Design, Implementation and Evaluation of a Scanned Filipino Text-to-Speech Device as a Reading Aid for the Blind and Visually Impaired

Christian Paulo L. De Leon
FEU – Institute of Technology
Manila, Philippines
+63 905 175 1817
paulo.deleon12@gmail.com

Allan Adrian B. Gatchalian
FEU – Institute of Technology
Manila, Philippines
+63 905 669 7092
allangatchi@gmail.com

Juan Miguel D. Ninobla
FEU – Institute of Technology
Manila, Philippines
+63 917 745 4516
jmd.ninobla@gmail.com

Nigel Roi D. Concepcion
FEU – Institute of Technology
Manila, Philippines
+63 906 391 4665
nigelroi@gmail.com

David Joseph O. Catemprate
FEU – Institute of Technology
Manila, Philippines
+63 905 966 8229
djcatemprate@gmail.com

Ronald M. Pascual
De La Salle University
Manila, Philippines
+63 (02) 524 4611
ronald.pascual@dlsu.ph

ABSTRACT

In this study, the authors provide new insights to the design, implementation and evaluation of a TTS system where the inputs are text images in Filipino. Many text-to-speech (TTS) systems have been studied and implemented in many languages. [8] Portable devices for Filipino TTS however have not been explored yet. With the aid of portable scanning device and Optical Character Recognition (OCR), our TTS system can generate synthetic Filipino speech from texts found on printed reading materials in Filipino.

The authors present the development of a scanned Filipino TTS converter device as a reading aid for the blind and visually impaired. The authors implemented the design using a raspberry pi as its main control unit, and interfacing it with a text scanner, a speaker system, and a power supply module. The system uses 3000-word database adapted from the most commonly used words in the Filipino speech corpus [2] from UP – Sentro ng Wikang Filipino. Employing a certain speech synthesis toolkit [8], the synthesized Filipino words were generated by creating phonetic records that emulate the Filipino speech. Listening tests were conducted to measure the quality of the synthesized Filipino speech. The aforementioned tests involved 60 respondents rating the quality of the output speech. Results showed that overall, on a scale of 1 to 5 with 5 being the highest, the respondents rated the quality of the output speech at 4.1333.

CCS Concepts

• Artificial intelligence → Natural language processing
• Human-centered computing → Accessibility → Accessibility technologies
• CCS → Hardware → Communication hardware, interfaces and storage → Signal processing systems → Digital signal processing

Keywords

Filipino speech; Text-to-speech; Synthesizers; Speech Processing

1. INTRODUCTION

Reading is an essential tool for learning and communication. Today, huge amounts of information are still found on printed materials such as books, magazines, newspapers, etc.

There are instruments invented and studies conducted that are intended for the blind and visually impaired who wish to learn from books e.g., braille. Braille printers however are expensive and cost around $2000 to $6000. [1] Therefore, only a limited number of people who can use this technology. Studies like text-to-speech (TTS) systems are done to help aid the problem.

Text-to-speech system is one that generates human-like speech from an input text. Speech synthesizers used in TTS have been developed over the years as memory resources become available and cheaper; As a result, large enhancements in the quality and the intelligibility of the synthesized speech have been achieved. [15]

Research works on text-to-speech conversion are expected to produce human-like speech, although it may seem to sound robotic and artificial. Concatenation of speech units should ideally be done in a way such that discontinuities at concatenation points are almost unnoticeable. A paper closely related with this study is the "Development of a Filipino TTS System Using Concatenative Speech Synthesis". [4] The aforementioned study synthesizes Filipino speech using concatenative speech synthesis. In this study however, the authors used formant speech synthesis which is widely used in other related studies.

A highlight of this study is the development of a device for the use of blind or visually impaired people who are looking for a convenient way to read printed Filipino materials.

In order to use TTS on a given printed material containing the text, Optical Character Recognition (OCR) is used. Optical Character Recognition is the process of extracting text from an image. [3] An OCR system first analyzes the layout of a document in order to extract the text blocks which are then segmented into lines. [14] Implementation of the foregoing processes on projects and experiments would typically require a small, affordable, and flexible computer such as the Raspberry Pi. [13] The authors used Tesseract OCR engine for the device. For the accuracy, the test was conducted by feeding the system documents containing the 3000 words database. The system will evaluate the words that are not correctly read as an error. From the data gathered, the authors calculated 0.3% error or 99.667% accurate.

