2023 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD) | 979-8-3503-5866-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICICT4SD59951.2023.10303591

Deep Learning and Ensemble Machine Learning for Bengali Vowel Recognition from Consonant Pronunciation using Face and Lip Images

A. Hasib Uddin
Computer Science and Engineering Discipline
Khulna University
Khulna, Bangladesh
Email: hasib1536@cseku.ac.bd

Abu Shamim Mohammad Arif
Computer Science and Engineering Discipline
Khulna University
Khulna, Bangladesh
Email: shamim@cseku.ac.bd

***Abstract*—Phonetic feature identification, particularly vowels, is crucial for speech recognition systems. Visual speech recognition in Bengali is a relatively unexplored area. This work focuses on Bengali vowel identification using consonant pronunciation from images. For this, we created a dataset consisting of six Bengali vowel pronunciations, each from four speakers; and 27 Bengali consonants, each with six corresponding categories from one speaker. After manual processing, the dataset contained a total of 2030 consonant-pronouncing images. We considered face and lip images in different steps. To establish a benchmark we applied four popular Machine Learning algorithms, namely Support Vector Machine, K-nearest Neighbors, Decision Tree, and Naive Bayes. Then, we proposed an ensemble ML model, which outperformed all the benchmark performances. We identified the challenging vowels through performance analysis, revealing two pairs of homonym-like vowels: "i" (in) and "e" (everything), as well as "u" (push) and "o" (bold). Next, we created two subsets of the dataset to address these challenges. The first subset contained data for four vowel pronunciations, including "i" and "u", while the second one contained data for four vowel pronunciations, including "e" and "o". Both the benchmark and proposed ML models were applied to these subsets, consistently demonstrating superior performance by the proposed ML approach. To further enhance classification efficiency, a Deep Learning architecture was proposed specifically for classifying the two subsets. The proposed DL model surpassed ML algorithms, including the proposed ensemble ML approach. This work will greatly benefit researchers and practitioners working on visual speech recognition in Bengali.**

Keywords—Bengali vowels, consonant pronunciation, Visual Speech Recognition, Machine Learning, Deep Learning

I. INTRODUCTION

The identification of phonetic features, particularly vowels, is a crucial aspect of speech analysis and processing. In the context of Bengali language, accurately recognizing and distinguishing between vowels plays a vital role in various applications, including speech recognition systems, linguistic research, and language learning tools. The ability to accurately identify Bengali vowels from consonant pronunciations holds significant potential for advancing the field of visual speech recognition.

Barbara and Ruth, in their book "Hearing by eye: The psychology of lip-reading," emphasized on the importance of lip-reading as a cognitive function [1]. Chung et al. worked on the recognition of phrases and sentences from talking faces [2], while Chung and Zisserman demonstrated lip reading from unseen videos, even in profile view [3]. Akhter and Chakrabarty focused on lip segmentation and viseme recognition using neural networks in the context of Bengali [4].

Prajwal et al. applied a visual attention model for sub-word level lip reading [5]. They introduced an attention-based pooling mechanism and utilized sub-word units, achieving significant performance on publicly available datasets like LRS2 and LRS3 [2, 3]. Their most accurate method achieved a word error rate of 22.6% on the LRS2 dataset. Ren et al. employed a cross-modal knowledge transfer scheme for lip reading, emphasizing the training of a model on audio and transferring that knowledge for visual speech recognition [6]. Their curriculum learning design improved the model's convergence and performed well in both word-level and sentence-level visual speech recognition. Sheng et al. proposed an Adaptive Semantic-Spatio-Temporal Graph Convolutional Network for lip reading, which incorporated both spatial and temporal information from videos [7].

Visual speech recognition in the Bengali language is a relatively unexplored area. To the best of our knowledge, there are only a few works on this topic in Bangla. For example, Mahboob et al. considered only five Bengali words [8]. They employed an end-to-end model with recurrent network spatio-temporal convolutions, focusing on sentence-level prediction using variable-length video frames. On the other hand, Akhter and Chakrabarty [4] utilized lip curvature and neural networks to recognize three Bengali vowels.

