-J. Kim, S.G. Sangodkar, Ü. Kaçar, M. Kirci, L. Yuan, J. Yuan, H. Zhao, F. Lu, J. Mao, X. Zhang, D. Yaman, F.I. Eyiokur, K.B. Özler, H.K. Ekenel, D.P. Chowdhury, S. Bakshi, P.K. Sa, B. Majhi, P. Peer, and V. Štruc, "The unconstrained ear recognition challenge 2019 - arxiv version with appendix," (2019). http://arxiv.org/abs/1903.04143. (Accessed December 17, 2024)
43) A. Buslaev, V.I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A.A. Kalinin, "Albumentations: fast and flexible image augmentations," *Information (Switzerland)*, 11 (*2*) (2020). doi:10.3390/INFO11020125.