Philippine Computing Journal Dedicated Issue on Natural Language Processing, pages 1–8
Vol. XIII No. 1 August 2018

One of the highlights of this work is the speech synthesis component. Speech synthesis is the process for the generation of speech waveforms with the use of a machine. [9] Phonemes are linguistically defined small units of speech. For example, /upa/ and /apa/ differ only in their initial vocal tract configurations, but are two different words, and thus comprise different sets of phonemes. [11] The authors used eSpeak, an open source software for speech synthesis.

With the use of the aforementioned technologies, Filipino texts from printed documents may be converted digitally and the speech sounds representing the words may be played back to the user and make reading easier for the visually disabled.

This project aims to help blind and visually impaired people read printed texts in the Filipino language.

This study has four specific objectives: First is to develop a software that converts scanned Filipino texts to speech; Second is to develop a hardware that uses Raspberry Pi as the main control unit for the text-to-speech conversion of the device; Third is to create a database of Filipino words to be converted to its equivalent speech for the speech synthesizer; And fourth is to design and conduct experiments that would evaluate the performance of the device in terms of quality, and speed of the system.

The speech quality of the system was evaluated using MOS or Mean Opinion Score, wherein the listeners that rated the system are visually impaired. [16] The usability of the system was evaluated through a survey conducted with blind and visually impaired people, while the speed of the system was evaluated by the authors.

The database of the system contains three thousand Filipino words. Whenever the system encounters words that are not included in the database, the system spells out the words for the user. The system is only applicable for printed single page texts and the system can be used for common non-cursive font types. The system cannot fully emulate the proper intonation of words as they are used in the sentence. For words that have multiple meanings depending on the syllable stress, the system only picks up the most commonly used.

Figure 1 shows the actual image of the prototype. It measures at 275mm*75mm*75mm. The measurements are based on the scanner size. It is built to accommodate the size of both the Raspberry Pi, the lithium Ion Batteries, and the Handheld Scanner. The paper feed opening width is designed such that the blind can easily insert bond papers.

2. SYSTEM DESIGN

2.1 System Diagram

Figure 2 shows the Block Diagram of the system developed in this

Figure 1. The Prototype

work. The input consists mainly of the printed image of the text to be decoded by the scanner. The system may also be operated using voice commands through the use of a Voice Recognition Module which in turn is interfaced with the Raspberry Pi through the serial port.

Voice commands are monitored by the voice recognition module. The voice recognition module is connected to the Raspberry Pi through the UART (Universal Asynchronous Receive Transmit) Port located at Pin 8 (TXD) and Pin 10 (RXD) of the Raspberry Pi. The Voice Recognition Module interprets the commands and sends the equivalent serial command to the Raspberry Pi which

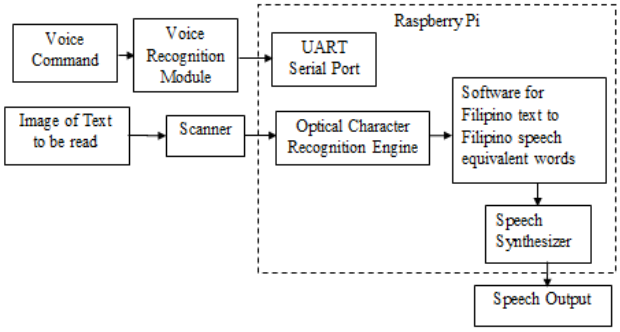

Figure 2. System Diagram

then interprets the command and executes a pre-programmed interrupt subroutine according to the voice command.

The image of the text to be scanned would be obtained by the system through the scanner. The scanner is connected to one of the four USB 2.0 ports of the Raspberry Pi. The system would then pass the image to an Optical Character Recognition Engine to obtain the Filipino text from the image. The text would then be processed by the software that converts the Filipino text to its phonetically equivalent synthesized word to emulate Filipino Speech. The words would then be sent to the speech synthesizer and would be spoken as Filipino Speech.

2.2 Process Flowchart

Figure 3 shows the System Flowchart. The system starts by initializing the microcontroller. It then waits for the user input to say the "Start" command. Once the "Start" command has been issued, the system starts by sending a "Scan" command to the scanner and waiting for the scanner to send the image to the Raspberry Pi. Once the image has been received, the image would then be sent to the OCR Engine to extract the text from the image. The text will be then converted to its equivalent words that will be able to emulate the Filipino speech through the use of the "Filipino text to Filipino Speech Equivalent Words" Function. The processed words would then be used for the speech synthesizer which converts the words to speech. The system stops then sends the timer data to the console where it can be viewed by the developer.

