mathematics

Article

Mispronunciation Detection and Diagnosis with Articulatory-Level Feedback Generation for Non-Native Arabic Speech

Mohammed Algabri [1,2,*], Hassan Mathkour [1,2], Mansour Alsulaiman [2,3] and Mohamed A. Bencherif [2,3]

[1] Computer Science Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia; mathkour@ksu.edu.sa
[2] Center of Smart Robotics Research (CS2R), College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia; msuliman@ksu.edu.sa (M.A.); mabencherif@ksu.edu.sa (M.A.B.)
[3] Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
* Correspondence: malgabri@ksu.edu.sa

Abstract: A high-performance versatile computer-assisted pronunciation training (CAPT) system that provides the learner immediate feedback as to whether their pronunciation is correct is very helpful in learning correct pronunciation and allows learners to practice this at any time and with unlimited repetitions, without the presence of an instructor. In this paper, we propose deep learning-based techniques to build a high-performance versatile CAPT system for mispronunciation detection and diagnosis (MDD) and articulatory feedback generation for non-native Arabic learners. The proposed system can locate the error in pronunciation, recognize the mispronounced phonemes, and detect the corresponding articulatory features (AFs), not only in words but even in sentences. We formulate the recognition of phonemes and corresponding AFs as a multi-label object recognition problem, where the objects are the phonemes and their AFs in a spectral image. Moreover, we investigate the use of cutting-edge neural text-to-speech (TTS) technology to generate a new corpus of high-quality speech from predefined text that has the most common substitution errors among Arabic learners. The proposed model and its various enhanced versions achieved excellent results. We compared the performance of the different proposed models with the state-of-the-art end-to-end technique of MDD, and our system had a better performance. In addition, we proposed using fusion between the proposed model and the end-to-end model and obtained a better performance. Our best model achieved a 3.83% phoneme error rate (PER) in the phoneme recognition task, a 70.53% F1-score in the MDD task, and a detection error rate (DER) of 2.6% for the AF detection task.

Keywords: mispronunciation detection and diagnosis; object detection; feedback generation; non-native Arabic corpus; end-to-end MDD; TTS

MSC: 68T10

check for updates

Citation: Algabri, M.; Mathkour, H.; Alsulaiman, M.; Bencherif, M.A. Mispronunciation Detection and Diagnosis with Articulatory-Level Feedback Generation for Non-Native Arabic Speech. *Mathematics* **2022**, *10*, 2727. https://doi.org/10.3390/math10152727

Academic Editors: Liang Zou, Liang Zhao and Yonghui Xu

Received: 11 June 2022
Accepted: 26 July 2022
Published: 2 August 2022

1. Introduction

The importance of online learning has increased in recent years, especially during the current coronavirus pandemic (COVID-19), where many universities and learning centers worldwide switched to online rather than face-to-face teaching [1,2]. A CAPT system is an example of an online system for learning a second language without the need for a direct human tutor [3–5]. The MDD module is a vital component of an enhanced versatile CAPT system [6]. The MDD detects mispronounced phonemes and provides different types of feedback to the learner. The CAPT system can be enhanced by providing articulatory feedback about the pronunciation error in addition to detecting and recognizing pronunciation errors. The effectiveness of articulatory feedback for CAPT systems was shown by the authors of [7]. The authors of [8] studied the different types of feedback in classrooms and selected those that are more effective in CAPT systems. In this paper, we propose a high-performance versatile MDD for Arabic learners. The proposed system can locate the error in pronunciation, recognize the mispronounced phonemes, and detect the corresponding AFs, not only in words but even in sentences. The proposed system can generate articulatory feedback. Recently, deep learning techniques have been used successfully in different application areas [9]. A powerful and very successful example is deep-learning-based object detectors in computer vision applications [10,11]. Object detection techniques aim to recognize and localize different objects in an image and have been used for many real applications, such as video surveillance [12], drone scene analysis, and autonomous vehicle driving [11]. The use of object detectors in computer vision applications has been investigated extensively in the published research. In contrast, applying object detection techniques for speech applications has rarely been used in previous studies, except in a few applications, such as keyword spotting [13]. We studied the effectiveness of applying object detection techniques for phoneme recognition [14] and multi-label articulatory features (AF) detection [15], using a small amount of speech. The performance and obtained results encouraged us to investigate the use of multi-label object detection for MDD for non-native Arabic speech. Hence, in this study, we investigate the use of a multi-label object detector to recognize the sequence of phonemes and their associated AFs from non-native Arabic utterances. With multi-label detection, we can recognize the phonemes and their corresponding AFs simultaneously, which is more time-efficient than using one model for each task. The sequence of recognized phonemes was used to locate pronunciation errors of non-native Arabic learners. Moreover, the proposed system has the capability of providing feedback to learners by detecting the AFs of the pronounced phoneme from the whole utterance rather than from isolated words or phonemes, as was achieved by other works in the literature. We anticipate that the proposed system will have a great impact on online Arabic learning for non-native sparkers. Moreover, in this study, we investigate the use of the genetic algorithm (GA) to fine-tune the detector parameters to enhance its performance.

CAPT systems have been studied extensively for non-Arabic languages [5–7], with little attention having been given to the Arabic language. One of the challenges that the CAPT system faces, in general, is the scarcity of non-native speech corpora compared to the corpora for automatic speech recognition (ASR) systems, which recently achieved promising results due to the availability of a considerable quantity of training data [16]. The authors of [16] proposed using cross-lingual transfer learning to overcome the scarcity of non-native data, and they used a multitask learning framework where one task was for phonetic modeling for mispronunciation detection and the second task was for articulatory modeling for mispronunciation diagnosis. To address the problem of the scarcity of non-native Arabic speech corpora, we developed the non-native Arabic-CAPT corpus (Arabic-CAPT), which comprises speakers from 20 nationalities. We also investigated the use of neural TTS to produce high-quality non-native Arabic speech to enhance the performance of the proposed system.

The main contributions of this paper can be summarized as follows:

- We show the robustness of applying deep-learning-based object detectors for MDD (we name this system MDD-Object) and providing articulatory feedback by treating the phonemes and their AFs as objects with multiple labels within spectral images.
- We evaluate the proposed systems using the developed corpora, Arabic-CAPT and Arabic-CAPT-S, and we compare the performance with a state-of-the-art end-to-end MDD technique (we name this system MDD-E2E).
- We propose applying fusion between the proposed MDD-Object and end-to-end MDD technique (we name this system MDD-Object -E2E) systems.

The rest of this paper is organized as follows. Section 2 presents the related work on MDD for Arabic and non-Arabic speech. In Section 3, we show the details of developing non-native Arabic speech corpora. The details of the proposed systems and their different models are presented in Sections 4–6. In Section 7, we provide the results and discussion of the proposed systems. Finally, in Section 8, we conclude the paper and provide potential directions for future work.

2. Related Work

In this section, we present an overview of the current research in MDD for non-native Arabic speech using traditional methods and deep learning methods. Then, we present a general look at the current work on MDD in other languages. We conclude this section by noting some of the difficulties in this area of research and our contributions to address them.

2.1. MDD for the Arabic Language Using Traditional Methods

HAFSS is one of the earliest CAPT systems for learning Arabic pronunciation and was developed in [17]. In more specific terms, the main goal of the HAFSS system was to learn Holy Quran recitation. It used HMM for acoustic modeling and MLLR for speaker adaptation. For detecting recitation errors, HAFSS used the confidence score of the speech decoder and analysis of phoneme duration to detect the errors. In terms of the size of the data, the authors of [17] mentioned only the size of the testing data, which include 507 utterances labeled by language experts. The system detected 62.4% pronunciation errors with a 14.9% false acceptance rate. Another system for teaching Quran recitation was proposed in [18], which used HMM for speech recognition and a classifier-based method for detecting errors in recitation. In terms of the database, the authors of [18] used 6.5 h of recitation for training and one hour for testing, where the testing part contained 2689 words with 23,198 phonemes, and the system achieved a 97.6% accuracy at the word level without considering the phoneme pronunciation errors, while the accuracy decreased to 84% when considering the phoneme level pronunciation errors. For pronunciation detection, they proposed two classifier-based methods: one for the discrimination of emphasized phoneme/R// and non-emphasized phoneme/R/, and the other classifier to detect the most confusing Arabic phonemes that have close articulatory features. By using the above two classifiers, the accuracy increased to 91.2%. For the two classifier situations, they investigated different machine learning techniques, such as SVM, MLP, bagging, HMM, and random committee.

Pronunciation error detection of non-native Arabic speakers from Pakistan and India was proposed in [19]. The authors studied the common pronunciation errors of these two nationalities in the KSU Arabic speech database [20,21], and they found that the speakers had more errors when they pronounced five Arabic phonemes; hence, the authors focused on those phonemes in their research. They used a goodness of pronunciation (GOP) score of HTK to decide whether the phonemes were pronounced correctly. In terms of the database, they used a subset of the KSU Arabic speech database. They trained the model using mixed native and non-native speech and used only non-native speech for testing. Seven language experts performed the annotation of the selected five phonemes. The authors achieved a PER of 28.8% for the phoneme recognition task. For the performance of pronunciation error detection, they achieved 94.8%, 87.3%, 97.2%, 100%, and 75% scoring accuracy for the selected five phonemes /θ/, /ħ/, /s^{ʕ}/, /d^{ʕ}/, and /ð^{ʕ}/, respectively. The authors of [22] compared four classifier-based systems to detect Arabic mispronunciation phonemes using acoustic-phonetic features (APFs). They focused only on the five most confusing Arabic phonemes, which are /t^{ʕ}/, /t/, /ħ/, /x/, and /h/. In terms of the database, they recorded speech from 200 Pakistani speakers, where the number of labeled phonemes in the database was 5000. HTK was used to segment the phoneme automatically, and phonemes were labeled as correct or incorrect by five language experts. The highest detection accuracy achieved in this study was 95.4%. Another study by some of the authors of [22] proposed using an artificial neural network (ANN) to detect mispronunciation using a discriminative APF in [23]. They used a database with all 28 Arabic consonants, where the total number of phonemes in the database was 5600, divided into 4450 for training and 1150 for testing, and approximately 35% of the databases were mispronounced phonemes. They achieved an 82.27% overall average accuracy.

2.2. MDD for Arabic Language Using Deep Learning Methods

Despite the considerable development in deep learning applications in speech processing, in general, and non-Arabic mispronunciation detection in particular, applying deep learning techniques for learning Arabic pronunciation is rare. In this section, we present the efforts in this regard, and we discuss the obstacles of using deep learning in MDD for Arabic speech.