Lip reading poses a challenge due to the requirement of comprehensive datasets. While publicly available datasets like LRS2 and LRS3 exist in English [2, 3], there is currently no comprehensive dataset specific to this topic in Bangla. This imposes significant challenge to advance this kind research in Bengali Language.

Despite advancements, lip-reading sentences still struggle to achieve accuracy levels comparable to word-based methods [9]. Automating the lip reading of individuals speaking phrases with different lexicons and incorporating unfamiliar terminology remains a challenging task. Every word and phrase are comprised with consecutive vowels. For this reason, identifying the correct vowel is fundamental for effective visual speech recognition.

This work focuses on the identification of Bengali vowels from Bengali consonant pronunciation using image analysis techniques. Accurate recognition of vowels plays a crucial role in various applications, including speech recognition and

natural language processing. By accurately identifying and distinguishing between Bengali vowels, researchers and practitioners can enhance the performance of systems designed for speech recognition and visual speech processing in the Bengali language.

To facilitate this research, a dataset was created, consisting of both Bengali vowel and consonant pronunciation videos. The dataset included six Bengali vowel pronunciation videos, each pronounced by four different speakers. Additionally, there were 27 Bengali consonants, each with six different vowel pronunciations, resulting in a total of 162 consonant pronunciation videos from a single speaker.

After extracting frames from these videos and manually labeling them, the dataset contained a substantial number of consonant and vowel pronouncing images for training, validation, and testing. Specifically, it comprised 2030 consonant pronouncing images for training, 81 vowel pronouncing images for validation, and 264 vowel pronouncing images for testing.

To establish a benchmark for performance evaluation, four popular Machine Learning (ML) algorithms, namely Support Vector Machine (SVM), K-nearest Neighbors (KNN), Decision Tree (DT), and Naive Bayes (NB), were applied to the dataset. Subsequently, an ensemble ML model was proposed, which outperformed all the benchmark algorithms, indicating the effectiveness of the approach in vowel identification.

Through performance analysis, the study identified certain challenging vowels that presented difficulties in accurate recognition. Specifically, two pairs of homonym-like vowels, namely "i" as in "in" and "e" as in "everything", as well as "u" as in "push" and "o" as in "bold", were found to be particularly challenging.

To address these challenges, two subsets of the dataset were created. One subset contained data for four vowel pronunciations, including "i" and "u", while another subset contained data for four vowel pronunciations, including "e" and "o". Both the benchmark and proposed ML models were applied to these subsets, consistently demonstrating superior performance by the proposed ML approach.

Furthermore, in order to further enhance the efficiency of vowel classification, a Deep Learning (DL) architecture specifically designed for classifying the challenging vowel subsets was proposed. The proposed DL model surpassed all the ML algorithms, including the proposed ML model, showcasing the potential of deep learning techniques in tackling the complexities of Bengali vowel identification.

The outcomes of this research hold significant implications for researchers and practitioners involved in visual speech recognition and natural language processing in the Bengali language. The accurate identification and recognition of Bengali vowels can greatly enhance the performance and effectiveness of various applications in speech processing, opening new avenues for advancements in this field.

The rest of the paper is organized as follows: Section II provides a detailed description of the dataset preparation process. Section III presents the methods employed in this study, including the benchmark ML algorithms used for vowel recognition. Section IV is dedicated to the performance analysis and discussion of the experimental results, examining the accuracy achieved by the benchmark ML models and highlighting the superior performance of the proposed ensemble ML model. Finally, in Section V, the paper concludes with a summary of the findings and proposes future works to expand and improve upon the research presented

II. Dataset Preparation

A. Data Collection

The dataset preparation begins by considering six Bengali vowels represented as "a", "aa", "i", "u", "e", and "o." The corresponding original Bengali characters and examples with similar pronunciation are provided in the Fig 1. Six Bengali vowel pronunciation video data were collected from four speakers. Additionally, 27 Bengali consonants, each combined with the six vowels, resulted in a total of 162 consonant pronunciation video data from one speaker. This comprehensive collection formed the basis of the dataset.

Bengali Vowel	English Representation	Word with Similar Pronunciation
অ	a	all
আ	aa	aah
ই	i	in
উ	u	push
এ	e	every
ও	o	bold

Fig. 1. Bengali characters, corresponding English representation for this paper, and example words with similar pronunciation.