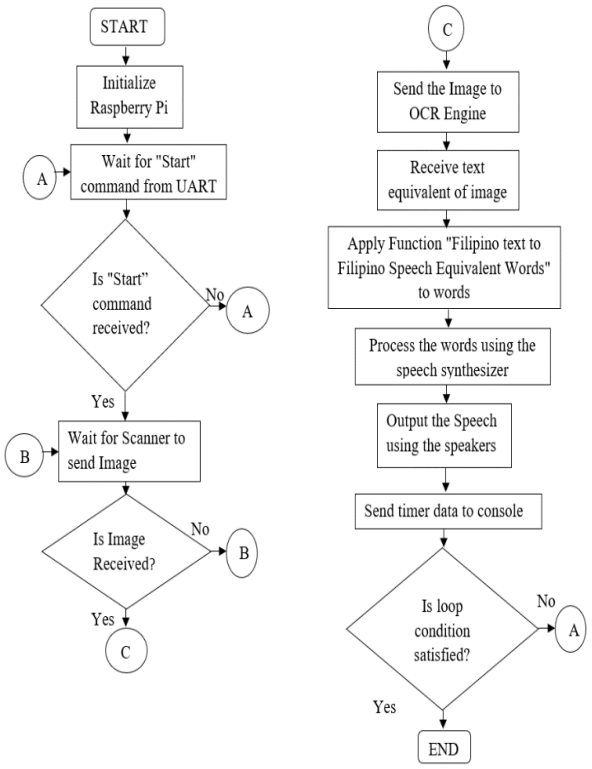

Figure 3. Process Flowchart

2.3 Design Considerations

The system is designed to be used by people who have difficulty in reading Filipino text, particularly for the visually impaired. It allows these people to read the Filipino texts through the use of the system.

The software developed was coded through Python language which was ran using the Raspberry Pi. It was coded to utilize the Raspberry Pi's GPIO (General Purpose Input Output) pins where data will be sent to and from the system. The software was set to run on boot time, where after loading the Raspberry pi will immediately run the software.

The software is responsible for interpreting the data coming from the scanner, which is connected to the USB Port, and the Voice Recognition Module, which is connected at the UART port. The buttons and other passive peripherals are connected to the GPIO where data handlers execute functions based on the inputs received.

The Filipino text to Filipino Speech Equivalent Words Function is based on phonetics and how words of different languages but with similar phonemes can be used to emulate each other. It converts the Filipino text to its equivalent collection of phonetically similar words that were able to emulate Filipino speech, which would then be sent to the eSpeak speech synthesizer. The function uses words or an arrangement of letters, which phonetically speaking is equivalent to the Filipino word if synthesized by the synthesizer to sound similar to the Filipino word when spoken. The synthesizer is composed of parameters that can be varied to emulate the correct pronunciation of the word in Filipino. These parameters are the rate, intonation, and stress. Rate determines the synthesizer's speed of utterance per word, intonation defines the variation in tone or pitch throughout an utterance of the word. Stress, depending on its position, differentiates the meaning of two words with same spelling.

This software is limited to the number of words defined in its database. Figure 4 shows the speech parameters and syllable stresses. It is the visualization of what the software does. It converts the Filipino words to its phonemic counterpart that when spoken by the speech synthesizer, would be identical to Filipino speech. Like in the example, a word may have a different meaning depending on the syllable stress.

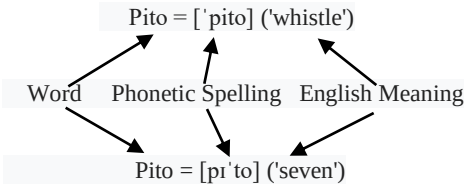

Figure 4. Speech Parameters and Syllable Stresses

Figure 5 shows the Process Flowchart for the Filipino text to Filipino Speech Equivalent Words Function. The function starts by being passed on a variable "Filipino Text" which is obtained from the OCR Engine. The function then splits the string variable to individual words using " " as a delimiter. The function then enters a loop which cycles through all the words contained within. In each loop cycle, the word is searched for in the dictionary. If the word is found in the dictionary, the word is then substituted to its equivalent collection of words that emulate Filipino speech; otherwise, it separates the letters of the word with a " " which will be read by letters. The converted word is then concatenated with the return string variable "Text Out" and the loop is then continued until it reaches the last word in the "Filipino Text" variable. Once the last word has been converted, the function then returns the string variable "Text Out".