Detecting pronunciation errors in the Arabic consonant using a convolutional neural network (CNN) was presented in [24]. They used a CNN as a backbone to extract the deep features that were fed to different classifiers, such as KNN, SVM, and ANN. Moreover, they improved the model performance by training the CNN using transfer learning. As a baseline model, they used the same classifiers with handcrafted features. In terms of the database used, speech from 400 Pakistani speakers was collected and labeled using five Arabic experts. The total number of samples for all 28 Arabic phonemes used in the study was 11,164. They achieved 82.9%, 91%, and 92% accuracy for the baseline, CNN, and CNN with transfer learning, respectively. The authors of [25] proposed using deep CNN features for Arabic mispronunciation detection at the word level. Features extracted from different CNN layers were used as input to classifiers, such as SVM, KNN, and random forest. They also investigated transfer learning. MFCC as an input to the classifiers was used as a baseline. In terms of the database used, they recorded speech of Arabic words that covered all Arabic phonemes. They recorded data from 30 Pakistanis speakers, each of whom recorded 27 Arabic words. All recorded words were labeled by three experts, who labeled 45% of the words as mispronounced. The best accuracy was obtained using SVM based on CNN features, at 93.2%. The most recent study in this field was proposed in [26], where they used a deep CNN (DCNN), AlexNet, and bidirectional long short-term memory (BiLSTM) for the pronunciation detection of Arabic alphabets and alphabet classification. In terms of the database, they used an in-house database collected from 20 native and 20 non-native speakers for pronunciation classification. Then, they augmented the samples by another 120 samples per alphabet. The total number of audio samples in this database, after augmentation, was 8120. The obtained pronunciation detection results were 97.88%, 99.14%, and 77.71% using DCNN, AlexNet, and BiLSTM, respectively.

2.3. Overview of Recent Works on MDD for Non-Arabic Languages

CAPT systems for learning languages other than the Arabic language have received more attention in the literature compared to the Arabic language. In this section, we present recent studies on MDD for other languages, and we identify the well-known corpora used in the studies. The authors of [16] proposed a system for pronunciation error detection and diagnosis for Japanese speakers learning the English language using DNN. The proposed system consisted of two stages. In the first stage, a non-native speech recognition system was used to generate confidence measures, such as GOP. Then, based on a predefined threshold, phoneme pronunciation was recognized as correct or incorrect. Then, in the second stage, based on the set of recognized articulatory attributes, the system diagnosed the error and provided feedback to the learners. The authors addressed the scarcity of non-native data by applying cross-lingual transfer learning between the L1 and L2 languages of the learners. They also proposed using a multitask learning framework to train the acoustic model and articulatory model jointly. To test the performance of the proposed system, they used speech recordings of seven Japanese speakers, where each speaker uttered 850 English words. With the success of end-to-end ASR systems [27,28], several studies have been proposed for end-to-end MDD systems. For English learners, several end-to-end MDD systems were proposed, such as [6,29–31], where they used the hybrid CTC-attention model, CNN-RNN, CNN-RNN-CTC, and Transformer methods, respectively. For Mandarin learners, a hybrid CTC-attention method for MDD was proposed in [29,32,33].

In terms of databases, numerous benchmark corpora were released for CAPT systems for languages other than Arabic, such as L2-ARCTIC [34], CU-CHLOE [35], iCALL [36], and N4 NATO [37].

2.4. Limitations of Research on MDD for Non-Native Arabic Speech

From the aforementioned review, we observe the following research points regarding MDD for non-native Arabic speech. To the best of our knowledge, there is no non-native Arabic speech corpus that is fully annotated by experts and consists of the continuous speech of speakers from numerous nationalities. The second point is that applying deep learning techniques for MDD of non-native Arabic speech was investigated only in a few works, such as [23–26], and these studies limited to speech consisting of words or phonemes and not continuous speech. A third point is that, to the best of our knowledge, no work has been performed to provide feedback to non-native Arabic learners at the articulatory level.

Therefore, we attempt herein to address some of these research points by means of the following contributions. First, we develop a fully annotated non-native Arabic corpus that consists of continuous utterances of 62 non-native Arabic speakers from 20 nationalities and 120 native speakers. We complement this database with a synthesized non-native Arabic corpus using neural TTS. Second, we propose a fast deep-learning-based system suitable for real-time MDD for non-native Arabic speech. Moreover, this system not only locates and recognizes mispronounced phonemes from the whole utterance, but also provides informative articulation-level feedback to learners.

3. The Developed Corpora: Arabic-CAPT and Arabic-CAPT-S

3.1. Non-Native Arabic Speech Corpus (Arabic-CAPT)

We studied in detail the existing Arabic speech corpora that contain non-native speech regardless of whether they were designed for CAPT systems or other speech processing applications, such as speaker recognition and speech recognition. From this study, we found that the KSU speech corpus [21] was the most suitable to our research due to the following reasons.

- The database was developed for Arabic speaker recognition as a main target, but it was also designed to be useful for other applications, mainly speech recognition and speech processing of non-native Arabic speech.
- The corpus was recorded over three different sessions in three different environments and using four different channels.
- The corpus consisted of recording male and female speakers, where the male speakers included Saudi, Arab, and non-Arab speakers.

In our study, we focused on the recording of male speakers in session 1 in an office location, where the total number of speakers was 266 and 62 of them were non-native speakers from 20 different nationalities. Another interesting feature of this corpus is that each speaker uttered 16 lists; some lists were common between all speakers, and others were distinct. In our study, we selected only the eight common lists. Table 1 shows the canonical text and the corresponding phonetic transcription of some examples of the selected text.

We automatically segmented the utterances to sentences using the Aeneas segmentation tool [38]. Then, we manually corrected the output of Aeneas. This was performed by an Arabic expert who adjusted the time boundaries of the output of Aeneas using Praat software and provided us the correct boundaries. Finally, we used Montreal forced aligner (MFA) version 1.0.0 [39] to perform force alignment to generate the word and phoneme time boundaries. For this whole process, we converted the Arabic text of all utterances to a phoneme sequence using the Arabic phonetiser developed by N. Halabi [40,41]. Arabic language has 34 phonemes (28 consonants and 6 vowels), and for processing, we used the corresponding English symbols, as presented in Table 2.

Table 1. Examples of the selected text.

Canonical Text	Phonetic Transcription
[صِفْرْ وَاحِدْ إِثنانْ ثَلاثهْ أَرْبعهْ خَمْسَهْ سِتَه سَبعهْ ثَمَانِيَهْ تِسْعَهْ]	[Sifr wa2Hid HZithna2n thala2thh HZarbEh xamsah sitah sabEh thama2niyah tisEah]
[فَحْصْ فَحْمْ فَسْحْ فَصْمْ مَزْحْ مَغْصْ نِصْفْ نَهْشْ نَفْعْ نَفْسْ]	[faHS faHam fasH faSm mazH magS niSf nahsh nafE nafs]
[بِالوَالِدَينِ إِحْسَانًا]	[bilwa2lidayni HZiHsa2nan]
[اِستَقِمْ كَمَا أُمِرْتَ]	[HZistaqim kama HZumirta]
[هَلْ هَارَ]	[hal ha2ra]
[ضَمِنْتُ شَغَفَكُمْ]	[Damintu shagafakum]
[أَبْصَرَ ثُعبَاناً وَلَم يَظلِمْهُ]	[HZabSara thuEba2nan walam yaZlimhu]

Table 2. Arabic phonemes, IPA symbols, and the corresponding English symbols used in this research.

Arabic phoneme	ء	ب	ت	ث	ج	ح	خ	د	ذ	ر	ز	س	ش	ص	ض	ط	ظ
IPA symbol	ʔ	b	t	θ	ʒ	ħ	x	d	ð	r	z	s	ʃ	$s^ʕ$	$d^ʕ$	$t^ʕ$	$ð^ʕ$
English symbol	HZ	b	t	th	j	H	x	d	TH	r	z	s	sh	S	D	T	Z
Arabic phoneme	ع	غ	ف	ق	ك	ل	م	ن	ه	و	ي	فتحه	ضمة	كسره	الف مد	واو مد	ياء مد
IPA symbol	ʕ	ɣ	f	q	k	l	m	n	h	w	j	a	u	i	a:	u:	i:
English symbol	E	g	f	q	k	l	m	n	h	w	y	a	u	i	a2	u2	i2

The previous three steps were performed for the selected utterances of all 266 male speakers (native and non-native) in session 1 from the KSU speech corpus. For speech annotation, we focused only on the speech of non-native Arabic speakers, which included 62 speakers from 20 different nationalities. Figure 1 shows the nationalities and the number of speakers from each nationality. We anticipate from the diversity of the speakers' nationalities that the produced corpus will significantly contribute to the research community in the Arabic-CAPT systems.

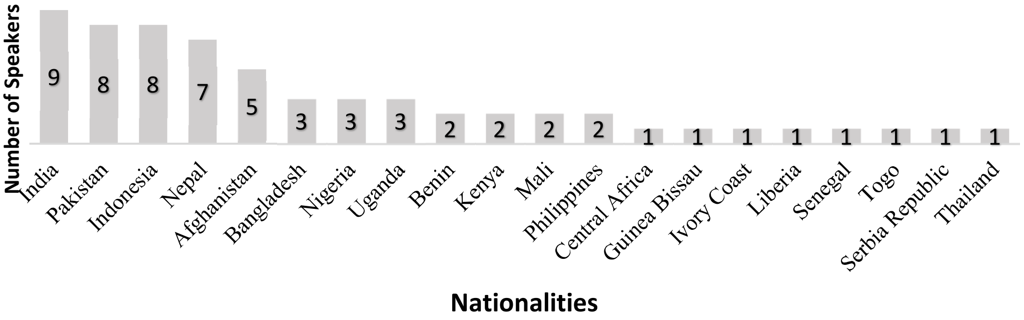


Figure 1. Nationality distribution of all speakers.

The annotation process is a core task for the corpora of CAPT applications, so we asked three native Arabic experts to annotate the utterances of all non-native speakers and decide the type of error: insertion, substitution, and deletion errors. We followed the annotation scheme of L2-ARCTIC [34] and annotated the substitution, insertion, and deletion errors in the following forms: (canonical phoneme, substituted phoneme, S), (@, inserted phoneme, I), and (deleted phoneme, @, D), respectively.

3.2. Synthesized Non-Native Arabic Speech Corpus (Arabic-CAPT-S)

Data augmentation techniques have been used to overcome the scarcity of training data [42], to enhance the generalization of deep models [43], and to reduce overfitting [44]. In ASR systems, the synthesized speech was used as augmented data to solve the problem of lack in labeled speech corpora and to enhance the performance of speech recognition systems [45,46]. Recently, an investigation of using synthesized speech to augment the speech corpus was performed by the authors of [47] who proposed using neural TTS to augment non-native English data for lexical stress detection by means of synthesized words with correct and incorrect lexical stress. The authors stated that their study was the first effort in lexical error detection that used a neural TTS for data augmentation.

Due to the lack of non-native Arabic corpora, in general, and the low number of pronunciation errors in our Arabic-CAPT corpus, we generated synthesized non-native Arabic speech. Hence, we trained the recent neural TTS, FastSpeech2, to produce high-quality non-native Arabic speech that contains some predefined substitution errors [48]. We selected the FastSpeech2 model because it is state-of-the-art, fast, and supports multi-speaker embedding, so we could generate a synthesized speech using the style of all non-native speakers of our Arabic-CAPT corpus. Because the size of the Arabic-CAPT corpus is not efficient in training FastSpeech2 from scratch, we trained FastSpeech2 using an Arabic native corpus. Hence, we selected 5 h of recording of Saudi speakers from the KSU speech database to train FastSpeech2. Then, we fine-tuned the model using the Arabic-CAPT corpus. Finally, we modified the canonical text of the Arabic-CAPT corpus by embedding the most common pronunciation errors of non-native Arabic speakers to produce text with substitution errors. Note that we focused on embedding substitution errors in the synthesized corpus because they are common errors in learning Arabic. Then, the generated text was fed to the trained TTS to generate high-quality synthesized speech. Finally, we used the generated transcript and synthesized audio as input to MFA to segment the synthesized Arabic-CAPT-S corpus at the word and phoneme levels.