B. Pre-processing

1) Pre-processing of Face Dataset

The Haar Cascade classifier was applied to identify the face in each image. Following that, an unsharp masking technique was employed to enhance the facial features. Finally, the lip area was cropped from each face image to isolate the relevant visual cues.

2) Pre-processing of Lip Dataset

Similar to the face dataset, the Haar Cascade classifier was used to identify the face in each image. Subsequently, the lip area was cropped from each face image to extract the specific lip movements.

Examples for face and lip images from each of the six Bengali vowels pronunciation from the test data is included in Fig 2.

Fig. 2. Example images of face and lip from six Bengali vowels.

C. Dataset Split

The dataset was divided into three sets for distinct purposes. The training set comprised the consonant pronouncing images. The validation set consisted of vowel pronouncing images from a different person than the one pronouncing the consonants. The testing set included vowel pronouncing images from three individuals: one person who pronounced the consonants (same as the training subset) and two other persons who differed from the person pronouncing the vowels in the validation set. This division allowed for investigating the generalizability of visual speech recognition

in Bengali. We used the validation only while training the proposed DL model.

D. Post-processing

After the dataset was prepared, several postprocessing steps were performed. First, image normalization was applied to ensure consistency in brightness and contrast across the dataset. The images were then resized to dimensions of 64x64 pixels for use in the Machine Learning (ML) models and 256x256 pixels for the Deep Learning (DL) model. Additionally, color space conversions were carried out: BGR to grayscale for the ML models and BGR to RGB for the DL model. These steps standardized the dataset and prepared it for further analysis and model training.

E. Subsets of Dataset

After a careful performance analysis of the ML models on the face and lip datasets with six vowel categories, we successfully identified a set of challenging vowels in our study. Specifically, we discovered two pairs of homonym-like vowels: "i" (pronounced as the first letter in "indication") and "e" (pronounced as the first letter in "everything"), and "u" (pronounced as the second letter in "push") and "o" (pronounced as the second letter in "bold"). These pairs posed difficulties in accurate recognition. To address this challenge, we devised a strategic approach by creating two subsets of the dataset. One subset consisted of data related to four vowel pronunciations, including "i" and "u". On the other hand, the later on encompassed data pertaining to four vowel pronunciations, including "e" and "o". This subdivision allowed us to specifically focus on and effectively tackle the complexities associated with these challenging vowel pairs within our dataset.

F. Final Data Setups

In this paper, we have worked on a total of ten different data setups. Details of these setups are listed in Table 1.

TABLE I. OVERALL DATA SETUPS USED IN THIS STUDY

Setup	Vowels	Feature	Channel	Dimension	Train	Validation	Test
S1	All six vowels	Face	Gray	64x64	2030	-	264
S2		Lip	Gray	64x64	2030	-	264
S3	"a", "aa", "i", "u"	Face	Gray	64x64	2030	-	264
S4			RGB	256x256	2030	81	264
S5		Lip	Gray	64x64	2030	-	264
S6			RGB	256x256	2030	81	264
S7	"a", "aa", "e", "o"	Face	Gray	64x64	2030	-	264
S8			RGB	256x256	2030	81	264
S9		Lip	Gray	64x64	2030	-	264
S10			RGB	256x256	2030	81	264

III. METHODS

We applied the methods in three broad steps-(i) benchmark, (ii) propose and apply ML method, and (iii) propose and apply DL method.

A. Benchmark

To obtain benchmark performance on our dataset, we applied SVM, KNN, DT, and NB on S1, S2, S3, S5, S7, and S9 data setups. We trained the models on Kaggle platform and utilized the default parameters provided by Scikit Learn in Python.

B. Proposed Ensemble ML Model

The proposed ensemble model is designed to combine the strengths of multiple classifiers to improve overall predictive performance. It consists of seven individual classifiers, each with its own distinct characteristics and decision-making strategies.

The first three classifiers are SVMs with different kernel functions. SVM with RBF kernel (Radial Basis Function) is effective in capturing non-linear relationships, while SVM with polynomial kernel can handle complex feature interactions. SVM with sigmoid kernel is suitable for modeling non-linear decision boundaries. These classifiers are parameterized with the regularization strength (C=1) to balance between fitting the training data and allowing for a wider margin.