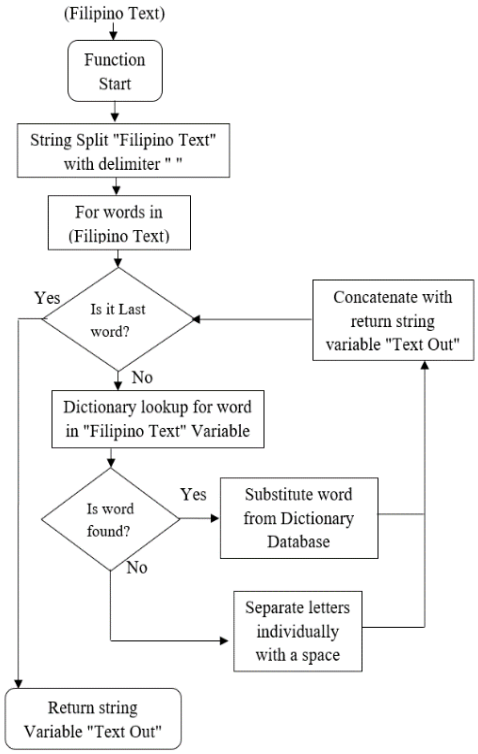

Figure 5. Process Flowchart

Figure 6 shows how the dictionary database was created. The system uses a Dictionary Database to search for known words which have been previously converted to its equivalent collection of words that emulate Filipino speech. The dictionary is created by reviewing the Filipino word and, through hearing, creates a word or a collection of words that, when sent to the speech synthesizer through whatever language setting, generates phonemes that are similar to those appearing in Filipino speech.

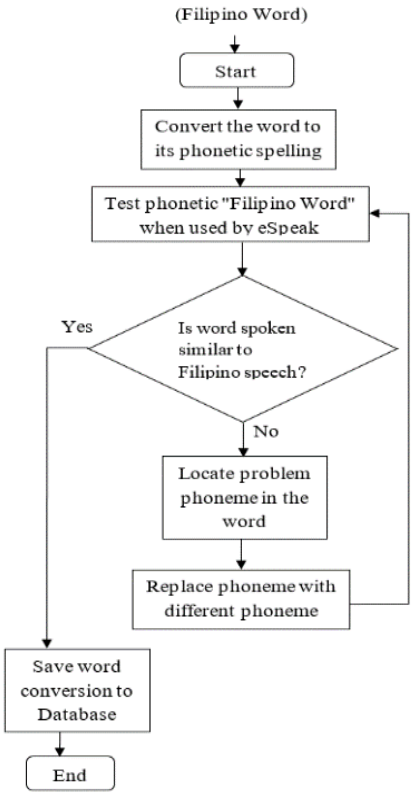

Figure 6. Flowchart of Software for Filipino-Speech-Equivalent-Words Function

The process starts by first converting the word to its equivalent phonetic spelling. It is then sent to the eSpeak program where the word will be spoken phonetically. If the word spoken is similar to Filipino speech, the converted word is then saved to the database. If the spoken word is not similar to Filipino speech, the problematic phoneme is located. Then, it will be replaced by a different phoneme and is once again sent to the eSpeak for testing. The testing was conducted by listening to the word spoken. Each word will have an equivalent phonetic spelling that, when sent to the eSpeak, will be similar to Filipino speech.

2.4 Speech Synthesis Tool

In this study, the authors employed eSpeak, a speech synthesis tool for English. It is able to convert text to phonemes with pitch and length information. It uses the "formant synthesis" method which allows the system to have many languages in a small size. The speech is clear enough to be understood and is fast enough for a concise spoken sentence. It is not based on human speech recordings which mean it is not as natural or smooth as compared to other speech synthesizers. eSpeak can be called from the command line, used as a shared library, executed as a stand-alone program, or ported to other platforms. It includes different alterable voices and can be exported to a *.wav file. Languages included are the following: Afrikaans, Aragonese, Armenian, Bulgarian, Croatian, Czech, Danish, Dutch, English, French, etc. [6] The absence of Filipino synthesizer makes it complicated to create the database as other languages sound differently.