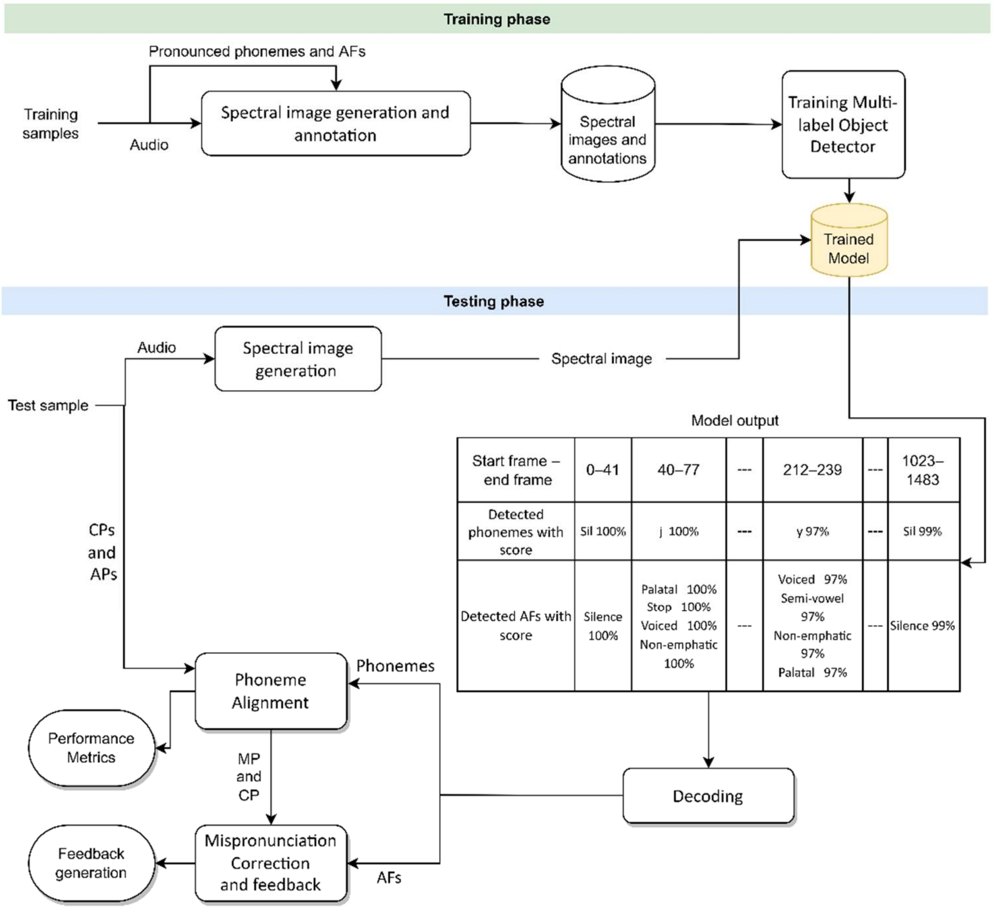
Table 3 shows the total number of phonemes, not including silence, and the total number of errors for the Arabic-CAPT and Arabic-CAPT-S. For the Arabic-CAPT, we show the number of each of the three types of error, while Arabic-CAPT-S has only substitution errors. We noticed that the number of pronunciation errors in the Arabic-CAPT corpus was 2899, which represented 5.1% of the total number of phonemes in the corpus, where substitution and insertion errors were more frequent than deletion errors. The number of substitution errors in the synthesized Arabic-CAPT-S corpus was 17,422, which represents 6.4% of the total number of phonemes, near the percentage of errors in the Arabic-CAPT. In terms of duration in hours, we can see from Table 3 that the Arabic-CAPT-S consisted of 7.11 recording hours and the Arabic-CAPT consisted of only 2.36 h. We expect that training the proposed system using real and synthesized corpora will make it more generalized and able to detect the most common pronunciation errors of non-native Arabic learners, especially for the most important type of pronunciation error, which is the substitution error.

Table 3. Statistics of the developed corpora.

		Arabic-CAPT	**Arabic-CAPT-S**
Type of data		Real	Synthetic
Speakers		62	62
Utterances		1611	7254
Recording hours		2.36	7.11
Correct phonemes		54,171	255,502
Substitution Errors	#	1080	17,422
	%	2	6.4
Insertions Errors	#	1139	-
	%	2.1	-
Deletion Errors	#	690	-
	%	1.3	-
Total Errors	#	2899	17,422
	%	5.1	6.4

4. The Proposed Object Detection-Based MDD (MDD-Object)

In this section, we present the details of our proposed system, where we investigate the use of a multi-label object detection technique for the simultaneous recognition of phonemes and their corresponding AFs from the spectral image of the spoken utterance, and we call it MDD-Object. From the recognized phonemes, we can detect and diagnose pronunciation errors. From the recognized AFs, we can provide informative feedback to the learner about the error, place, and manner of articulation. Our idea of training the phoneme and AFs simultaneously coincides with the idea in [16], where a multitask learning framework for MDD was used for phonetic modeling and articulatory modeling. The proposed system and its different modules are shown in Figure 2. The first module performs speech-to-image conversion and annotation, which produces three-channel spectral images for the speech utterances and the corresponding multi-label bounding boxes (in the training phase). The second module, which is the core of the proposed system, uses a multi-label object detection technique to detect the phonemes and AFs as objects in the spectral image. In the third module, we propose applying the greedy decoding technique to the outputs of the multi-label object detection module to remove the duplicates phonemes and AFs and deal with the overlap between the bounding boxes. In the fourth module, we align the canonical, detected, and annotated sequence of phonemes to identify and diagnose the pronunciation errors and calculate the performance metrics of the proposed system. Finally, in the fifth module, once we detected the mispronounced phoneme, we used the corresponding detected AFs to provide feedback at the articulatory level to the learners to enable them to correct their pronunciation. In the following sections, we discuss the details of each module. Moreover, at the end of this section, we show the details of applying end-to-end MDD to non-native Arabic speech.


Training phase
Pronounced phonemes and AFs
Training samples
Audio
Spectral image generation and annotation
Spectral images and annotations
Training Multi-label Object Detector
Trained Model
Testing phase
Audio
Spectral image generation
Spectral image
Test sample
CPs and APs
Model output
Phonemes
Phoneme Alignment
Performance Metrics
MP and CP
Decoding
Feedback generation
Mispronunciation Correction and feedback
AFs

Start frame – end frame	0–41	40–77	---	212–239	---	1023–1483
Detected phonemes with score	Sil 100%	j 100%	---	y 97%	---	Sil 99%
Detected AFs with score	Silence 100%	Palatal 100% Stop 100% Voiced 100% Non-emphatic 100%	---	Voiced 97% Semi-vowel 97% Non-emphatic 97% Palatal 97%	---	Silence 99%

Figure 2. General overview of the MDD-Object system. CPs: canonical phonemes; APs: annotated phonemes; MP: mispronounced phonemes.

4.1. Spectral Image Generation and Annotation

We propose addressing the phonemes and the associated AFs of utterances as objects in spectral images and applying object detection techniques to detect them. To accomplish this, we converted the speech utterances into three-channel spectral images to fulfill the requirements of object detectors, as presented in more detail in our previous work [14], where the log Mel-spectrogram was calculated, and its delta and delta–delta were also calculated and appended to produce a three-channel image that is visually shown as an RGB image.

To label the bounding boxes, in order to train the object detector, we used each pronounced phoneme and its sequence of AFs as multi-label for the bounding boxes. The process of representing the phonemes by their corresponding AFs for an end-to-end automatic speech recognition system was performed by some researchers, such as in [49] for the English language. In the literature, few researchers have investigated extracting Arabic phonetics distinctive features (PDF), such as [15,50,51]. In this research, we focused on selected AFs that are important to teach the correct pronunciation to non-native Arabic learners [52]. We used the AFs of place and manner of articulation that were presented in [53], and we grouped them into five categories: place, manner, manner–voice, and manner–emphatic for consonant phonemes and vowel category for vowel phonemes so that all classes inside the same category are disjointed, as shown in Table 4.

Table 4. AF categories, classes of each category, and phonemes and their IPA of each class used in this study.

AF Category (# of Classes)	Class	Arabic Phonemes	IPA
Place (10)	Bilabial	م، ب، و	m, b, w
	Labio-dental	ف	f
	Inter-dental	ث، ذ، ظ	θ, ð, ð^{ʕ}
	Alveo-dental	ت، د، ر، ز، س، ص، ض، ط ل، ن	t, d, r, z, s, s^{ʕ}, d^{ʕ}, t^{ʕ}, l, n
	Palatal	ي	j
	Alveo-palatal	ج، ش	ʒ, ʃ
	Velar	ك	k
	Uvular	خ، غ، ق	x, ɣ, q
	Pharyngeal	ح، ع	ħ, ^{ʕ}
	Glottal	ء، هـ	ʔ, h
Manner (6)	Stop	ء، ب، ت، د، ض، ط، ق، ك	ʔ, b, t, d, d^{ʕ}, t^{ʕ}, q, k
	Fricative	ث، ح، خ، ذ، ز، س، ش، ص، ظ، ع، غ، ف، هـ	θ, ħ, x, ð, z, s, ʃ, s^{ʕ}, ð^{ʕ}, ^{ʕ}, ɣ, f, h
	Affricate	ج	ʒ
	Nasal	م، ن	m, n
	Laterals	ل	l
	Trill	ر	r
Manner–Voice (2)	Voiced	ب، ج، د، ذ، ر، ز، ض، ظ، ع، غ، ل، م، ن، و، ي	b, ʒ, d, ð, r, z, d^{ʕ}, ð^{ʕ}, ^{ʕ}, ɣ, l, m, n, w, j
	Unvoiced	ء، ت، ث، ح، خ، س، ش، ص، ط، ف، ق، ك، هـ	ʔ, t, θ, ħ, x, s, ʃ, s^{ʕ}, t^{ʕ}, f, q, k, h
Manner–Emphatic (3)	Emphatic-stop	ض، ط	d^{ʕ}, t^{ʕ}
	Emphatic-fricative	ص، ظ	s^{ʕ}, ð^{ʕ}
	Non-emphatic	ء، ب، ت، ث، ج، ح، خ، د، ذ، ر، ز، س، ش، ع، غ، ف، ق، ك، ل، م، ن، هـ، و، ي	ʔ, b, t, θ, ʒ, ħ, x, d, ð, r, z, s, ʃ, ^{ʕ}, ɣ, f, q, k, l, m, n, h, w, j
Vowel (2)	Vowel	فتحة، ضمة، كسرة، الف مد، واو مد، ياء مد	a, u, I, a:, u:, i:
	Semi-vowel	و، ي	w, j

For each phoneme, we extracted the corresponding AF labels based on the mapping of Table 4. For example, the Arabic word "فَحْصْ" consists of the following phonemes (/f/, /a/, /H/, /S/), and each phoneme is represented by multi-label AFs, as presented in Table 5. Then, using the time boundary of each phoneme, we calculated the bounding box coordinates of the phoneme and its corresponding AFs to create the annotation file of each utterance, as we presented in [14].

The spectral images and the corresponding annotations were used for training the model. To test the model, we used only the audio utterances to create the spectral image, and then we fed the spectral image to the trained model, as shown in the test part of Figure 2.

Table 5. Examples of mapping phonemes to their corresponding AFs.