The fourth classifier is the Random Forest (RF) with 100 estimators, which constructs multiple decision trees and combines their predictions through an ensemble approach. By aggregating the outputs of multiple trees, the Random Forest can capture complex relationships and handle noise and outliers.

The fifth classifier is the KNN algorithm, which classifies data instances based on the classes of its nearest neighbors. The number of neighbors considered is 6.

The sixth classifier is the DT, which uses a tree-like structure to make decisions based on the values of features. The tree was allowed to expand until all leaves were pure or until all leaves contain less than the minimum number of samples (set to 2) required to split an internal node.

The seventh classifier is Gaussian Naive Bayes, which assumes that features follow a Gaussian distribution. This classifier calculates class probabilities based on this assumption.

All individual classifiers are combined into an ensemble model using the maximum voting. The ensemble model employs a soft voting strategy, where the probabilities predicted by each classifier are taken into account. By leveraging the collective knowledge of the individual classifiers, the ensemble model aims to make more accurate predictions.

The combination of these diverse classifiers with the experimentally selected parameter settings and kernel functions enables the ensemble model to capture different aspects of the data, leveraging the strengths of each classifier to enhance overall performance.

C. Proposed DL Model

1) Model Architecture

In our proposed deep learning architecture, we begin with feature extraction using the InceptionResNetV2 model pre-trained on the ImageNet dataset. This allows us to leverage the rich learned representations from a large-scale image classification task.

Moving on to our custom model (Fig 3), we define several layers to further process the extracted features. In Layer 1, we employ a ConvLSTM1D layer with 128 units and a kernel size of 2. This layer captures temporal dependencies in the data, incorporating both convolutional and LSTM operations.

To prevent overfitting, we introduce Layer 2, a Dropout layer with a rate of 0.5, which randomly drops out 50% of the input units during training.

In Layer 3, we flatten the output from the previous layer, converting it into a 1-dimensional feature vector.

Layer 4 consists of a Dense layer with 1024 units, using the 'elu' activation function. This layer introduces non-linearity to the model and enables it to learn complex patterns in the data.

Another Dropout layer, Layer 5, with a rate of 0.5, is added to further regularize the model and reduce the risk of overfitting.

In Layer 6, we have another Dense layer with 512 units and 'elu' activation. This layer helps to further extract high-level features from the flattened representation.

For the final output layer, Layer 7, we use a Dense layer with the number of units equal to the number of classes in the classification task. We apply the softmax activation function, which converts the output into class probabilities, allowing us to interpret the model's prediction in terms of class likelihoods.

We trained the model for a total of 10 epochs. Fig 4 visualizes the learning curve of our proposed DL model on the S4 setup (best performance).

Overall, this architecture combines both the power of transfer learning from InceptionResNetV2 and a custom-designed model with ConvLSTM1D, dense layers, and dropout regularization. This allows us to leverage pre-trained knowledge while tailoring the model to our specific classification task, providing a robust and accurate solution.

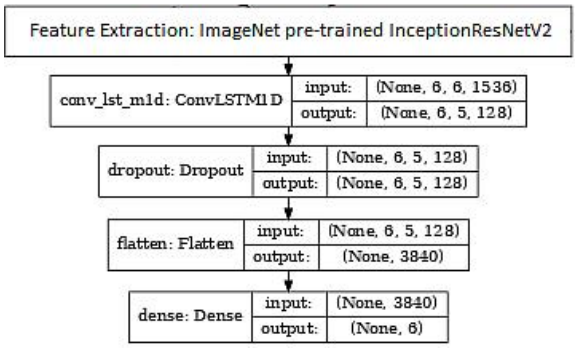

Fig. 3. Proposed Deep Learning model architecture. No of neurons in the last layer will be the number of classes.

2) Hyper-parameter Settings

For our proposed DL model, we performed hyper-parameter tuning. The tuned hyper-parameters are provided below:

Batch Size: We set the batch size to 16 for training, while for validation and testing, the batch size was 1.

Learning Rate: We chose a smaller learning rate of 0.001 indicates for more cautious and gradual adjustment of weights and biases.