2.5 Formants

The acoustic resonance of the human vocal tract is called a formant. However, in signal processing point of view, formant is a range of frequency in the sound spectrum, where there is an absolute or relative maximum occurred [10]. Thus, formant can be defined as either a resonance of vocal tract or the spectrum maximum that the resonance produces and it can be measured from amplitude peak in the sound frequency spectrum. [6]

Formants can be seen very clearly in a wideband spectrogram, where they are displayed as dark bands. The darker the formant in a spectrogram (more energy), the stronger it is (more audible). Figure 7 shows the Formants in a spectrogram. The arrows indicate the six instances of the lowest formant. The next formant offers at between 1kHz and 2kHz. Formants are present in both vowels and consonants as the vocal tract filters a source sound. Formants occur at frequencies corresponding to the resonances of the vocal tract. [18]

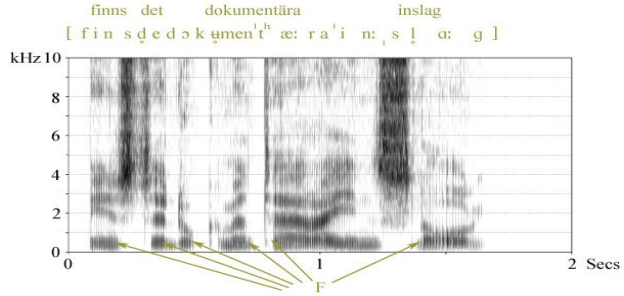

Figure 7. Formants Visualization [18]

Figure 8 shows an example spectogram of the word "kasalukuyan". The red dots illustrate the formants. It is very noticable that the vowels have a darker formant compared to the consonants as vowels are more audible compared to consonants.

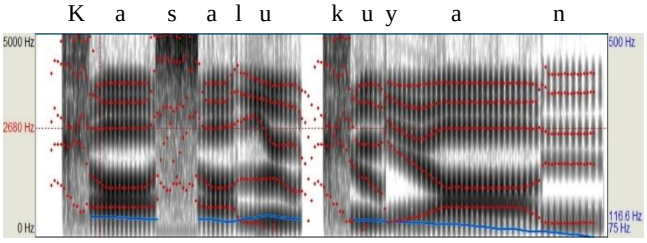

Figure 8. "Kasalukuyan" Formants Visualization

2.6 Design Constraints

The first social constraint the authors met is that the study is limited to Filipino language. The system is designed wherein the output emulates the pronunciation of Filipino speech. The 3000-word database is based from the most frequently used words of the Filipino speech corpus of UP - Sentro ng Wikang Filipino. [2] The authors only used 3000 words from the corpus due to time constraints and manageability aspect of the project in creating the database. The size of the database can however be updated and expanded easily in future works.

The second social constraint is that the study is for blind and visually impaired. The authors have designed a device, while taking into consideration of the demographics' disability. The authors limited the buttons to be used to, on and off button, scan

button, play button and stop button. The authors also implemented features of automatic reading and voice command to be able to make things easier for the users of the device.

3. HARDWARE IMPLEMENTATION

3.1 Raspberry Pi

Raspberry pi 3B is faster than the previous models because of its quad core processor running at 1.2GHz and 1gb of DDR2 RAM which will be beneficial for the system speed.

Figure 9 shows the Raspberry Pi 3B, which controls the system. It is responsible for automating the reading, and speaking out the words for the user. It is powered by a 5v, 2A power supply.

Figure 9. Raspberry Pi

3.2 USB Scanner

Figure 10 shows the USB Scanner, which is used to scan the document that the user is required to be read. It is responsible for converting the printed text to an image that the system requires to read the words printed on it. It is connected to the Raspberry Pi via one of the USB ports. The USB port also powers the scanner. The scanner has a maximum resolution of 600x600 dpi and can support up to 8.5”x 32” size of paper.

Figure 10. USB Scanner

3.3 Voice Command Module

Elechouse Voice recognition Module V3 is a compact and easy to control speech recognition board. It supports up to 80 voice commands in all. It has a digital interface of 5V TTL level for UART and GPIO. It also has 99% accuracy under ideal environment. [17]

Figure 11 shows the Voice Command Module used by the prototype to interpret voice commands by the user. It is connected to the Raspberry Pi through the Serial port. The voice command module was trained to respond to specific commands such as "Start", "Play", and "Stop" which are converted to serial commands that the Raspberry Pi can interpret and then execute the corresponding subroutine associated with the command.