		AFs Categories				
		Place	**Manner**	**Manner–Voice**	**Manner–Emphatic**	**Vowel**
Phonemes	/f/	Labio-dental	Fricative	Unvoiced	Non-emphatic	-
	/a/	-	-	-	-	Vowel
	/H/	Pharyngeal	Fricative	Unvoiced	Non-emphatic	-
	/S/	Alveo-dental	Fricative	Unvoiced	Emphatic-fricative	-

4.2. Multi-Label Object Detector: Selection, Optimization, and Training

In this section, we present the details of using multi-label object detection for phoneme and AF recognition. We also justify our selection of the object detector and present the process of choosing the detector hyper-parameters using a GA to enhance the performance of the detector. The training process of the various proposed models is also shown.

4.2.1. Selection of Object Detector and Hyper-Parameter Optimization

To select the object detector for our proposed system, we considered two conditions: the system should operate in real-time and should be able to detect multi-label objects. Based on our observations in our published work [15], we selected the tiny version of the third YOLO (YOLOv3-tiny) as the detector of our proposed system, since it best satisfied the above two conditions.

The YOLO detector was proposed for computer vision tasks, so we needed to fine-tune the detector's parameters to adapt it to work with speech applications. Due to the existence of a vast number of parameters and the large number of possibilities, it is difficult to apply simple grid search algorithms to select the best hyper-parameters. Therefore, we proposed using a GA for selecting the optimal parameters of the YOLOv3-tiny detector. Table 6 shows the hyper-parameters that selected using the GA.

Table 6. Hyper-parameters of YOLO architecture.

Parameter	**Default Values**	**Best Values Selected by GA**
Learning rate	0.001	0.000482
Momentum	0.9	0.898543
Decay	0.0005	0.000567
Flip	0	False
blur	0	True
Gaussian_noise	0	True
mixup	0	False
mosaic	0	False
Saturation	1.5	0.709779
Exposure	1.5	0.760187
Hue	0.1	0.039356
Activation function	leaky	swish
Filter size	3	7
Maximum objects	200	226
Jitter	0.3	0.100393
Ignore threshold	0.7	0.551836
Anchors	(4,7), (7,15), (13,25), (25,42), (41,67), (75,49), (91,162), (158,205), (250,332)	(9,32), (14,32), (19,32), (24,32), (30,32), (40,32), (61,32), (127,32), (325,32)

To perform the hyper-parameter selection using GA, we used 20 population sizes, where each population corresponds to a YOLO architecture. The created model for each architecture was trained for 2K iterations for non-native phoneme recognition using the training and validation sets of the developed Arabic-CAPT corpus. We ran the GA for 50 generations. We showed in our previous study [14] that there is a direct relationship between the mean average precision (mAP), which is a famous metric in the object detection area, and the phoneme error rate (PER), which is a famous metric in the speech area, so we used the mAP as the fitness function. The best hyper-parameters selected by GA are shown in the last column of Table 6.

4.2.2. Model Training

We demonstrated in our previous work [14] the effectiveness of transfer learning between corpora of the same language or corpora of different languages. Hence, we trained different models from the selected detector with and without different types of transfer learning. In the following, we present the different models that we investigated, and then we present how we trained these models.

We used the detector with default parameters as a baseline, and we called it the MDD-Object model. We trained the MDD-Object model using only real non-native speech without any initial training. In the second model, MDD-Object-G, which was produced from the optimization process in the previous section, we trained the model from scratch without any initial training using the developed Arabic-CAPT corpus. Then, we trained various versions of it by initially training the detector using native speech, synthesized speech, and a combination of native and synthesized speech, and we called these versions MDD-Object-G/N, MDD-Object-G/S, and MDD-Object-G/NS, respectively. In the MDD-Object-G/N model, we initially trained using native speech, and then we transferred the weights by freezing the first four layers of the backbone network of the detector and fine-tune the remaining layers using the real non-native Arabic-CAPT corpus. In MDD-Object-G/S, we trained the model using synthesized speech from the Arabic-CAPT-S corpus, which is approximately four times the size of the original Arabic-CAPT corpus, as shown in Table 3. Then, we transferred the weights of the detector and fine-tuned them using the real Arabic-CAPT corpus. Finally, in MDD-Object-G/NS, we initially trained the model using the combination of native speech and synthesized speech. Then, we also transferred the weights of the detector and fine-tuned them using the real Arabic-CAPT corpus.

For the third model, we observed in our previous work [14] that the long version of YOLOv3, which is called YOLOv3, achieved better results than the tiny version; hence, we also investigated this detector using the same selected hyper-parameters as in the previous section, and we called this model MDD-Object-G-Large/NS. We also initially trained it using the combination of native speech and synthesized speech. Then, we transferred the weights of the detector and fine-tuned them using the Arabic-CAPT corpus. Table 7 shows the type of speech that was used for the initial training and fine-tuning phases for each of the proposed models.

Table 7. Speech type for the proposed models. N: native speech; S: synthesized speech; NN: non-native speech.

	Initial Training Phase		**Fine Tuning Phase**		
	Training Set	**Validation Set**	**Training Set**	**Validation Set**	**Testing Set**
MDD-object (baseline)	-	-	NN	NN	NN
MDD-object-G	-	-			
MDD-object-G/N	N	N			
MDD-object-G/S	S	S			
MDD-object-G/NS	N+S	N+S			
MDD-object-G-Large/NS	N+S	N+S			

For a fair comparison, we fixed some parameters among all models, such as the number of epochs, batch size, and width/height of the network.

Table 8 presents the statistics of non-native, synthesized, and native corpora that were used for the training, validation, and testing of the proposed models. For the Arabic-CAPT corpus, we used 75% of speakers for training and validation and the remaining 25% for testing. All speakers in training, validation, and testing were distinct. We attempted to map the diversity of the nationalities in the database in the test set; hence, the 15 speakers of the test set belonged to 12 different nationalities. In Arabic-CAPT-S, we used the synthesized speech of the speakers of the training and validation sets of Arabic-CAPT, as shown in Table 8. In native speech, we used all 146 Saudi speakers in session 1 from the KSU speech database and divided them into 132 for training and 14 for validation.

Table 8. Specifications of the developed corpora used in the training and testing phases.

	Arabic-CAPT			**Arabic-CAPT-S**		**Native Speech**	
Set	**Train**	**Valid**	**Test**	**Train**	**Valid**	**Train**	**Valid**
Speakers	42	5	15	42	5	132	14
Utterances	1091	130	390	4914	585	3426	361
Duration (Hours)	1.65	0.16	0.54	4.79	0.6	4.11	0.43

4.3. Decoder

The output of the previous module is a sequence of bounding boxes, where each box has a multi-label (i.e., phonemes and AFs), and some of these boxes overlap and lead to duplicate phonemes and AFs. To deal with this, we propose using decoding techniques that are used in sequence-to-sequence acoustic models, such as CTC. These decoding techniques need a matrix of probabilities for the phonemes of the frames, and so to accomplish this, we started by converting the confidence scores matrix to a matrix of probabilities by applying softmax so that the sum of the probabilities in each frame is equal to one. In this study, we investigated the fast and simplest decoding technique, which is greedy search decoding. The greedy decoder was applied on the matrix of probabilities and selected the best recognition in each time step (i.e., frames F), leading to the best path of decoding calculated from the output of detector (YOLO) given the image of spectrogram (X) [54].

4.4. Phoneme Alignment and Performance Metrics Evaluation

In this step, we aligned the detected phonemes of the proposed system, the canonical phoneme, and the annotated phoneme of the human experts to calculate the system performance and evaluation metrics, as shown in Figure 2. We calculated the performance metrics of the proposed models for the non-native phoneme recognition task and MDD task as follows:

For the phoneme recognition task, we used a well-known metric in this field, PER, and calculated it as in Equation (1) [55].

$$\mathrm{PER} = 100 - \frac{\mathrm{N} - \mathrm{S} - \mathrm{D} - \mathrm{I}}{\mathrm{N}} \quad (1)$$

where N, S, D, and I are the number of samples, substitution errors, deletion errors, and insertion errors, respectively. Once we detected any mispronounced phonemes, these phonemes were sent to the mispronunciation correction and feedback module to provide corrective feedback.

For the MDD task, we used a hierarchical evaluation that was presented in [56] and used in many MDD studies such as [36,57], where the phonemes of the sequence annotated by the experts were classified as correct and mispronounced. By comparing the output of our proposed models with the annotations of the experts, we obtained four cases: true acceptance (TA), false acceptance (FA), false rejection (FR), and true rejection (TR). TR was divided into correct diagnosis (CD) and diagnosis error (DE). CD corresponded to the annotator agreeing with system in the recognized mispronounced phoneme, while DE corresponded to the annotator disagreeing with system in the recognized mispronounced phoneme. We calculated the performance metrics of the MDD task using the following equations:

$$\text{Precision} = \frac{\text{TR}}{\text{TR} + \text{FR}} \tag{2}$$

$$\text{Recall} = \frac{\text{TR}}{\text{TR} + \text{FA}} \tag{3}$$

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

$$\text{FAR} = 1 - \text{Recall} \tag{5}$$

$$\text{FRR} = \frac{\text{FR}}{\text{TA} + \text{FR}} \tag{6}$$

$$\text{Diagnosis Accuracy (DA)} = \frac{\text{CD}}{\text{CD} + \text{DE}} \tag{7}$$

4.5. Mispronunciation Correction and Feedback

We aimed in this phase to correct the mispronunciation of the learner and provide feedback at the articulatory level. The mispronounced phonemes are detected in the previous step and the purpose of this step is to compare the detected AFs of the mispronounced phoneme with the AFs of the canonical phoneme. From this comparison, we provided corrective feedback to the learner at the articulatory level. As shown in the example in Figure 3, the canonical phoneme (/S/) has the following AF classes, fricative, unvoiced, emphatic-fricative, and alveo-dental, and the detected phoneme /s/ has the detected AF classes, fricative, unvoiced, non-emphatic, and alveo-dental. Hence, the learner had a problem with an emphatic class, while there was no problem with other classes. Therefore, feedback can be provided and a 2D/3D animation of the correct pronunciation of phoneme /S/ can be shown to the learners.

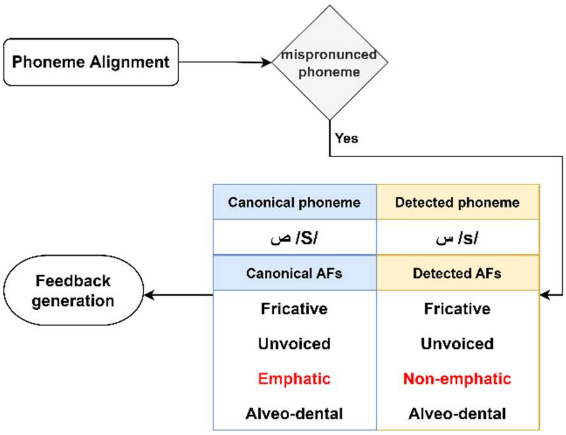


Figure 3. An example of feedback generation.

An interesting point is that our system provides the score of detection of each phoneme and AF, which is called the objectness score in YOLO. This score can be used as a measure of the goodness of pronunciation of each pronounced phoneme.