Learning Rate Decay: It prevents division by zero and ensures smooth convergence during training. We set the decay rate at 0.0001 to maintain numerical stability.

Loss Function: We used the categorical cross-entropy loss function for our classification problems.

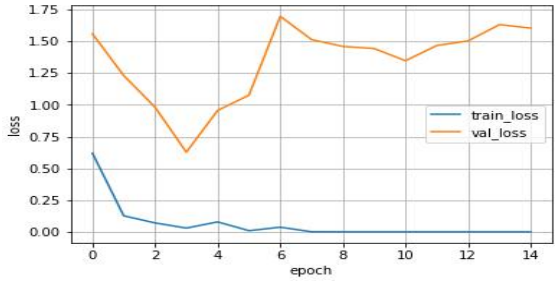

Fig. 4. Learning curve for applying the proposed DL model on S4 data setup.

TABLE II. PERFORMANCE COMPARISON (BEST PERFORMANCES ARE DENOTED AS BOLD).

Vowels	Feature	Setup	Model	Accuracy (%)
"a", "aa", "i", "u", "e", "o"	Face	S1	SVM	42.42
			KNN	51.51
			DT	48.10
			NB	49.24
			Proposed ML	**59.84**
	Lip	S2	SVM	44.31
			KNN	57.95
			DT	44.69
			NB	48.48
			Proposed ML	**58.33**
"a", "aa", "i", "u"	Face	S3	SVM	42.42
			KNN	51.51
			DT	44.31
			NB	49.24
			Proposed ML	56.06
		S4	Proposed DL	**83.43**
	Lip	S5	SVM	38.26
			KNN	53.91
			DT	40.57
			NB	47.24
			Proposed ML	56.52
		S6	Proposed DL	**76.00**
"a", "aa", "e", "o"	Face	S7	SVM	42.42
			KNN	51.51
			DT	44.31
			NB	49.24
			Proposed ML	**62.50**
		S8	Proposed DL	59.78
	Lip	S9	SVM	38.26
			KNN	53.91
			DT	43.18
			NB	47.24
			Proposed ML	55.07
		S10	Proposed DL	**58.66**

Optimizer: The optimizer determines the algorithm used to update the network parameters based on the computed gradients. "RMSProp" (Root Mean Square Propagation) is an optimization algorithm that adapts the learning rate for each weight based on the historical gradient information. It helps converge faster and efficiently in different regions of the parameter space.

IV. PERFORMANCE ANALYSIS AND DISCUSSION

The detailed performance comparison among the benchmark, proposed ML, and proposed DL models is provided in Table 2.

In case of six vowels recognition from face images, the SVM, KNN, DT, and NB models achieved 42.42%, 51.51%, 48.10%, and 49.24% accuracy. On the other hand, our proposed ensemble ML model outperformed all the benchmark models by achieving 59.84% accuracy.

Similarly, for six vowels recognition from lip images, SVM, KNN, DT, and NB gained 44.31%, 57.95%, 44.69%,

and 48.48% accuracy, while the proposed ML model gained 58.33% accuracy, surpassing all the benchmarks.

Then, for four vowels ("a", "aa", "i", "u") classification from face images, the accuracy of SVM, KNN, DT, and NB were 42.42%, 51.51%, 44.31%, and 49.24%, respectively. Again, our proposed ML model outperformed the benchmarks by gaining 56.06% accuracy, while our proposed DL model outperformed all the ML approaches, including our proposed one by securing 83.43% accuracy.

Next, for the same four vowels recogntion from lip images, the benchmark ML models obtained 38.26%, 53.91%, 40.57%, and 47.24% respective accuracy. In contrast, our proposed ML model obtained the second best accuracy of 56.52%, while the proposed DL architecture secured the best performance of 76.00% accuracy.

After that, for the other four vowels ("a", "aa", "e", "o") classification from face images, the benchmark algorithms achieved 42.42%, 51.51%, 44.31%, and 49.24%, accordingly. Both the proposed ML and DL models surpassed all the benchmark values. Exceptionally, however, in this one case, the proposed ML model (62.50%) outperformed the proposed DL model (59.78%).