Figure 11. Voice Command Module [12]

3.4 Power Supply Module

Figure 12 shows the Lithium-Ion Battery used on the system. It is a 3.7v, 5000mAh battery capable of currents up to 20A. The system will use 2 Lithium-Ion batteries in a 2S1P (2 series, 1 parallel) configuration to produce a nominal voltage of 7.2vDC which the linear regulator will convert to 5vDC for the system.

Figure 13 shows the Lithium-Ion Battery Charge Controller for the system. It is responsible for charging the batteries in series using a voltage level of 5.1vDC. The device has indication lights, which informs the user that the battery is at full capacity.

Figure 13. Lithium-Ion Charger

4. EVALUATION AND RESULTS

4.1 Testing Procedures

Figure 14 shows the actual testing conducted for the evaluation of the prototype.

Figure 14. Actual Testing

The authors conducted tests that evaluate the system performance. The tests are designed to obtain a subjective and objective analysis of the Synthesized Filipino Speech and the test durations respectively. The test time stamps are obtained by viewing the developer console which contains the running time and duration of the key processes that the system undergoes. Each respondent from Resources for the Blind Inc. agreed that the authors take pictures and videos of the tests for documentation purposes. The testing procedure is as follows:

1. Turn “on” the system
2. Wait for the system to be ready for input (indicated by the system speaking “Handa na bumasa”)
3. Place a document to be read on the scanner feed
4. Scan the selected document for testing by pressing the “Scan button”
5. Record the start time (seen on the developer console) once the system responds with “Babasahin ang dokumento”
6. Wait for the system to speak
7. Evaluate the speech produced by the system
8. Record the process durations (as seen from the developer console) and results

The above stated testing procedure is designed to obtain real-world testing data from the system. The duration of each step of the process is recorded to determine its processing speed. The speech produced by the system was evaluated by using MOS or Mean Opinion Score wherein the listeners that rated the system are visually impaired. To prove that they have understood the document, they must repeat the sentence spoken by the system.

4.2 Statistical Treatment to be Used

The system reliability is tested in order to assess the output of the system. The number of trials that the authors conducted was based from the computation shown below.

Equation 1 shows the formula to be used to obtain the number of tests required for a 95% reliability. Given the Confidence Level, Allowable Failures, and the Required Reliability, the number of tests can be calculated. The number of tests 'n' can be obtained by supplying the other variables, such as Confidence Level 'CL', allowable failures 'r', and the Required Reliability 'R' and solving for 'n'. [7]

$$1-CL=\sum_{i=0}^{r}\binom{n}{i}(1-R)^{i}\,R^{n-i}$$

Equation 1. Bayesian Reliability Demonstration Test

The authors used 95% as a value for Reliability 'R' and Confidence Level "CL". Setting the allowable failures 'r' to 0, the formula is then simplified to obtain the number of tests requires 'n' with an allowable error as 0 as seen in Equation 2. Using the formula, the number of tests to be taken needs to be at least 58.4. The authors conducted 60 tests in order to assure at least 95% reliability in testing the device.

$$n=\frac{\log(1-CL)}{\log(R)}$$

Equation 2. Calculated Number of Trials

4.3 System Evaluation

The Text-to-speech software was evaluated using MOS or Mean Opinion Score; this method is based on ITU-T Rec. P.800. The MOS is generated by averaging the results of a set of standard subjective tests where a number of listeners rate the heard audio quality of test sentences. According to ITU-T Rec. P.800, there are various five-point category judgment scales which can be used for different purposes.

For the quality of the system the listening-quality scale was used as seen on Table 1:

Table 1. Listening-Quality Scale

Quality of Speech	Score
Excellent	5
Good	4
Fair	3
Poor	2
Bad	1

Equation 3 shows the formula for obtaining the Mean Opinion Score. The MOS was computed using the arithmetic mean of all the individual scores. 'X' are the individual ratings for a given stimulus by 'k' subjects.

$$MOS=\frac{\sum_{n=1}^{k}x_n}{k}$$

Equation 3. Formula for Obtaining MOS

The processing speed of the system was measured through the process duration commencing from the moment the printed text is scanned until the Filipino speech is spoken by the system. It is an important factor in the performance of the system since the comparison of the natural speaking voice and the synthesizer speed can determine if the synthesizer can be a viable alternative compared to other pre-recorded materials e.g., cassette tapes, CD, DVD, or other digitally recorded audio. The Filipino language is spoken as it is spelled. For example, two successive vowels, such as the word ‘oo’ (English meaning: ‘yes’), would be pronounced as 2 syllables, ‘o-o’. [5] Syllable-timed languages are observed to have near-equal syllable lengths, regardless of the number of stresses in a morphological or syntactic construction under this classification, linguists have traditionally classified Filipino speech as syllable timed. Because of this, the sample phrases or sentences to be used for the computation of the speed of the system consists of words with at least the same number of syllables.