5. Application of End-to-End MDD to Non-Native Arabic Speech (MDD-E2E)

End-to-end deep learning systems have achieved promising results in automatic speech recognition [58,59] and MDD [29,30,33]. Hence, to evaluate the performance of the proposed system, we compared its performance with a state-of-the-art end-to-end model using the developed Arabic-CAPT corpus. Many of the researchers on end-to-end MDD methods [29,60–62], used CNN-RNN-CTC [30]; hence, we selected it as the end-to-end method for comparison. The original CNN-RNN-CTC was proposed for the end-to-end MDD task for Chinese learners of English using the CU-CHLOE corpus. We used the implementation provided by [63] to train and evaluate the CNN-RNN-CTC using our developed corpora. To the best of our knowledge, this is the first investigation of applying end-to-end deep learning techniques in an MDD system for non-native Arabic speech. Note that, because the CNN-RNN-CTC model is not a multi-label model, we used it for the phoneme recognition task only and not the AF detection task.

Furthermore, we enhanced the performance of the CNN-RNN-CTC model by using different combinations of native, synthesized, and non-native speech. We trained the baseline model CNN-RNN-CTC using only the non-native Arabic-CAPT corpus, while we trained other models, namely CNN-RNN-CTC/N, CNN-RNN-CTC/T, and CNN-RNN-CTC/NT, using combinations of the data, as shown in Table 9.

Table 9. End-to-End MDD models for non-native Arabic speech. N: native speech; S: synthesized speech; NN: non-native speech.

	Training Set	Validation Set	Testing Set
CNN-RNN-CTC	NN	NN	NN
CNN-RNN-CTC/N	N + NN	NN	NN
CNN-RNN-CTC/S	S + NN	NN	NN
CNN-RNN-CTC/NS	N + S + NN	NN	NN

6. Application of Decision Level Fusion between MDD-Object and MDD-E2E (MDD-Object-E2E)

Fusion techniques have been successfully used and applied to improve the generalization of the machine learning models [64]. Our MDD-Object system and MDD-E2E system have different structures and are based on different principles, and hence to benefit from the performance of the two systems, we propose applying the fusion technique between the MDD-Object system and MDD-E2E system. We used the decision level fusion between the best model of the MDD-Object system, which is MDD-Object-G-Large/NS, and the best model of the E2E-MDD system, which is CNN-RNN-CTC/NS. To achieve this, we calculated the weighted average of the output probabilities of the two models before decoding the final output (i.e., sequence of phonemes).

To make the output of the two models more consistent in the number of frames in the output, we set the frame length and number of Mels of the CNN-RNN-CTC system to the same values of the MDD-Object system. This change in the CNN-RNN-CTC system made it necessary to change the size of the input RNN layer. These changes improved the performance of the CNN-RNN-CTC system, as will be shown later.

Another point that we addressed to achieve fusion between the two systems is that the CNN-RNN-CTC model reduces the number of frames by a factor of 4. We dealt with this by down sampling the output of the MDD-Object by a factor of 4. Next, we applied a weighted average to the probabilities of the two models as in Equation (8).

$$Probs_{AVG} = \alpha \times Probs_{E2E} + (1 - \alpha) \times Probs_{YOLO} \tag{8}$$

where *Probs* is the logsoftmax array output with the dimensions number of frames × number of phonemes. To find the optimum weight (α), we used a basic grid search to find the best value of alpha that resulted in the lowest phoneme error rate from the following values of α: (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1). From Figure 4, we chose α to be 0.1.

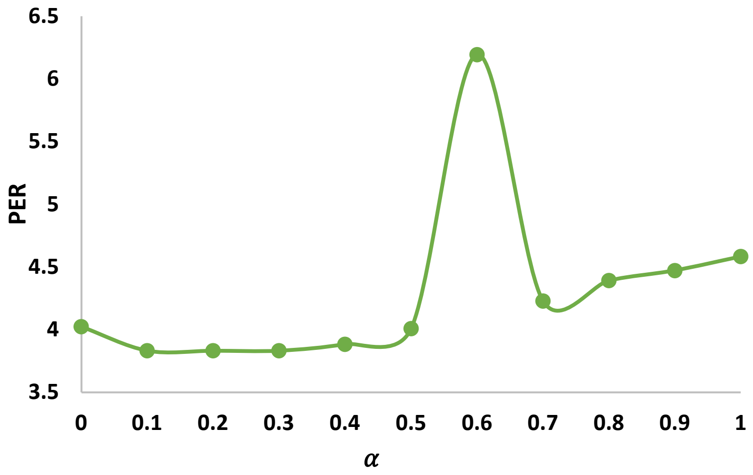


Figure 4. Selection of the parameter (α) of the weighted average.

7. Experimental Results

In this section, we present the performance of the proposed system and its different models for MDD of non-native Arabic speech. We first present the performance of the proposed models and the baseline model for the non-native phoneme recognition task. Next, we present the performance of the proposed models for the MDD task. In both sections, we compare our different models with an end-to-end system and our different enhanced versions of it. Finally, in Section 7.3, we present the performance of the proposed system for the AF detection task.

7.1. Results of the Non-Native Phoneme Recognition Task

The performance of MDD-Object models, the MDD-E2E models, and the MDD-Object-E2E model for the non-native phoneme recognition task is presented in Table 10. For our proposed MDD-Object system, the PER of the baseline MDD-Object model was 7.56%, as shown in Table 10. This value improved to 4.93% when we used the GA to select the optimal parameters of the model in MDD-Object-G. When we used native and synthesized speech for the initial training of the model, the PERs improved to 4.75% and 4.71% using MDD-Object-G/N and MDD-Object-G/S, respectively. The result of the MDD-Object-G/S confirmed our anticipation of the benefit of using synthesized speech in transfer learning to solve the scarcity of non-native speech. When we used the combination of native and synthesized speech for the initial training of MDD-Object-G/NS, we achieved PERs of 4.54% resulting in relative improvements of 40% compared to the baseline. Though using the native and synthesized speech did not offer a better result than MDD-Object-G/N, it resulted in a better performance for the MDD task, as will be shown later in Section 7.2. The best performance of our proposed system was achieved by MDD-Object-G-Large/NS, which achieved 4.05%, resulting in relative improvements of 46.42% compared to the baseline.

For the MDD-E2E system, the baseline model CNN-RNN-CTC achieved 8.93% PER. This value decreased to 5.17%, resulting in a relative improvement of 42.1% when we added native speech for training the CNN-RNN-CTC/N model. The same observation is made when we added synthesized speech for training the CNN-RNN-CTC/S model, where we achieved a PER of 6.17%, resulting in a relative improvement of 30.9% compared to the baseline. This improvement confirms our anticipation of the usefulness of synthesized speech for solving the scarcity of non-native speech. When we added native speech and synthesized speech for training the CNN-RNN-CTC/NS model, we achieved a 4.59% PER, which was a relative improvement of 48.6% compared to the baseline.

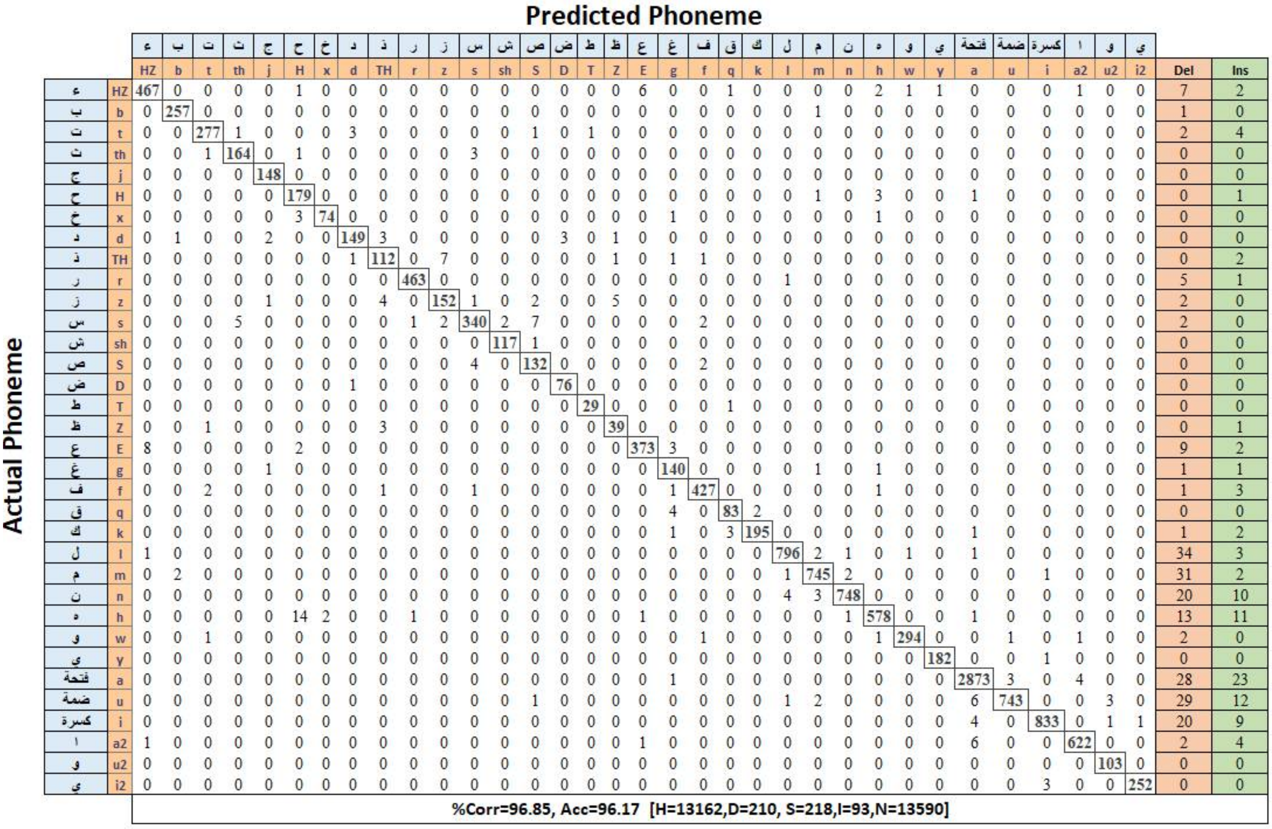
It should be noted that our proposed model MDD-Object-G-Large/NS achieved better results than all the models of the state-of-the-art end-to-end technique CNN-RNN-CTC.

Table 10. PER of the proposed MDD-object models MDD-E2E models and the fusion model.

System	Model	PER (%)
MDD-Object	MDD-Object (baseline)	7.56
	MDD-Object-G	4.93
	MDD-Object-G/N	4.75
	MDD-Object-G/S	4.71
	MDD-Object-G/NS	4.54
	MDD-Object-G-Large/NS	4.05
MDD-E2E	CNN-RNN-CTC (baseline)	8.93
	CNN-RNN-CTC/N	5.17
	CNN-RNN-CTC/S	6.17
	CNN-RNN-CTC/NS	4.59
MDD-Object-E2E	YOLO-CNN-RNN-CTC	3.83

The fusion system between the CNN-RNN-CTC/NS and MDD-Object-G-Large/NS models achieved the best PER of 3.83% amounting to 16.5% relative improvement compared to the performance of CNN-RNN-CTC/NS and 5.4% relative improvement compared to the performance of MDD-Object-G-Large/NS. This result demonstrates the benefit of applying the decision-level fusion.

For a better analysis of the performance of each phoneme, we present in Figure 5 the confusion matrix of our best model YOLO-CNN-RNN-CTC using the test set of the Arabic-CAPT corpus. The total number of phonemes in the test set, without leading and trailing silence, is 13590 phonemes belonging to 390 utterances.