Finally, for the same four vowels classification from lip images, the corresponding benchmark performances were 38.26%, 53.91%, 43.18%, and 47.24%. The proposed ML model agained gained the second best performance with 55.07% accuracy, while the best accuracy (58.66%) was gained by the proposed DL model.

Overall, among all the cases, the best performance was achieved by our proposed DL model for ("a", "aa", "i", "u") recognition from face images.

V. Conclusion and Future Works

This work focused on the identification of Bengali vowels from Bengali consonant pronunciation using images. A dataset consisting of vowel and consonant pronunciation videos was created and processed to extract relevant frames for training and testing. various ML algorithms were employed, and an ensemble ML model was proposed, which outperformed the benchmark performances. Challenging vowels were identified, and subsets of the dataset were created to address these challenges. A DL architecture specifically designed for classifying the subsets achieved superior performance compared to the ML approaches in most of the cases. The findings of this study will greatly benefit researchers and practitioners working on visual speech recognition in Bengali.

Further exploration and improvement can be done in several areas. Firstly, the dataset can be expanded to include more speakers and a larger variety of pronunciations to enhance the model's generalization capability. Additionally, incorporating temporal information by utilizing sequential models such as Recurrent Neural Networks (RNNs) or Transformers can potentially improve the accuracy of vowel recognition. Furthermore, deploying the proposed DL architecture in real-time applications and evaluating its performance in different contexts and noise conditions would be valuable for practical implementation.

Acknowledgment

This work is funded by the division of Information and Communication Technology (ICT), Ministry of Posts, Telecommunications and Information Technology, Government of the People's Republic of Bangladesh.

References

[1] B. Dodd and R. Campbell, "Hearing by eye: The psychology of lip-reading", Lawrence Erlbaum Associates, Inc, 1987.

[2] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Lip reading sentences in the wild", in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3444-3453, 2017.

[3] J. S. Chung and A. P. Zisserman, "Lip reading in profile", 2017.

[4] N. Akhter and A. Chakrabarty, "Viseme Recognition using lip curvature and Neural Networks to detect Bengali Vowels", Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 9, no. 4, pp. 7-11, 2017.

[5] K. R. Prajwal, T. Afouras, and A. Zisserman, "Sub-word level lip reading with visual attention", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5162-5172, 2022.

[6] S. Ren, Y. Du, J. Lv, G. Han, and S. He, "Learning from the master: Distilling cross-modal advanced knowledge for lip reading", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13325-13333, 2021.

[7] C. Sheng, X. Zhu, H. Xu, M. Pietikainen, and L. Liu, "Adaptive semantic-spatio-temporal graph convolutional network for lip reading", IEEE Transactions on Multimedia, 2021.

[8] K. Mahboob, H. Nizami, F. Ali, and F. Alvi, "Sentences Prediction Based on Automatic Lip-Reading Detection with Deep Learning Convolutional Neural Networks Using Video-Based Features", in International Conference on Soft Computing in Data Science, pp. 42-53, Springer, Singapore, 2021.

[9] S. Fenghour, D. Chen, K. Guo, B. Li, and P. Xiao, "An Effective Conversion of Visemes to Words for High-Performance Automatic Lipreading", Sensors, vol. 21, no. 23, p. 7890, 2021.

[10] V. N. Vapnik and A. Ya Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities", in Measures of Complexity: Festschrift for Alexey Chervonenkis, 2015, pp. 11-30.

[11] C. M. Bishop and N. M. Nasrabadi, "Pattern recognition and machine learning", vol. 4, no. 4, New York, NY: Springer, 2006.

[12] T. M. Mitchell, "Machine learning", vol. 1, New York, NY: McGraw-Hill, 2007.

[13] "Scikit-learn: Decision Trees", scikit-learn.org. [Online]. Available: https://scikit-learn.org/stable/modules/tree.html#:~:text=scikit-learn%20uses%20an%20optimized,support%20categorical%20variables%20for%20now. [Accessed: May 30, 2023].

[14] F. Murtagh and M. M. Farid, "Pattern Classification, by Richard O. Duda, Peter E. Hart, and David G. Stork", Journal of Classification, vol. 18, no. 2, pp. 273-275, 2001.

[15] C. Robert, "Machine learning, a probabilistic perspective", 2014, pp. 62-63.