Table 2 was used for the evaluation of the system speed.

Table 2. Speed of the System

Sample phrase or sentence	Total duration (sec)
1. Tanaw sa bintana ng aking silid ang karagatan	18.09
2. Tinatawag ka kanina ni nanay sa kusina	17.26
3. Nagulat din ako nang malaman	16.68
…	
60. Totoong nangyari ang lahat ng iyon kagabi	19.46
Average Duration	17.2608

Equation 4 shows the formula for the speed of the system. The authors used a software to measure the duration of the system from the start of scanning until the utterance of the word by the synthesizer. The average speed of the system is computed by using the arithmetic mean of all the samples. 'X' are the individual ratings for a given stimulus by 'k' subjects.

$$System\ speed=\frac{\sum_{n=1}^{k}x_n}{k}$$

Equation 4. Formula for System Speed

4.4 Listening Quality

Figure 15 represents the summarized data gathered from the test for the mean opinion score of listening-quality of speech. The testing of the device’s speech quality was conducted by having the blind respondents rate one sentence to test the output speech quality of the system.

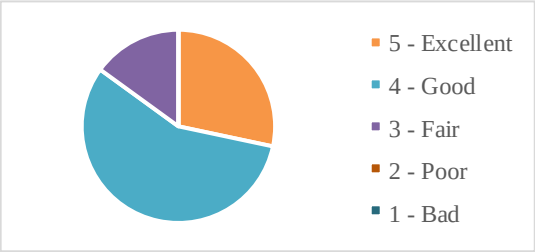

Figure 15. Listening Quality Results

The results may vary from each other because the respondents have different opinions in the listening test. There were times where the respondents had difficulties in understanding the speech output wherein the user replays the file to understand the sentence, but some of the respondents understood the output in their first try of the device. Overall, the authors gathered a value of 4.133333 in the test for the mean opinion score of the quality of speech. This value indicates a good quality of speech produced by the system.

4.5 System Processing Speed

Figure 16 contains the time measurements for the calculation of the average system processing speed. The system was given a sample phrase or sentence. The processing times were recorded by the system from the times when the paper was fed until the system finishes analyzing the data. The duration times varied depending on the length of the scanned sentences. The calculated average duration was 17.2608 seconds per phrase or sentence. The calculated syllables per second was 1.926 or approximately 2 syllables per second.

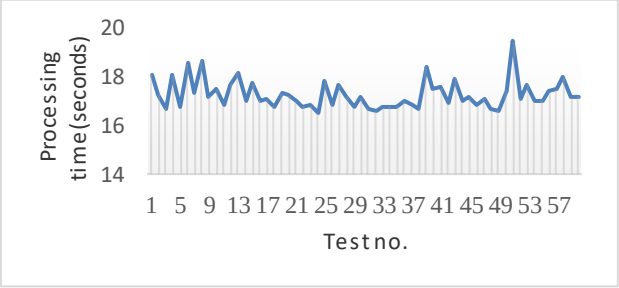

Figure 16. System Processing Speed Measurements

5. CONCLUSION AND RECOMMENDATIONS

5.1 Conclusions

The authors were able to design and evaluate a scanned text-to-Filipino Speech converter device as a reading aid for the blind and visually impaired.

The Raspberry Pi was successfully programmed using Python language, after scanning the image of the Filipino text, the image was converted to text information through the OCR Engine. Then the software converted the Filipino text information to its equivalent emulated Filipino speech using the e-speak speech synthesizer, which called out the equivalent speech output. The hardware was successfully designed and executed to be able to hold the scanner and the programmed Raspberry Pi, which converts the scanned Filipino text to audio output.

The authors were able to create a database from the speech corpus of a research from UP Diliman. The database contains the top 3000 Filipino words that are commonly used. These words have been converted to its phonetic spelling using a dictionary lookup and through the e-speak speech synthesizer embedded in the Raspberry Pi.