Predicted Phoneme

Actual Phoneme

		ء	ب	ت	ث	ج	ح	خ	د	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ	ف	ق	ك	ل	م	ن	ه	و	ي	فتحة	ضمة	كسرة	ا	و	ي		
		HZ	b	t	th	j	H	x	d	TH	r	z	s	sh	S	D	T	Z	E	g	f	q	k	l	m	n	h	w	y	a	u	i	a2	u2	i2	Del	Ins
ء	HZ	467	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	0	0	1	0	0	0	0	2	1	1	0	0	0	1	0	0	7	2
ب	b	0	257	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
ت	t	0	0	277	1	0	0	0	3	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	4
ث	th	0	0	1	164	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ج	j	0	0	0	0	148	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ح	H	0	0	0	0	0	179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	3	0	0	1	0	0	0	0	0	0	1
خ	x	0	0	0	0	0	3	74	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
د	d	0	1	0	0	2	0	0	149	3	0	0	0	0	0	3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ذ	TH	0	0	0	0	0	0	0	1	112	0	7	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
ر	r	0	0	0	0	0	0	0	0	0	463	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	5	1
ز	z	0	0	0	0	1	0	0	0	4	0	152	1	0	2	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
س	s	0	0	0	5	0	0	0	0	0	1	2	340	2	7	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
ش	sh	0	0	0	0	0	0	0	0	0	0	0	0	117	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ص	S	0	0	0	0	0	0	0	0	0	0	0	4	0	132	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ض	D	0	0	0	0	0	0	0	1	0	0	0	0	0	0	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ط	T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	29	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ظ	Z	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ع	E	8	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	373	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	2
غ	g	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	140	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	1
ف	f	0	0	2	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	427	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	3
ق	q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	83	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ك	k	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	3	195	0	0	0	0	0	0	1	0	0	0	0	0	1	2
ل	l	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	796	2	1	0	1	0	1	0	0	0	0	0	34	3
م	m	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	745	2	0	0	0	0	0	1	0	0	0	31	2
ن	n	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	3	748	0	0	0	0	0	0	0	0	0	20	10
ه	h	0	0	0	0	0	14	2	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	578	0	0	1	0	0	0	0	0	13	11
و	w	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	294	0	0	1	0	1	0	0	2	0
ي	y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	182	0	0	1	0	0	0	0	0
فتحة	a	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	2873	3	0	4	0	0	28	23
ضمة	u	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	2	0	0	0	0	6	743	0	0	3	0	29	12
كسرة	i	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	833	0	1	1	20	9
ا	a2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	6	0	0	622	0	0	2	4
و	u2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	103	0	0	0
ي	i2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	252	0	0

%Corr=96.85, Acc=96.17 [H=13162,D=210, S=218,I=93,N=13590]

Figure 5. Confusion matrix of the phoneme detection task using the YOLO-CNN-RNN-CTC model.

The confusion matrix in Figure 5 asserts the effectiveness of the proposed system, where most of the phonemes were classified correctly. The percentage of confusion was very small where confusion occurred between two similar Arabic phonemes. For example, the phoneme /s/ was wrongly classified seven times as /S/ and phoneme /S/ was wrongly classified four times as /s/, where the two phonemes had the same place, manner, and manner–voice, and they differ only by manner–emphatic. The same observation occurs with the phoneme /h/, which was wrongly classified 14 times as /H/.

To study the effect of the nationality on the PER, we present in Figure 6 the PER for each speaker of the test set using our best model YOLO-CNN-RNN-CTC. The four worst values in PER were for speakers NS260, NS255, NS253, and NS252, who were from Togo, Guinea Bissau, Kenya, and Liberia, respectively. Speakers of these nationalities were not included in the speakers of the training set, except for the speaker from Kenya. Nonetheless, speakers NS250 and NS251 were from the Ivory Coast and Senegal, and the system had PERs of 4.9% and 3.1% for them, respectively, although there were no speakers from these nationalities in the training set. Achieving these values without any speaker from these nationalities in the training dataset shows the effectiveness of our system.

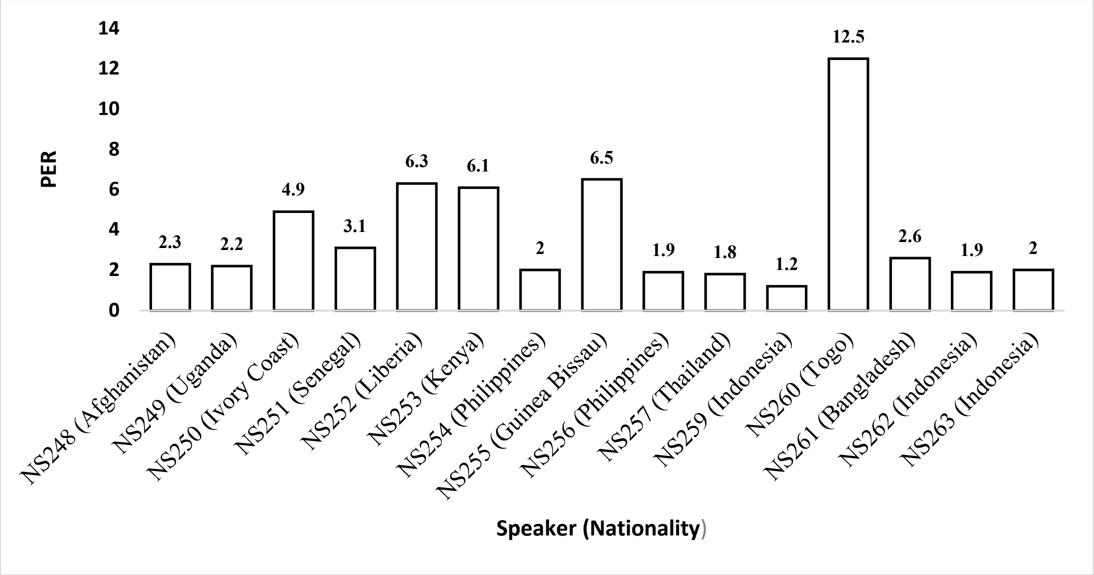


Figure 6. PER for each speaker using our best model: YOLO-CNN-RNN-CTC.

7.2. Results of MDD Task

In Table 11, we present the performance of the various models of the proposed MDD-Object system, MDD-E2E system, and their fusion for the MDD task.

For the MDD-Object system, we can see that the results for our baseline model MDD-Object for FAR, FRR, and P were 29.33%, 5.26%, and 40.29%, respectively. These results improved to 24.16%, 2.04%, and 65.06%, respectively, using our best model MDD-Object-G-Large/NS. Our baseline model obtained 76.99% diagnostic accuracy (DA), while our best model MDD-Object-G-Large/NS achieved a DA of 84.97%. Our best model MDD-Object-G-Large/NS obtained a 70.04% F1-score with relative improvement of 36.47% compared to our baseline model which achieved a F1-score of 51.32%.

For the MDD-E2E system, we can see from Table 11 that the CNN-RNN-CTC/NS model achieved the best result among the CNN-RNN-CTC models in FAR, FRR, P, and F1 with values of 24.92%, 1.91%, 66.31%, and 70.42%, respectively. The model also achieved the best result in the mispronunciation diagnosis, where the DA was 84.01%.

Table 11. MDD results of the proposed models and the baselines.

System	Model	TA (%)	FAR (%)	FRR (%)	DA (%)	DER (%)	P (%)	R (%)	F1 (%)
MDD-Object	MDD-object (baseline)	94.74	29.33	5.26	76.99	23.01	40.29	70.67	51.32
	MDD-object-G	97.55	29.79	2.45	82.68	17.32	59.0	70.21	64.12
	MDD-object-G/N	97.99	32.83	2.01	81.22	18.78	62.70	67.17	64.86
	MDD-object-G/S	97.63	30.55	2.37	81.62	18.38	59.51	69.45	64.10
	MDD-object-G/NS	97.84	28.88	2.16	84.19	15.81	62.32	71.12	66.43
	MDD-object-G-Large/NS	97.96	24.16	2.04	84.97	15.03	65.06	75.84	70.04
MDD-E2E	CNN-RNN-CTC (baseline)	94.14	20.67	5.86	75.10	24.9	40.47	79.33	53.59
	CNN-RNN-CTC/N	97.62	24.32	2.38	82.53	17.47	61.48	75.68	67.85
	CNN-RNN-CTC/S	96.44	23.40	3.56	82.74	17.26	51.91	76.60	61.88
	CNN-RNN-CTC/NS	98.09	24.92	1.91	84.01	15.99	66.31	75.08	70.42
MDD-object-E2E	YOLO-CNN-RNN-CTC	98.12	25.08	1.88	85.19	14.81	66.62	74.92	70.53

In general, we can see from Table 11 the improvement in the performance of the systems due to using different enhancements of fine-tuning the hyper-parameters and training using synthesized and native speech. Moreover, we noticed that our fusion system achieved the best performance in all metrics, which indicates the usefulness of applying fusion.

It should be noted that in educational applications such as CAPT systems, achieving low FRR is more important, as discussed in [31,65]. Our best model of MDD-Object system achieved the best trade-off between FAR and FRR.

7.3. Result of AF Detection Task

In this section, we present the performance of the proposed model for the AF detection task. As the performance metric, we used the detection error rate (DER), which was used in some previous work, such as [49]. Figure 7 shows the DER of the five AF categories using our six proposed models. Note that, because the CNN-RNN-CTC model is not a multi-label model, we did not use it for the AF detection task.

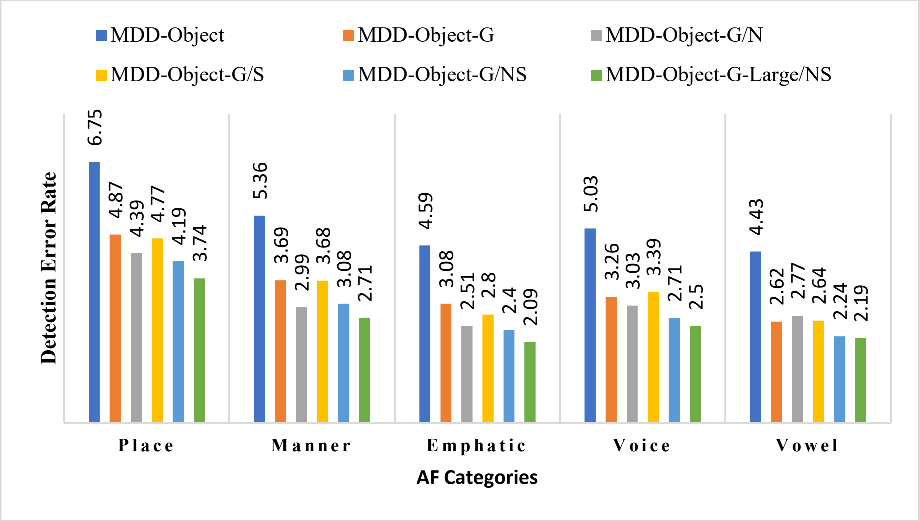
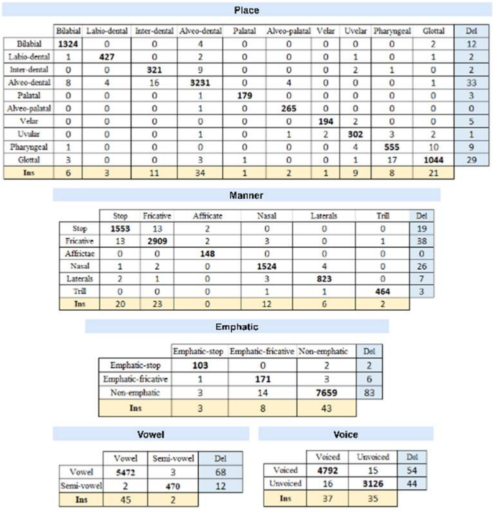


Figure 7. Detection error rate (DER) for each AF category using our proposed models.