The authors used Mean Opinion Score as a measuring instrument, which is based on ITU-T Rec.P.800. It used a five-point category judgement scale which was used to measure quality and ease of use. For the quality of speech, the calculated mean opinion score is 4.166667 out of 5. This shows that the system produced good speech quality. The authors used a software to measure the duration of the system from scanning until the utterance of the synthesized word. The average speed of the system using the arithmetic mean of all the samples is 17.2608 seconds per phrase or sentence. The average syllables per seconds is 1.926 or approximately 2 syllables per second.

The respondents gave a positive feedback regarding the speech quality and system speed. One statement from the respondents said “I like the portability of the device and it is easy to use. The speech is a bit unnatural but very understandable. And the processing speed is

5.2 Recommendations

Feedbacks collected from the respondents after conducting the system tests gave the authors ideas on how to improve the system in the future.

Future studies could use a better speech synthesizer. There may be a great development to the quality of speech in the near future since there are vast studies in the literature for the improvement of speech synthesizers.

Better emulation and more natural intonation and prosodic patterns of words as used in a particular context are also future topics of interest. It may be possible to automatically recognize words with multiple meanings and to automatically adjust the output speech intonation and stresses to match the actual meaning of the word used in a sentence.

Development of the Filipino word database is very essential for the benefit of future studies relating to Filipino text-to-speech.

6. REFERENCES

[1] Braggins, B., Brown, M., Cleary, P., and Witkowski, J. Low cost braille embosser, Retrieved November, 2017, from http://www.mie.neu.edu

[2] Cajote, R. 2016. Filipino speech corpus, University of the Philippines – Sentro ng Wikang Filipino

[3] Chowdhury, M., Islam, M., and Bipul, B. 2015. Implementation of an Optical Character Reader (OCR) for Bengali language. DOI= https://doi.org/10.1109/ICODSE.2015.7436984

[4] Corpus, M., et al. 2009. Development of a Filipino TTS System Using Concatenative Speech Synthesis, Undergraduate Thesis, University of the Philippines.

[5] Guevara, R., Co, M., Espina, E., Garcia, I., Tan, E., Ensomo, R., and Sagum, R. Development of a Filipino speech corpus. in 3rd National ECE Conference, Philippines, 2002

[6] Ghosh, T., Saha, S., and Iftekharul Ferdous, H. M., Formant Analysis of Bangla Vowel for Automatic Speech Recognition, Signal & Image Processing: An International Journal, vol.07, October 2016.

[7] Guo, H., Jin, T. and Mettas A. 2011. Designing reliability demonstration tests for one-shot systems under zero component failure," IEEE Transactions on Reliability, 286-294. DOI= https://doi.org/10.1109/TR.2010.2085552

[8] Hamiti, M. and Kastrati, R. 2014. Adapting eSpeak for converting text into speech in Albanian, South East European University.

[9] Lazaro, L. S., Policarpio, L. L., and Guevara, R. L. 2009. Incorporating Duration and Intonation Models in Filipino Speech Synthesis, Undergraduate Thesis, UP Diliman.

[10] Lee, S. H., Hsiao, T. Y., and Lee, G. S., Audio–vocal responses of vocal fundamental frequency and formant during sustained vowel vocalizations in different noises, Hearing Research, 324 (June 2015) , pp. 1- 6.

[11] Mesa, Q. B. and Kyung, T. 2014. Development of Tagalog speech corpus. in International Conference on Multidisciplinary Trends in Academic Research, Thailand.

[12] Prasanth, K., Introduction to Voice Recognition with Elechouse V3 and Arduino, Retrieved November, 2017, from https://www.instructables.com/id/Introduction-to-Voice-Recognition-With-Elechouse-V/

[13] Richardson, M. and Wallace, S. 2013. Getting started with Raspberry Pi, Sebastopol, CA: Maker Media, Inc.

[14] Sanjana, M. A. and Usha, B. Data and Structure adaptive Optical Character Reader, International Journal of Computer Systems, 2 (April 2015).

[15] Taya, D. M. 2014. Towards Expressive Arabic Text to Speech, Cairo University

[16] Viswanathan, M. and Viswanathan, M. 2005. Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (MOS) scale, 55-83. DOI= https://doi.org/10.1016/j.csl.2003.12.001.

[17] Voice Recognition Module V3, Retrieved August, 2018, from https://www.elechouse.com/elechouse/images/product/VR3/VR3_manual.pdf

[18] Wood, S. 2005, What are formants?, Retrieved November, 2017, from http://person2.sol.lu.se/SidneyWood/praate/whatform.html