We can see from Figure 7 that our best model MDD-Object-G-Large/NS achieved DERs of 3.47, 2.71, 2.09, 2.5, and 2.19 for all five categories, place, manner, emphatic, voice, and vowel, respectively. Similar to the phoneme recognition task, we can clearly notice the usefulness of fine-tuning the hyper-parameters and using synthesized speech to enhance the performance of the proposed system.

Figure 8 presents the confusion matrix of each of the five AF categories using our best model, MDD-Object-G-Large/NS. These confusion matrices show the ability of the proposed system to distinguish between the different classes within the categories with little confusion. In the matrices, we can see that the confusion is between near classes. For example, for the place category, the highest confusion occurs in the glottal vs. pharyngeal and alveo-dental vs. interdental classes, and these two pairs of places are near to each other. Another example is that in the manner category, the highest confusion occurs between stop and fricative classes, which are near to each other, and this is consistent with what was shown in English AF detection in [66]. From these interesting results, we can see that our proposed system provides us with the capability of using the detected AFs to teach non-native Arabic learners the correct pronunciation by providing suitable feedback at the articulatory level.



Place

	Bilabial	Labio-dental	Inter-dental	Alveo-dental	Palatal	Alveo-palatal	Velar	Uvelar	Pharyngeal	Glottal	Del
Bilabial	**1324**	0	0	4	0	0	0	0	0	2	12
Labio-dental	1	**427**	0	2	0	0	0	1	0	1	2
Inter-dental	0	0	**321**	9	0	0	0	2	1	0	2
Alveo-dental	8	4	16	**3231**	0	4	0	0	0	1	33
Palatal	0	0	0	1	**179**	0	0	0	0	0	3
Alveo-palatal	0	0	0	1	0	**265**	0	0	0	0	0
Velar	0	0	0	0	0	0	**194**	2	0	0	5
Uvular	0	0	0	1	0	1	2	**302**	3	2	1
Pharyngeal	1	0	0	0	0	0	0	4	**555**	10	9
Glottal	3	0	0	3	1	0	0	1	17	**1044**	29
Ins	6	3	11	34	1	2	1	9	8	21	

Manner

	Stop	Fricative	Affricate	Nasal	Laterals	Trill	Del
Stop	**1553**	13	2	0	0	0	19
Fricative	13	**2909**	2	3	0	1	38
Affrictae	0	0	**148**	0	0	0	0
Nasal	1	2	0	**1524**	4	0	26
Laterals	2	1	0	3	**823**	0	7
Trill	0	0	0	1	1	**464**	3
Ins	20	23	0	12	6	2	

Emphatic

	Emphatic-stop	Emphatic-fricative	Non-emphatic	Del
Emphatic-stop	**103**	0	2	2
Emphatic-fricative	1	**171**	3	6
Non-emphatic	3	14	**7659**	83
Ins	3	8	43	

Vowel

	Vowel	Semi-vowel	Del
Vowel	**5472**	3	68
Semi-vowel	2	**470**	12
Ins	45	2	

Voice

	Voiced	Unvoiced	Del
Voiced	**4792**	15	54
Unvoiced	16	**3126**	44
Ins	37	35	

Figure 8. Confusion matrix of each of the AF categories using our best model: MDD-object-G-Large/NS.

8. Conclusions

An efficient system for MDD and feedback generation is proposed in this paper by adapting the object detection technique for phoneme recognition and AF detection. For a better adaptation of the selected detector to this new task, we applied GA to select the best hyper-parameters of the model, and the obtained results show the effectiveness of this process. Due to the scarcity of a non-native Arabic speech corpus, we developed an L1 diversified and fully annotated non-native speech corpus, Arabic-CAPT. Then, to solve the problem of a small amount of non-native Arabic speech, we investigated using a synthesized non-native Arabic speech corpus to improve the system performance. The different results for the different systems proved the advantage of using synthesized speech to improve the performance of the proposed MDD models.

The ability of our MDD-Object system to detect phonemes and the corresponding AFs concurrently makes the proposed system more effective in teaching the Arabic language as a second language because it will have the capability to provide additional informative articulatory feedback to learners. We compared the performance of our system to the performance of the state-of-the-art end-to-end technique for the MDD task of non-native Arabic speech and our proposed system had better performance. The enhancement of the end-to-end technique using the developed synthesized corpus was also investigated and showed that the developed synthesized corpus improves the results. Moreover, we investigated fusing our MDD-Object and MDD-E2E systems and obtained excellent results that were better than each of the two systems alone.

In future work, we aim to examine the proposed MDD-Object system for other languages, such as English and Mandarin. We may also investigate the advantage of adding a language model to our proposed system.

Author Contributions: Conceptualization, M.A. (Mohammed Algabri), H.M. and M.A. (Mansour Alsulaiman); methodology, M.A. (Mohammed Algabri) and M.A.B.; validation, M.A. (Mansour Alsulaiman), H.M. and M.A.B.; investigation, M.A. (Mohammed Algabri), M.A. (Mansour Alsulaiman), H.M. and M.A.B.; data curation, M.A. (Mohammed Algabri), M.A. (Mansour Alsulaiman) and M.A.B.; writing—original draft preparation, M.A. (Mohammed Algabri) and M.A. (Mansour Alsulaiman); writing—review and editing, M.A. (Mohammed Algabri) and M.A. (Mansour Alsulaiman); visualization, M.A. (Mohammed Algabri) and M.A.B.; supervision, H.M. and M.A. (Mansour Alsulaiman); project administration, M.A. (Mansour Alsulaiman); funding acquisition, M.A. (Mansour Alsulaiman). All authors have read and agreed to the published version of the manuscript.

Funding: This Project was funded by the National Plan for Science, Technology and Innovation (MAARIFAH), King Abdulaziz City for Science and Technology, Kingdom of Saudi Arabia, Award Number (3-17-09-001-0003).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Daniel, J. Education and the COVID-19 pandemic. *Prospects* **2020**, *49*, 91–96. [CrossRef] [PubMed]
2. Ghazi-Saidi, L.; Criffield, A.; Kracl, C.L.; McKelvey, M.; Obasi, S.N.; Vu, P. Moving from Face-to-Face to Remote Instruction in a Higher Education Institution during a Pandemic: Multiple Case Studies. *Int. J. Technol. Educ. Sci.* **2020**, *4*, 370–383. [CrossRef]
3. Neri, A.; Cucchiarini, C.; Strik, H.; Boves, L. The pedagogy-technology interface in computer assisted pronunciation training. *Comput. Assist. Lang. Learn.* **2002**, *15*, 441–467. [CrossRef]
4. Revell-Rogerson, P.M. Computer-Assisted Pronunciation Training (CAPT): Current Issues and Future Directions. *RELC J.* **2021**, *52*, 189–205. [CrossRef]
5. Cheng, V.C.-W.; Lau, V.K.-T.; Lam, R.W.-K.; Zhan, T.-J.; Chan, P.-K. Improving English Phoneme Pronunciation with Automatic Speech Recognition Using Voice Chatbot. In Proceedings of the International Conference on Technology in Education, Online, 17 December 2020; pp. 88–99.
6. Yan, B.C.; Wu, M.C.; Hung, H.T.; Chen, B. An end-to-end mispronunciation detection system for L2 English speech leveraging novel anti-phone modeling. In Proceedings of the Annual Conference of the International Speech Communication Association, Shanghai, China, 25–29 October 2020; pp. 3032–3036. [CrossRef]
7. Duan, R.; Kawahara, T.; Dantsuji, M.; Nanjo, H. Efficient learning of articulatory models based on multi-label training and label correction for pronunciation learning. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 6239–6243.
8. Engwall, O.; Bälter, O. Pronunciation feedback from real and virtual language teachers. *Comput. Assist. Lang. Learn.* **2007**, *20*, 235–262. [CrossRef]
9. Balas, V.E.; Roy, S.S.; Sharma, D.; Samui, P. *Handbook of Deep Learning Applications*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 136.
10. Pal, S.K.; Pramanik, A.; Maiti, J.; Mitra, P. Deep learning in multi-object detection and tracking: State of the art. *Appl. Intell.* **2021**, *51*, 6400–6429. [CrossRef]
11. Jiao, L.; Zhang, F.; Liu, F.; Yang, S.; Li, L.; Feng, Z.; Qu, R. A Survey of Deep Learning-Based Object Detection. *IEEE Access* **2019**, *7*, 128837–128868. [CrossRef]
12. Elhoseny, M. Multi-object Detection and Tracking (MODT) Machine Learning Model for Real-Time Video Surveillance Systems. *Circuits Syst. Signal Process.* **2020**, *39*, 611–630. [CrossRef]
13. Segal, Y.; Fuchs, T.S.; Keshet, J. Speechyolo: Detection and localization of speech objects. In Proceedings of the Annual Conference of the International Speech Communication Association, Graz, Austria, 15–19 September 2019; pp. 4210–4214.
14. Algabri, M.; Mathkour, H.; Bencherif, M.A.; Alsulaiman, M.; Mekhtiche, M.A. Towards Deep Object Detection Techniques for Phoneme Recognition. *IEEE Access* **2020**, *8*, 54663–54680. [CrossRef]

15. Algabri, M.; Mathkour, H.; Alsulaiman, M.M.; Bencherif, M.A. Deep learning-based detection of articulatory features in arabic and english speech. *Sensors* **2021**, *21*, 1205. [CrossRef]
16. Duan, R.; Kawahara, T.; Dantsuji, M.; Nanjo, H. Cross-Lingual Transfer Learning of Non-Native Acoustic Modeling for Pronunciation Error Detection and Diagnosis. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *28*, 391–401. [CrossRef]
17. Abdou, S.M.; Hamid, S.E.; Rashwan, M.; Samir, A.; Abdel-Hamid, O.; Shahin, M.; Nazih, W. Computer aided pronunciation learning system using speech recognition techniques. In Proceedings of the Ninth International Conference on Spoken Language Processing, Pittsburgh, PA, USA, 17–21 September 2006.
18. Tabbaa, H.M.A.; Soudan, B. Computer-Aided Training for Quranic Recitation. *Procedia Soc. Behav. Sci.* **2015**, *192*, 778–787. [CrossRef]
19. Hindi, A.A.; Alsulaiman, M.; Muhammad, G.; Al-Kahtani, S. Automatic pronunciation error detection of nonnative Arabic Speech. In Proceedings of the 2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA), Doha, Qatar, 10–13 November 2014; pp. 190–197.
20. Alsulaiman, M.; Ali, Z.; Muhammed, G.; Bencherif, M.; Mahmood, A. KSU speech database: Text selection, recording and verification. In Proceedings of the 2013 European Modelling Symposium, Manchester, UK, 20–22 November 2013; pp. 237–242.
21. Alsulaiman, M.; Muhammad, G.; Bencherif, M.A.; Mahmood, A.; Ali, Z. KSU rich Arabic speech database. *Information* **2013**, *16*, 4231–4253.
22. Maqsood, M.; Habib, H.A.; Anwar, S.M.; Ghazanfar, M.A.; Nawaz, T. A Comparative Study of Classifier Based Mispronunciation Detection System for Confusing Arabic Phoneme Pairs. *Nucleus* **2017**, *54*, 114–120.
23. Maqsood, M.; Habib, H.A.; Nawaz, T. An efficientmis pronunciation detection system using discriminative acoustic phonetic features for arabic consonants. *Int. Arab J. Inf. Technol.* **2019**, *16*, 242–250.
24. Nazir, F.; Majeed, M.N.; Ghazanfar, M.A.; Maqsood, M. Mispronunciation detection using deep convolutional neural network features and transfer learning-based model for Arabic phonemes. *IEEE Access* **2019**, *7*, 52589–52608. [CrossRef]
25. Akhtar, S.; Hussain, F.; Raja, F.R.; Ehatisham-ul-haq, M.; Baloch, N.K.; Ishmanov, F.; Zikria, Y.B. Improving mispronunciation detection of Arabic words for non-native learners using deep convolutional neural network features. *Electronics* **2020**, *9*, 963. [CrossRef]
26. Ziafat, N.; Ahmad, H.F.; Fatima, I.; Zia, M.; Alhumam, A.; Rajpoot, K. Correct Pronunciation Detection of the Arabic Alphabet Using Deep Learning. *Appl. Sci.* **2021**, *11*, 2508. [CrossRef]
27. Boyer, F.; Rouas, J.-L. End-to-End Speech Recognition: A review for the French Language. *arXiv* **2019**, arXiv:1910.08502.
28. Watanabe, S.; Boyer, F.; Chang, X.; Guo, P.; Hayashi, T.; Higuchi, Y.; Hori, T.; Huang, W.-C.; Inaguma, H.; Kamo, N. The 2020 ESPnet update: New features, broadened applications, performance improvements, and future plans. In Proceedings of the 2021 IEEE Data Science and Learning Workshop (DSLW), Toronto, ON, Canada, 5–6 June 2021. [CrossRef]
29. Feng, Y.; Fu, G.; Chen, Q.; Chen, K. SED-MDD: Towards Sentence Dependent End-To-End Mispronunciation Detection and Diagnosis. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 3492–3496.
30. Leung, W.-K.; Liu, X.; Meng, H. CNN-RNN-CTC based end-to-end mispronunciation detection and diagnosis. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 8132–8136.
31. Zhang, Z.; Wang, Y.; Yang, J. Text-conditioned Transformer for automatic pronunciation error detection. *Speech Commun.* **2021**, *130*, 55–63. [CrossRef]
32. Lo, T.H.; Weng, S.Y.; Chang, H.J.; Chen, B. An effective end-to-end modeling approach for mispronunciation detection. In Proceedings of the Annual Conference of the International Speech Communication Association, Shanghai, China, 25–29 October 2020; pp. 3027–3031. [CrossRef]
33. Zhang, L.; Zhao, Z.; Ma, C.; Shan, L.; Sun, H.; Jiang, L.; Deng, S.; Gao, C. End-to-End Automatic Pronunciation Error Detection Based on Improved Hybrid CTC/Attention Architecture. *Sensors* **2020**, *20*, 1809. [CrossRef] [PubMed]
34. Zhao, G.; Sonsaat, S.; Silpachai, A.; Lucic, I.; Chukharev-Hudilainen, E.; Levis, J.; Gutierrez-Osuna, R. L2-Arctic: A non-native English speech corpus. In Proceedings of the Annuale Conference International Speech Communication Association Interspeech, Hyderabad, India, 2–6 September 2018; pp. 2783–2787. [CrossRef]
35. Li, K.; Qian, X.; Meng, H. Mispronunciation detection and diagnosis in l2 english speech using multidistribution deep neural networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2016**, *25*, 193–207. [CrossRef]
36. Chen, N.F.; Tong, R.; Wee, D.; Lee, P.; Ma, B.; Li, H. iCALL corpus: Mandarin Chinese spoken by non-native speakers of European descent. In Proceedings of the Sixteenth Annual Conference of the International Speech Communication Association, Dresden, Germany, 6–10 September 2015.
37. Benarousse, L.; Grieco, J.; Geoffrois, E.; Series, R.; Steeneken, H.; Stumpf, H.; Swail, C.; Thiel, D. The NATO native and non-native (N4) speech corpus. In Proceedings of the Workshop on Multilingual Speech and Language Processing, Aalborg, Denmark, 17 April 2001.
38. Pettarin, A. Aeneas is a Python/C Library and a Set of Tools to Automagically Synchronize Audio and Text (Aka Forced Alignment). GitHub In Repository; GitHub. 2017. Available online: https://github.com/readbeyond/aeneas (accessed on 10 June 2022).

39. McAuliffe, M.; Socolof, M.; Mihuc, S.; Wagner, M.; Sonderegger, M. Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi. *Interspeech* **2017**, *2017*, 498–502.
40. Halabi, N. Modern Standard Arabic Phonetics for Speech Synthesis. Ph.D. Thesis, University of Southampton, Southampton, UK, 2016.
41. Halabi, N. Arabic Phonetiser, GitHub In Repository; GitHub. 2016. Available online: https://github.com/nawarhalabi/Arabic-Phonetiser (accessed on 10 June 2022).
42. Salamon, J.; Bello, J.P. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Process. Lett.* **2017**, *24*, 279–283. [CrossRef]
43. Raileanu, R.; Goldstein, M.; Yarats, D.; Kostrikov, I.; Fergus, R. Automatic Data Augmentation for Generalization in Deep Reinforcement Learning. *arXiv* **2020**, arXiv:2006.12862.
44. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
45. Rosenberg, A.; Zhang, Y.; Ramabhadran, B.; Jia, Y.; Moreno, P.; Wu, Y.; Wu, Z. Speech recognition with augmented synthesized speech. In Proceedings of the 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Singapore, 14–18 December 2019; pp. 996–1002.
46. Li, J.; Gadde, R.; Ginsburg, B.; Lavrukhin, V. Training Neural Speech Recognition Systems with Synthetic Speech Augmentation. *arXiv* **2018**, arXiv:1811.00707.
47. Korzekwa, D.; Barra-Chicote, R.; Zaporowski, S.; Beringer, G.; Lorenzo-Trueba, J.; Serafinowicz, A.; Droppo, J.; Drugman, T.; Kostek, B. Detection of lexical stress errors in non-native (L2) english with data augmentation and attention. In Proceedings of the Annual Conference of the International Speech Communication Association, Brno, Czech Republic, 15–19 September 2021; Volume 2, pp. 1446–1450. [CrossRef]
48. Ren, Y.; Hu, C.; Tan, X.; Qin, T.; Zhao, S.; Zhao, Z.; Liu, T.Y. FastSpeech 2: Fast and High-Quality End-to-End Text to Speech. In Proceedings of the International Conference on Learning Representations, Online, 26 April–1 May 2020.
49. Lin, Y.; Wang, L.; Dang, J.; Li, S.; Ding, C. End-to-End articulatory modeling for dysarthric articulatory attribute detection. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 7349–7353.
50. Qamhan, M.A.; Alotaibi, Y.A.; Seddiq, Y.M.; Meftah, A.H.; Selouani, S.A. Sequence-to-Sequence Acoustic-to-Phonetic Conversion using Spectrograms and Deep Learning. *IEEE Access* **2021**, *9*, 80209–80220. [CrossRef]
51. Seddiq, Y.; Alotaibi, Y.A.; Selouani, S.-A.; Meftah, A.H. Distinctive Phonetic Features Modeling and Extraction Using Deep Neural Networks. *IEEE Access* **2019**, *7*, 81382–81396. [CrossRef]
52. Abdultwab, K.S. Sound substitution in consonants by learners of Arabic as a second language:Applied study on students of Arabic Linguistics Institute. In Proceedings of the Third International Conference for the Arabic Linguistics Institute in King Saud University, Riyadh, Saudi Arabia, 6–7 March 2019; pp. 157–202. (In Arabic).
53. Alghamdi, M. *Arabic Phonetics and Phonology*; Al-Toubah Bookshop: Al Riyadh, Saudi Arabia, 2015. (In Arabic)
54. Zenkel, T.; Sanabria, R.; Metze, F.; Niehues, J.; Sperber, M.; Stüker, S.; Waibel, A. Comparison of decoding strategies for CTC acoustic models. In Proceedings of the Annual Conference of the International Speech Communication Association, Stockholm, Sweden, 20–24 August 2017; pp. 513–517. [CrossRef]
55. Young, S.; Evermann, G.; Gales, M.J.F.; Hain, T. *The HTK Book*; Cambridge University Engineering Department: Cambridge, UK, 2002; Volume 3, p. 12.
56. Qian, X.; Soong, F.K.; Meng, H. Discriminative acoustic model for improving mispronunciation detection and diagnosis in computer-aided pronunciation training (CAPT). In Proceedings of the Eleventh Annual Conference of the International Speech Communication Association, Chiba, Japan, 26–30 September 2010.
57. Wang, Y.-B.; Lee, L. Supervised detection and unsupervised discovery of pronunciation error patterns for computer-assisted language learning. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2015**, *23*, 564–579. [CrossRef]
58. Amodei, D.; Ananthanarayanan, S.; Anubhai, R.; Bai, J.; Battenberg, E.; Case, C.; Casper, J.; Catanzaro, B.; Cheng, Q.; Chen, G.; et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19 June 2016; pp. 173–182.
59. Li, J.; Wu, Y.; Gaur, Y.; Wang, C.; Zhao, R.; Liu, S. On the comparison of popular end-to-end models for large scale speech recognition. *arXiv* **2020**, arXiv:2005.14327. [CrossRef]
60. Zhang, Z.; Wang, Y.; Yang, J. Mispronunciation Detection and Correction via Discrete Acoustic Units. *arXiv* **2021**, arXiv:2108.05517.
61. Jiang, S.W.F.; Yan, B.C.; Lo, T.H.; Chao, F.A.; Chen, B. Towards Robust Mispronunciation Detection and Diagnosis for L2 English Learners with Accent-Modulating Methods. In Proceedings of the 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Cartagena, Colombia, 13–17 December 2021; pp. 1065–1070. [CrossRef]
62. Wu, M.; Li, K.; Leung, W.K.; Meng, H. Transformer based end-to-end mispronunciation detection and diagnosis. In Proceedings of the Annual Conference International Speech Communication Association Interspeech, Brno, Czech Republic, 30 August–3 September 2021; Volume 2, pp. 1471–1475. [CrossRef]
63. Fu, K.; Lin, J.; Ke, D.; Xie, Y.; Zhang, J.; Lin, B. A Full Text-Dependent End to End Mispronunciation Detection and Diagnosis with Easy Data Augmentation Techniques. *arXiv* **2021**, arXiv:2104.08428.
64. Ganaie, M.A.; Hu, M.; Malik, A.K.; Tanveer, M.; Suganthan, P.N. Ensemble deep learning: A review. *arXiv* **2021**, arXiv:2104.02395. [CrossRef]

65. Eskenazi, M. An overview of spoken language technology for education. *Speech Commun.* **2009**, *51*, 832–844. [CrossRef]
66. King, S.; Taylor, P. Detection of phonological features in continuous speech using neural networks. *Comput. Speech Lang.* **2000**, *14*, 333–353. [CrossRef]